

2016 Internet Survival Guide

An Edited Guide of How to Stay Safe on the Internet

By: DADOES

PREFACE	12
A DECLARATION OF THE INDEPENDENCE OF CYBERSPACE.....	13
GOVERNMENT CORRUPTION AND YOUR RIGHTS TO PRIVACY, ENCRYPTION, FREEDOM, AND PEACE OF MIND	15
INTRODUCTION.....	15
GOVERNMENT INFRINGEMENT ON PRIVACY	16
METADATA.....	23
MAN IN THE MIDDLE (MITM).....	24
YOUR RIGHTS TO PRIVACY	24
WHY NOT WINDOWS –.....	27
RESTRICTIONS	27
THE SMALL PRINT IN THE CONTRACT.....	27
THE MEANING BEHIND	28
MORE DETAILS ON RESTRICTIONS	28
POOR SUPPORT FOR OPEN STANDARDS.....	29
STANDARDS THAT CHANGE ALL THE TIME	29
DEFAULT PROGRAMS YOU CAN'T UNINSTALL.....	30
MONOPOLISTIC PRACTICES.....	30
WHAT ABOUT SOURCE CODE?	30
INSECURITY MEANS BUSINESS	31
YOU CAN TRUST FREE SOFTWARE.....	31
PROFESSOR NORM MATLOFF'S BEGINNER'S GUIDE TO INSTALLING AND USING LINUX	32
1 BACKGROUND NEEDED.....	32
2 INSTALL TO WHERE?.....	32
3 WHICH LINUX DISTRIBUTION IS BEST?	32
4 INSTALLATION.....	33
4.1 <i>The Short Answer</i>	33
4.2 <i>Installing Linux to a USB Key or External Hard Drive</i>	33
4.2.1 <i>Installation Method I (for Slax Linux)</i>	33
4.2.2 <i>Other Methods</i>	34
5 POST-INSTALLATION CONFIGURATION	34
5.1 <i>Configuring Your Search Path ("Why can't I run my a.out?")</i>	34
5.2 <i>Configuring a Printer</i>	34
5.3 <i>Switching from GNOME/Ubuntu Unity</i>	34
5.4 <i>Configuring KDE/GNOME for Convenient Window Operations</i>	35
5.4.1 <i>Autoraise Etc.</i>	35
6 SOME POINTS ON LINUX USAGE.....	35
6.0.2 <i>Ubuntu Root Operations</i>	35
6.1 <i>More on Shells/Terminal Windows</i>	36
6.2 <i>Cut-and-Paste Window Operations</i>	36
6.3 <i>Mounting Other Peripheral Devices</i>	36
6.3.1 <i>Mount Points</i>	36
6.3.2 <i>Using USB Devices</i>	37
7 LINUX APPLICATIONS SOFTWARE.....	37
7.1 <i>GUI Vs. Text-Based</i>	37
7.2 <i>My Favorite Unix/Linux Apps</i>	38
7.2.1 <i>Text Editing</i>	38
7.2.2 <i>Web Browsing and Java</i>	38
7.2.3 <i>HTML Editing</i>	38

7.2.4 Compilers.....	38
7.2.5 Integrated Software Development (IDE).....	39
7.2.6 Word Processing.....	39
7.2.7 Playing Movies, Music, Etc.	39
7.2.8 Video Editing	41
7.2.9 Image Viewing, Manipulation and Drawing	41
7.2.10 FTP.....	41
7.2.11 Statistical Analysis.....	41
7.2.12 Video Chat	41
7.2.13 Running Windows Applications from Within Linux	42
7.3 Downloading New Software.....	42
7.3.1 How to Find It	42
7.3.2 Automatic Download/Installation.....	42
7.3.3 Debian/Ubuntu .deb Files.....	43
7.3.4 Using RPMs	43
8 LEARNING MORE ABOUT LINUX	43
9 ADVANCED LINUX USAGE	44
9.1 Dual-Boot Issues.....	44
9.2 Live CDs or USB-Key Based Linux As Rescue Tools.....	44
9.3 Troubleshooting	45
9.3.1 Tools	45
9.3.2 WiFi Networking.....	45
9.3.3 General Information.....	45
9.3.4 Network Management Tools.....	46
9.3.5 Individual Linux Network Commands	46
9.3.6 If You Have a Problem	47
9.3.7 A Program Freezes	48
9.3.8 Screen Freezes	48
9.4 Accessing Your Windows Files from Linux	48
A WHAT IS LINUX?	49
B WHAT IS PARTITIONING?	50
B.1 Partitioning Using GParted	50
TOR – THE ONION RELAY	52
TOR: OVERVIEW	52
Topics	52
OVERVIEW.....	52
WHY WE NEED TOR.....	53
THE SOLUTION: A DISTRIBUTED, ANONYMOUS NETWORK	53
STAYING ANONYMOUS	55
THE FUTURE OF TOR.....	55
TAILS	56
ONLINE ANONYMITY AND CENSORSHIP CIRCUMVENTION	56
TOR.....	56
I2P	57
USE ANYWHERE BUT LEAVE NO TRACE	57
STATE-OF-THE-ART CRYPTOGRAPHIC TOOLS.....	57
WHAT'S NEXT?.....	58
INSTALLING TAILS	59
MANUAL INSTALLATION USING LINUX	59
MANUAL INSTALLATION USING WINDOWS.....	61
MANUAL INSTALLATION USING MAC	65

JOLLY ROGER'S SECURITY THREAD FOR BEGINNERS	67
INTRODUCTION TO SECURE COMMUNICATION - TOR, HTTPS, SSL	68
PGP, TAILS, VIRTUAL BOX	69
PGP CONTINUED	72
WHOLE DISK ENCRYPTION AND FILE SHREDDING	74
JAVASCRIPT VULNERABILITIES AND REMOVING PERSONAL METADATA FROM FILES.....	76
GENERAL SECURITY PRECAUTIONS WHEN POSTING ONLINE, LEARN FROM OTHERS' MISTAKES	78
EXIF DATA.....	79
RETAINING A LAWYER, HOW TO HANDLE GETTING CAUGHT OR INTERROGATED	80
COMBINING TOR WITH A VPN.....	81
COMBINING TOR WITH A VPN CONTINUED	84
TRACKING COOKIES.....	87
LEARNING FROM OTHERS' MISTAKES. LIBERTAS, DPR, SABU, LULZSEC	88
HOW FAR WILL LAW ENFORCEMENT GO?	89
FRAUDULENT PRIVATE MESSAGES.....	92
LEARNING FROM OTHERS' MISTAKES. HOW THEY BUSTED SABU	93
LEARNING FROM OTHERS' MISTAKES. SABU BECAME FBI INFORMANT AND BETRAYED JEREMY HAMMOND.....	96
WHERE YOU MIGHT CONSIDER RUNNING TO, IF YOU HAD NO OTHER CHOICE	99
SECURING YOUR ACCOUNT FROM FBI MONITORING	101
HOW TO CONNECT TO TOR OVER TOP OF TOR	103
HOW TO VERIFY YOUR DOWNLOADED FILES ARE AUTHENTIC.....	104
VERIFYING SIGNED MESSAGES WITH SIGNATURES AND SIGNING YOUR OWN MESSAGES.....	109
AN EXAMPLE OF REALLY BAD OPSEC - SMARTEN UP!.....	112
TOR CHAT	114
OBTAINING, SENDING AND RECEIVING BITCOINS ANONYMOUSLY	116
CLEARNET VS HIDDEN SERVICES - WHY YOU SHOULD BE CAREFUL.....	120
THEY ARE WATCHING YOU - VIRUSES, MALWARE, VULNERABILITIES	121
MONITORING YOU WITH AN ANTENNA	124
COOKIES & JAVASCRIPT REVISITED, PLUS FLASH COOKIES AND OTHER BROWSER TRACKING	127
A FEW RECOMMENDATIONS	129
COLD BOOT ATTACKS, UNENCRYPTED RAM EXTRACTION	130
THE STRENGTH OF CRYPTOGRAPHY AND ANONYMITY WHEN USED PROPERLY	136
ANOTHER SCAM EMAIL - BEWARE	138
AN INTRODUCTION TO AN EXPERT ON OPSEC, PLUS MD5 & SHA-1 CHECKSUMS	139
IT IS OBVIOUS WHEN YOU ARE USING TOR	142
ARE YOU USING SAFE-MAIL.NET ?.....	143
LOCALBITCOINS PART 1 - POLICE ARE WATCHING IT!.....	144
LOCALBITCOINS PART 2 - THIEVES, SCAMMERS AND COUNTERFEIT BILLS!.....	147
LOCALBITCOINS PART 3 - MORE SCAM STORIES.....	151
LOCALBITCOINS PART 4 - SELLERS BUSTED FOR MONEY LAUNDERING.....	154
HIDING TOR FROM YOUR ISP - PART 1 - BRIDGES AND PLUGGABLE TRANSPORTS	156
CAPABILITIES OF THE NSA	165
WHY YOU SHOULD ALWAYS BACK UP YOUR DRIVES, ESPECIALLY ENCRYPTED DRIVES.....	166
BITCOIN CLIENTS IN TAILS - BLOCKCHAIN AND ELECTRUM	167
YET ANOTHER EXAMPLE OF HOW STRONG CRYPTOPGRAPHY AND PROPER OPSEC CAN PROTECT EVEN PEDOPHILES....	169
DENIABILITY, IDENTIFYING TAILS USERS, AND CAN YOU BE FORCED TO GIVE UP YOUR PASSWORDS?	174
SECURITY CULTURE: A HANDBOOK FOR ACTIVISTS.....	181
INTRODUCTION:.....	181
SECURITY WHAT IT IS, WHY WE NEED IT AND HOW WE IMPLEMENT IT.....	182
SO WHAT IS A SECURITY CULTURE?.....	182
WHAT NOT TO SAY	182
THREE EXCEPTIONS	183
SECURITY MEASURES	183

SECURITY VIOLATING BEHAVIOURS	184
EDUCATE TO LIBERATE	184
DEALING WITH CHRONIC SECURITY PROBLEMS	185
A BRIEF PRIMER ON THE CANADIAN STATE SECURITY APPARATUS	185
AN OVERVIEW OF DOMESTIC INTELLIGENCE ORGANIZATIONS	186
THE COUNTER- INSURGENCY MODEL.....	187
EVERYTHING YOU EVER WANTED TO KNOW ABOUT INFORMERS AND INFILTRATORS	188
CRYPTO ANARCHY AND VIRTUAL COMMUNITIES	190
EXTENDED ABSTRACT.....	190
1 INTRODUCTION	191
2 MODERN CRYPTOGRAPHY.....	192
3 VIRTUAL COMMUNITIES	193
4 OBSERVABILITY AND SURVEILLANCE	194
5 CRYPTO ANARCHY	194
6 TRUE NAMES AND ANONYMOUS SYSTEMS	195
7 EXAMPLES AND USES	196
8 COMMERCE AND COLONIZATION OF CYBERSPACE	197
9 IMPLICATIONS.....	197
10 HOW LIKELY?	198
11 CONCLUSIONS	199
12 ACKNOWLEDGMENTS.....	200
13 REFERENCES AND NOTES	200
THE CRYPTOPARTY HANDBOOK.....	203
1 INTRODUCING CRYPTOPARTY.....	211
1.1 ABOUT THIS BOOK	211
1.2A CRYPTOPARTY MANIFESTO	213
1.3HOW TO CRYPTOPARTY	215
1.4PARTY LIKE IT'S DECEMBER 31ST 1983	218
1.4.1 <i>What is CryptoParty?</i>	218
1.5PREFACE	218
1.6WHY PRIVACY MATTERS	219
2 UNDERSTANDING EMAIL	220
2.1BASIC TIPS.....	220
2.1.1 <i>In brief:</i>	220
2.1.2 <i>Passwords</i>	220
2.1.3 <i>Reading Email in Public Places</i>	221
2.1.4 <i>Cache Cunning</i>	221
2.1.5 <i>Securing your communication</i>	222
2.1.6 <i>DNSSEC & DANE</i>	222
2.1.7 <i>Account Separation</i>	222
2.1.8 <i>A note about hosted email</i>	223
2.2TYPES OF EMAIL.....	223
2.2.1 <i>Remotely hosted email ('webmail'), resourced using a web browser</i>	223
2.2.2 <i>Remotely hosted email, resourced using an email program or using a web browser</i>	223
2.2.3 <i>Context considerations</i>	224
2.2.4 <i>Email & Metadata</i>	224
2.2.5 <i>Self-administered email server</i>	225
2.2.6 <i>'Free' email services</i>	225
2.2.7 <i>Non-profit</i>	225
2.2.8 <i>Notes on email forwarding</i>	225
2.3FEARS.....	225

2.3.1	Random abuse and theft by malicious hackers	226
2.3.2	Targeted abuse, harassment, and spying	227
2.3.3	When Encryption Goes Wrong	228
2.4	SECURE CONNECTIONS.....	228
2.4.1	Can other people read along when I check my email?.....	228
2.4.2	Notes	229
2.5	SECURE EMAILS	229
2.5.1	What software can I use to encrypt my email?.....	230
3	UNDERSTANDING BROWSING	230
3.1	BASIC TIPS.....	230
3.1.1	In Brief:.....	230
3.1.2	Your browser talks about you behind your back	230
3.1.3	Web sites can track you as you browse.....	231
3.1.4	Searching online can give away information about you	231
3.1.5	More eyes than you can see.....	231
3.1.6	Your right to be unknown.....	232
3.2	FEARS	232
3.2.1	Social Networking - what are the dangers?	232
3.2.2	Who can steal my identity?.....	233
3.2.3	Can I get in trouble for Googling weird stuff?.....	234
3.2.4	Who is keeping a record of my browsing and am I allowed to hide from them?.....	234
3.2.5	How to not reveal my Identity?.....	234
3.2.6	How to avoid being tracked?.....	234
3.3	WHAT HAPPENS WHEN YOU BROWSE	234
3.3.1	A topography of you: footprints.....	235
3.4	ACCOUNTS AND SECURITY.....	237
3.4.1	Can malicious web sites take over my accounts?.....	237
3.5	TRACKING.....	238
3.5.1	How do they track us?.....	238
3.5.2	How can I prevent tracking?	239
3.5.3	A word of warning.....	245
3.6	ANONYMITY.....	245
3.6.1	Intro.....	245
3.6.2	Proxy.....	246
3.6.3	Tor	246
3.7	VPN.....	247
4	PUBLISHING AND DISTRIBUTION	249
4.1	PUBLISHING ANONYMOUSLY	249
4.1.1	Several Don'ts.....	250
4.2	ANONYMOUS EMAIL.....	251
4.2.1	Sending From Throw-away Email Accounts	251
4.2.2	Be Careful about what you say!	251
4.3	FILE SHARING	252
4.3.1	BitTorrent	253
4.3.2	SoulSeek	254
4.3.3	I2P	255
5	SECURE CALLS AND SMS	256
5.1	SECURE CALLS	256
5.1.1	iOS - Installing Signal.....	256
5.1.2	Android - Installing RedPhone.....	256
5.2	SECURE MESSAGING.....	256
5.2.1	Android.....	257

6 BASIC EMAIL SECURITY	257
6.1 START USING THUNDERBIRD	257
6.1.1 Installing Thunderbird on Windows	257
6.1.2 Installing Thunderbird on Ubuntu	261
6.1.3 Installing Thunderbird on Ubuntu 12.04 or newer	261
6.1.4 Installing Thunderbird on Mac OS X	263
6.1.5 Starting Thunderbird for the first time	265
6.2 SETTING UP SECURE CONNECTIONS	266
6.2.1 Configuration requirements	266
6.2.2 Preparing a Gmail account for use with Thunderbird	266
6.2.3 Configuring Thunderbird to use SSL/TLS	267
6.2.4 Manual setup	268
6.2.5 Finishing the setup, different encryption methods	271
6.2.6 Returning to the configuration screens	271
6.3 SOME ADDITIONAL SECURITY SETTINGS	271
6.3.1 Junk mail settings	272
6.3.2 Scam detection and warning system	273
6.3.3 Anti-virus integration	274
6.3.4 Set a master password	275
6.3.5 Adaptive junk mail controls	278
7 EMAIL ENCRYPTION	280
7.1 INTRODUCING MAIL ENCRYPTION (PGP)	280
7.1.1 Using a key-pair to encrypt your mail	281
7.1.2 Sending encrypted mails to other people: you need their public key	281
7.1.3 Receiving encrypted mails from other people: they need my public key	281
7.1.4 Conclusion: encryption requires public key distribution!	282
7.2 INSTALLING PGP ON WINDOWS	282
7.2.1 Installing PGP (GPG) on Microsoft Windows	282
7.2.2 Installing with the Enigmail extension	283
7.2.3 Installation steps	283
7.3 INSTALLING PGP ON OSX	285
7.3.1 Getting started	285
7.3.2 Downloading and installing the Software	285
7.3.3 Installing up Enigmail	293
7.4 INSTALLING PGP ON UBUNTU	295
7.5 INSTALLING GPG ON ANDROID	296
7.5.1 APG	296
7.5.2 GPG enabled e-mail on Android: K-9 Mail	297
7.6 CREATING YOUR PGP KEYS	297
7.7 DAILY PGP USAGE	306
7.7.1 Encrypting attachments	307
7.7.2 Entering your pass-phrase	307
7.7.3 Receiving encrypted e-mails	308
7.7.4 Sending and receiving public keys	308
7.7.5 Receiving public keys and adding them to your keyring	309
7.7.6 Using public key servers	312
7.7.7 Signing emails to an individual	317
7.7.8 Sending encrypted mails to an individual	318
7.7.9 Automating encryption to certain recipients	319
7.7.10 Verifying incoming e-mails	323
7.7.11 Revoking your GPG key-pair	324
7.7.12 What to do when you have lost your secret key, or forgot your passphrase	325
7.7.13 What to do when your secret key has been stolen, or compromised	325

7.7.14	Receiving a revocation certificate	325
7.7.15	Preparing for the worst: backup your keys	326
7.7.16	Further reading	327
7.8	WEBMAIL AND PGP	327
8	SAFER BROWSING.....	328
8.1	WHY FIREFOX?.....	328
8.2	ACCESSING FIREFOX ON UBUNTU	328
8.3	INSTALLING ON MAC OS X	329
8.4	INSTALLING FIREFOX ON WINDOWS.....	334
8.4.1	Troubleshooting	338
8.5	EXTENDING FIREFOX.....	338
8.5.1	HTTPS Everywhere.....	338
8.5.2	Installation	339
8.5.3	Configuration	340
8.5.4	Usage	341
8.5.5	If networks block HTTPS	343
8.5.6	Adding support for additional sites in HTTPS Everywhere	343
8.5.7	Enforcing secure HTTPS server connections	344
8.5.8	Adblock Plus	344
8.5.9	Getting started with Adblock Plus.....	344
8.5.10	Choosing a filter subscription	345
8.5.11	Creating personalized filters.....	346
8.5.12	Enabling and disabling Adblock Plus for specific elements or Web sites	346
8.5.13	Other extensions that can improve your security.....	346
8.6	PROXY SETTINGS.....	347
8.6.1	Default Firefox proxy configuration	347
8.7	USING TOR?	349
8.7.1	Using Tor Browser Bundle	350
8.7.2	Downloading Tor Browser Bundle.....	350
8.7.3	Running a Relay or Bridge	351
8.8	EXTENDING GOOGLE CHROME.....	351
8.8.1	Disabling Instant Search.....	351
8.8.2	Adblock for Chrome.....	351
8.8.3	HTTPS Everywhere.....	351
8.8.4	PrivacyFix	351
9	PASSWORDS.....	352
9.1	KEEPING PASSWORDS SAFE	352
9.1.1	Password length and complexity.....	352
9.1.2	Easy to remember and secure passwords	352
9.1.3	Minimizing damage.....	352
9.1.4	Using a password manager.....	352
9.1.5	Physical protection	353
9.1.6	Other caveats	353
9.2	INSTALLING KEEPASS	353
9.2.1	Installing KeePassX on Ubuntu	353
9.2.2	Installing KeePass on Windows	354
9.2.3	Installing KeePass on Mac OS X.....	360
9.3	ENCRYPTING PASSWORDS WITH A PASSWORD MANAGER	367
9.3.1	Encrypting Passwords with KeePassX on Ubuntu.....	367
9.3.2	Encrypting Passwords with KeePass on Windows.....	373
9.3.3	Encrypting Passwords with Keychain on Mac OSX.....	379
10	USING VPN	382

10.1	GETTING, SETTING-UP AND TESTING A VPN ACCOUNT.....	382
10.1.1	An account from a commercial VPN provider	382
10.1.2	Setting up OpenVPN client	384
10.1.3	Caveats & Gotchas	385
10.2	VPN ON UBUNTU	385
10.2.1	Preparing Network Manager for VPN networks	385
10.2.2	Configuring an OpenVPN network	390
10.2.3	Using your new VPN connection	396
10.3	VPN ON MACOSX	398
10.3.1	Setup	398
10.4	VPN ON WINDOWS	411
10.4.1	Setup	411
10.5	MAKING SURE YOUR VPN WORKS.....	424
11	DISK ENCRYPTION *** TRUENCRYPT COMPROMISED***	425
11.1	INSTALLING VERACRYPT	425
11.1.1	Installing on Ubuntu/Debian	425
11.1.2	Installing on OSX	428
11.1.3	Installing on Windows	431
11.2	USING VERACRYPT	432
11.2.1	Creating a VeraCrypt Container	432
11.2.2	Mounting the Encrypted Volume	438
11.2.3	What does this mean?.....	441
11.2.4	Remember to dismount!	441
11.3	SETTING UP A HIDDEN VOLUME.....	441
11.4	SECURELY DESTROYING DATA.....	445
11.4.1	A note on Solid State Hard Drives.....	446
11.4.2	Securely delete data under Windows.....	446
11.4.3	Securely delete data under MacOSX	448
11.4.4	Securely delete data under Ubuntu/Linux.....	452
11.5	ABOUT LUKS.....	460
11.5.2	Encrypting a device	461
11.5.3	Using an encrypted device	464
12	CALL ENCRYPTION	465
12.1	INSTALLING CSIPSIMPLE.....	465
12.1.1	Introducing The OSTN Network.....	465
12.1.2	CSipSimple	466
13	INSTANT MESSAGING ENCRYPTION	470
13.1	SETTING UP ENCRYPTED INSTANT MESSAGING.....	470
13.1.1	Android - Installing Gibberbot.....	470
13.1.2	iOS - Installing ChatSecure	470
13.1.3	Ubuntu - Installing Pidgin.....	470
13.1.4	OS X - Installing Adium	470
13.1.5	Windows - Installing Pidgin.....	471
13.1.6	All OS - crypto.cat.....	471
13.1.7	Chat Log Files	472
14	SECURE FILE SHARING.....	472
14.1	INSTALLING I2P ON UBUNTU LUCID LYNX (AND NEWER) AND DERIVATIVES LIKE LINUX MINT & TRISQUEL.....	472
14.2	INSTRUCTIONS FOR DEBIAN LENNY AND NEWER	474
14.3	STARTING I2P	474
14.4	ANONYMOUS BITTORRENT WITH I2PSNARK.....	475

15 APPENDICES.....	476
15.1 CRYPTOGRAPHY AND ENCRYPTION	476
15.1.1 Encryption examples	477
15.1.2A Warning!	477
15.1.3 Historical ciphers	477
15.1.4 Modern ciphers	480
15.1.5 Quantum Cryptography	481
15.1.6 Challenges & Implications	481
15.2 GLOSSARY	481
15.2.1 aggregator	481
15.2.2 anonymity	481
15.2.3 anonymous remailer	482
15.2.4 ASP (application service provider)	482
15.2.5 backbone	482
15.2.6 badware	482
15.2.7 bandwidth	482
15.2.8 bash (Bourne-again shell)	482
15.2.9 BitTorrent	482
15.2.10 blacklist	482
15.2.11 bluebar	483
15.2.12 block	483
15.2.13 bookmark	483
15.2.14 bridge	483
15.2.15 brute-force attack	483
15.2.16 cache	483
15.2.17 censor	483
15.2.18 censorware	483
15.2.19 CGI (Common Gateway Interface)	484
15.2.20 chat	484
15.2.21 cipher	484
15.2.22 circumvention	484
15.2.23 Common Gateway Interface	484
15.2.24 command-line interface	484
15.2.25 cookie	484
15.2.26 country code top-level domain (ccTLD)	484
15.2.27 cryptography	485
15.2.28 DARPA (Defense Advanced Projects Research Agency)	485
15.2.29 decryption	485
15.2.30 disk encryption	485
15.2.31 domain	485
15.2.32 DNS (Domain Name System)	485
15.2.33 DNS leak	485
15.2.34 DNS server	485
15.2.35 DNS tunnel	486
15.2.36 Eavesdropping	486
15.2.37 e-mail	486
15.2.38 embedded script	486
15.2.39 encryption	486
15.2.40 exit node	486
15.2.41 file sharing	487
15.2.42 file spreading engine	487
15.2.43 filter	487
15.2.44 Firefox	487
15.2.45 forum	487

15.2.46	frame	487
15.2.47	FTP (File Transfer Protocol)	487
15.2.48	full disk encryption	487
15.2.49	gateway	488
15.2.50	GNU Privacy Guard	488
15.2.51	GPG	488
15.2.52	honeypot	488
15.2.53	hop	488
15.2.54	HTTP (Hypertext Transfer Protocol)	488
15.2.55	HTTPS (Secure HTTP)	488
15.2.56	IANA (Internet Assigned Numbers Authority)	488
15.2.57	ICANN (Internet Corporation for Assigned Names and Numbers)	489
15.2.58	Instant Messaging (IM)	489
15.2.59	Intermediary	489
15.2.60	Internet	489
15.2.61	IP (Internet Protocol) Address	489
15.2.62	IRC (Internet relay chat)	489
15.2.63	ISP (Internet Service Provider)	489
15.2.64	JavaScript	489
15.2.65	KeePass, KeePassX	489
15.2.66	keychain software	489
15.2.67	keyword filter	490
15.2.68	latency	490
15.2.69	log file	490
15.2.70	low-bandwidth filter	490
15.2.71	malware	490
15.2.72	man in the middle	490
15.2.73	middleman node	490
15.2.74	monitor	490
15.2.75	network address translation (NAT)	491
15.2.76	network operator	491
15.2.77	node	491
15.2.78	non-exit node	491
15.2.79	obfuscation	491
15.2.80	open node	491
15.2.81	OTR/Off-the-Record messaging	491
15.2.82	packet	491
15.2.83	password manager	492
15.2.84	pastebin	492
15.2.85	peer-to-peer	492
15.2.86	perfect forward secrecy	492
15.2.87	Pretty Good Privacy (PGP)	492
15.2.88	PHP	492
15.2.89	plain text	492
15.2.90	plaintext	493
15.2.91	privacy	493
15.2.92	private key	493
15.2.93	POP3	493
15.2.94	port	493
15.2.95	protocol	493
15.2.96	proxy server	493
15.2.97	Psiphon node	493
15.2.98	private node	494
15.2.99	public key	494
15.2.100	public key encryption/public-key cryptography	494

15.2.101publicly routable IP address	494
15.2.102regular expression	494
15.2.103remailer	494
15.2.104router	494
15.2.105root name server	495
15.2.106RSS (Real Simple Syndication)	495
15.2.107scheme	495
15.2.108shell	495
15.2.109SOCKS	495
15.2.110screenlogger	495
15.2.111script	495
15.2.112smartphone	496
15.2.113spam	496
15.2.114SSH (Secure Shell)	496
15.2.115SSL (Secure Sockets Layer)	496
15.2.116steganography	496
15.2.117subdomain	496
15.2.118threat analysis	496
15.2.119Top-Level Domain (TLD)	497
15.2.120TLS (Transport Layer Security)	497
15.2.121TCP/IP (Transmission Control Protocol over Internet Protocol)	497
15.2.122Tor bridge	497
15.2.123traffic analysis	497
15.2.124tunnel	497
15.2.125UDP (User Datagram Packet)	497
15.2.126URL (Uniform Resource Locator)	497
15.2.127Usenet	498
15.2.128VoIP (Voice over Internet Protocol)	498
15.2.129VPN (virtual private network)	498
15.2.130whitelist	498
15.2.131World Wide Web (WWW)	498
15.2.132Webmail	498
15.2.133Web proxy	499
15.2.134WHOIS	499
15.3THE NECESSITY OF OPEN SOURCE	499

Preface

I was unaware of how to go about this project when I first began it. I thought that maybe I should write all of the guides by hand using my own knowledge. However, I found that there are others who have been able to construct specific guides in a more informative manner than myself. From this I thought that it might be useful to use other people's guides in my work and then I would pick up where they left off. Yet, after doing research I thought that it would be best to just combine the best guides that I found into one document and folder. I truly hope that this guide is useful for at least one person on the internet. I will begin this guide with a brief overview of government corruption and why you should care about your privacy. After that, everything else is either a guide or text written by others which I have fully credited. If there is one thing that I can say before reading this guide, it would be to **read Jolly Roger's Guide**. It is probably the most down to earth and comprehensible guide for anyone at any skill level. Thank you for downloading and reading this file.

A Declaration of the Independence of Cyberspace

by John Perry Barlow <barlow@eff.org>

Governments of the Industrial World, you weary giants of flesh and steel, I come from Cyberspace, the new home of Mind. On behalf of the future, I ask you of the past to leave us alone. You are not welcome among us. You have no sovereignty where we gather.

We have no elected government, nor are we likely to have one, so I address you with no greater authority than that with which liberty itself always speaks. I declare the global social space we are building to be naturally independent of the tyrannies you seek to impose on us. You have no moral right to rule us nor do you possess any methods of enforcement we have true reason to fear.

Governments derive their just powers from the consent of the governed. You have neither solicited nor received ours. We did not invite you. You do not know us, nor do you know our world. Cyberspace does not lie within your borders. Do not think that you can build it, as though it were a public construction project. You cannot. It is an act of nature and it grows itself through our collective actions.

You have not engaged in our great and gathering conversation, nor did you create the wealth of our marketplaces. You do not know our culture, our ethics, or the unwritten codes that already provide our society more order than could be obtained by any of your impositions.

You claim there are problems among us that you need to solve. You use this claim as an excuse to invade our precincts. Many of these problems don't exist. Where there are real conflicts, where there are wrongs, we will identify them and address them by our means. We are forming our own Social Contract. This governance will arise according to the conditions of our world, not yours. Our world is different.

Cyberspace consists of transactions, relationships, and thought itself, arrayed like a standing wave in the web of our communications. Ours is a world that is both everywhere and nowhere, but it is not where bodies live.

We are creating a world that all may enter without privilege or prejudice accorded by race, economic power, military force, or station of birth.

We are creating a world where anyone, anywhere may express his or her beliefs, no matter how singular, without fear of being coerced into silence or conformity.

Your legal concepts of property, expression, identity, movement, and context do not apply to us. They are all based on matter, and there is no matter here.

Our identities have no bodies, so, unlike you, we cannot obtain order by physical coercion. We believe that from ethics, enlightened self-interest, and the commonweal, our governance will emerge. Our identities may be distributed across many of your jurisdictions. The only law that all our constituent cultures would generally recognize is the Golden Rule. We hope we will be able to build our particular solutions on that basis. But we cannot accept the solutions you are attempting to impose.

In the United States, you have today created a law, the Telecommunications Reform Act, which repudiates your own Constitution and insults the dreams of Jefferson, Washington, Mill, Madison, DeToqueville, and Brandeis. These dreams must now be born anew in us.

You are terrified of your own children, since they are natives in a world where you will always be immigrants. Because you fear them, you entrust your bureaucracies with the parental responsibilities you are too cowardly to confront yourselves. In our world, all the sentiments and expressions of humanity, from the debasing to the angelic, are parts of a seamless whole, the global conversation of bits. We cannot separate the air that chokes from the air upon which wings beat.

In China, Germany, France, Russia, Singapore, Italy and the United States, you are trying to ward off the virus of liberty by erecting guard posts at the frontiers of Cyberspace. These may keep out the contagion for a small time, but they will not work in a world that will soon be blanketed in bit-bearing media.

Your increasingly obsolete information industries would perpetuate themselves by proposing laws, in America and elsewhere, that claim to own speech itself throughout the world. These laws would declare ideas to be another industrial product, no more noble than pig iron. In our world, whatever the human mind may create can be reproduced and distributed infinitely at no cost. The global conveyance of thought no longer requires your factories to accomplish.

These increasingly hostile and colonial measures place us in the same position as those previous lovers of freedom and self-determination who had to reject the authorities of distant, uninformed powers. We must declare our virtual selves immune to your sovereignty, even as we continue to consent to your rule over our bodies. We will spread ourselves across the Planet so that no one can arrest our thoughts.

We will create a civilization of the Mind in Cyberspace. May it be more humane and fair than the world your governments have made before.

Davos, Switzerland

February 8, 1996

Government Corruption and Your Rights to Privacy, Encryption, Freedom, and Peace of Mind

By: DADOES

Introduction

Would you allow someone to stalk you and know where you are going to be at all times, know what you recently purchased, know who you are talking to, what you are saying, and what you are doing at all times?

Today we live in a world where the majority of all information is stored online in some form or another. This information spans from simple updates of the weather to possibly some of your most private and intimate moments sent via SMS, email, instant messaging, or other forms of communication via internet. Some information, such as the weather, is hosted on public domains for everyone to see while the conversations that you may have with family members, a significant other, coworkers, etc. are usually stored on private servers owned by private companies. I am sure that the majority of you would not like to publicly share many of the conversations that you have had with such people. However, in recent years it has come to our attention that these intimate conversations have been being actively monitored by the National Security Agency (NSA), Government Communications Headquarters (GCHQ), and others. Not only are these private conversations being monitored but so are our:

- Online Banking Transactions
- Pictures sent via SMS, Email, and Instant Messaging
- Phone Records
- Locations
- Internet Browser History
- Google Searches
- Social Media Activity (Facebook, Twitter, Instagram, etc.)

Many people do not seem to actively have any concerns over the fact that all of these aspects of their life are being monitored though. Some use the common “nothing to hide, nothing to fear argument” while others use the argument that, for layman's terms, “there are bigger fish to fry than me.” There are many problems with both of these arguments and I will go into them in further detail later. The main point that needs to be conveyed here is why you should care that all of these parts of your life are being monitored. First of all, if you are aware that you are constantly being surveyed then it is more likely that you are going to be cautious as to what you will say or send through the internet. This is what is called a chilling effect. The true definition of a chilling effect is “the inhibition or discouragement of the legitimate exercise of natural and legal rights by the threat of legal sanction.” This could even span as far as far as you being afraid to look up the latest news on terrorist attacks around the world because your searches are monitored and therefore you may be associated with a terrorist affiliation. This chilling effect also reduces creativity amongst people and their peers as well as scrutiny to those in power... which are supposed to be given scrutiny by us, the people. Another reason that you care is because there are people spying on every part of your life. Would you agree to let a complete stranger set up a microphone or camera into your room and then agree to carry it around with you everywhere you go? I most certainly think that you would not. Yet, that is exactly what you are doing with your latest smartphones. Would you agree to show random strangers risqué pictures that your significant other sent to you? According to this [article](#), Edward Snowden states that such pictures

have been being around in underground trading circles throughout the NSA. Would you allow someone to stalk you and know where you are going to be at all times, know what you recently purchased, know who you are talking to, what you are saying, and what you are doing at all times? If you are a regular user of the internet, a debit or credit card, a smartphone, or anything of the 21st century then all of this is happening to you on a daily basis.

I will begin this guide with background information on how your internet activity has been being monitored constantly, who is monitoring it, and what the implications are. After that I will go into the programs and practices that you should use in order to keep your personal information safe. I have included shortcuts to topics in the table of contents if you do not wish to read the background information. However, if you are not familiar with it I do advise you to read it.

Government Infringement on Privacy

The modern world was shaken in 2013 by the leaks of Edward Snowden. These leaks revealed that the NSA, GCHQ, and an overarching intelligence alliance deemed the “Five Eyes” had been conducting mass surveillance that had only been imagined in Orwellian fiction.¹ However, the uses of mass surveillance have been around long before the Snowden leaks. In 2001, shortly after the attacks of September 11, the program Stellarwind was implemented. Stellarwind was the code name of information collected under the President's Surveillance Program. With this program implemented, the United States government was able to conduct large scale data mining of the communications of American citizens. This large scale data mining did not require warrants in order to collect the information of American citizens. Snowden would later detail the Stellarwind program in great length with his leaks. The amount of preparation in which the governments of the world have had leading up to this point in mass surveillance is vast and therefore I will not be discussing it here. However, I do plan on detailing in later on in a separate paper. What I will focus on here are the Snowden leaks and the era which I will refer to as post-Snowden.

Snowden revealed to us what is known as the PRISM surveillance program. PRISM is the number one source of raw intelligence for the NSA analytic reports and accounts for 91% of the NSA's internet traffic. This program initially began in 2007 through the Protect America Act under the Bush Administration. Here is just one of many slides leaked by Snowden detailing the PRISM program. The caption of the slide reads:

¹ The Five Eyes intelligence alliance include Australia, Canada, New Zealand, the United Kingdom, and the United States.

PROVIDERS AND DATA: The PRISM program collects a wide range of data from the nine companies, although the details vary by provider.

TOP SECRET//SI//ORCON//NOFORN

Gmail facebook Hotmail YAHOO! Google skype paltalk.com YouTube AOL mail

SPECIAL SOURCE OPERATIONS (TS//SI//NF) **PRISM Collection Details** **PRISM**

Current Providers

What Will You Receive in Collection (Surveillance and Stored Comms)? It varies by provider. In general:

- Microsoft (Hotmail, etc.)
- Google
- Yahoo!
- Facebook
- PalTalk
- YouTube
- Skype
- AOL
- Apple

- E-mail
- Chat – video, voice
- Videos
- Photos
- Stored data
- VoIP
- File transfers
- Video Conferencing
- Notifications of target activity – logins, etc.
- Online Social Networking details
- **Special Requests**

Complete list and details on PRISM web page:
Go PRISMFAA


TOP SECRET//SI//ORCON//NOFORN

I am very certain that most people use either Microsoft, Google, Facebook, YouTube, Skype, or Apple multiple times throughout the regular day. Then carefully read the gray filled box and see just how many items that the PRISM program naturally requests. This does not even include the “**Special Requests**” field which is not touched on. From this we can extrapolate that *every* online communication that we send is being recorded. Every instance of our lives: our emotions, our opinions, arguments, intimacy, *everything*, is being monitored and recorded.

There is no way they could ever look through all of this information though, right? If PRISM is recording everyone's online information then it must be impossible to pinpoint certain things? Wrong. There is a program used to query through all of the NSA's database. This program is called XKEYSCORE. What is this program capable of? Take a look.

TOP SECRET//COMINT//REL TO USA, AUS, CAN, GBR, NZL

What is XKEYSCORE?



1. DNI Exploitation System/Analytic Framework
2. Performs strong (e.g. email) and soft (content) selection
3. Provides real-time target activity (tipping)
4. "Rolling Buffer" of ~3 days of ALL unfiltered data seen by XKEYSCORE:
 - Stores full-take data at the collection site – indexed by meta-data
 - Provides a series of viewers for common data types

1. Federated Query system – one query scans all sites
 - Performing full-take allows analysts to find targets that were previously unknown by mining the meta-data

TOP SECRET//COMINT//REL TO USA, AUS, CAN, GBR, NZL

In just around three days all of the unfiltered data can be searched for certain keywords, peoples, images, etc. Say you just happen to look up the latest news on terrorist attacks through Google. All that has to happen is for one NSA agent to XKEYSCORE the term "terror" or "terrorism" and your name, IP address, physical address, online banking information, emails, text messages, pictures sent, and *everything else* that you have EVER looked up or done online is available to said agent. I am not sure about you, but I do not want every interaction that I have ever had recorded. And if you're a Verizon customer... well then give this leaked document a quick read through:

TOP SECRET//SI//NOFORN

UNITED STATES
FOREIGN INTELLIGENCE SURVEILLANCE COURT
WASHINGTON, D.C.

IN RE APPLICATION OF THE
FEDERAL BUREAU OF INVESTIGATION
FOR AN ORDER REQUIRING THE
PRODUCTION OF TANGIBLE THINGS
FROM VERIZON BUSINESS NETWORK SERVICES,
INC. ON BEHALF OF MCI COMMUNICATION
SERVICES, INC. D/B/A VERIZON
BUSINESS SERVICES.

Docket Number: BR

13 - 8 0

SECONDARY ORDER

This Court having found that the Application of the Federal Bureau of Investigation (FBI) for an Order requiring the production of tangible things from **Verizon Business Network Services, Inc. on behalf of MCI Communication Services Inc., d/b/a Verizon Business Services (individually and collectively "Verizon")** satisfies the requirements of 50 U.S.C. § 1861,

IT IS HEREBY ORDERED that, the Custodian of Records shall produce to the National Security Agency (NSA) upon service of this Order, and continue production

TOP SECRET//SI//NOFORN

Derived from: Pleadings in the above-captioned docket
Declassify on: 12 April 2038

TOP SECRET//SI//NOFORN

on an ongoing daily basis thereafter for the duration of this Order, unless otherwise ordered by the Court, an electronic copy of the following tangible things: all call detail records or "telephony metadata" created by Verizon for communications (i) between the United States and abroad; or (ii) wholly within the United States, including local telephone calls. This Order does not require Verizon to produce telephony metadata for communications wholly originating and terminating in foreign countries.

Telephony metadata includes comprehensive communications routing information, including but not limited to session identifying information (*e.g.*, originating and terminating telephone number, International Mobile Subscriber Identity (IMSI) number, International Mobile station Equipment Identity (IMEI) number, etc.), trunk identifier, telephone calling card numbers, and time and duration of call. Telephony metadata does not include the substantive content of any communication, as defined by 18 U.S.C. § 2510(8), or the name, address, or financial information of a subscriber or customer.

IT IS FURTHER ORDERED that no person shall disclose to any other person that the FBI or NSA has sought or obtained tangible things under this Order, other than to: (a) those persons to whom disclosure is necessary to comply with such Order; (b) an attorney to obtain legal advice or assistance with respect to the production of things in response to the Order; or (c) other persons as permitted by the Director of the FBI or the Director's designee. A person to whom disclosure is made pursuant to (a), (b), or (c)

TOP SECRET//SI//NOFORN

TOP SECRET//SI//NOFORN

shall be subject to the nondisclosure requirements applicable to a person to whom an Order is directed in the same manner as such person. Anyone who discloses to a person described in (a), (b), or (c) that the FBI or NSA has sought or obtained tangible things pursuant to this Order shall notify such person of the nondisclosure requirements of this Order. At the request of the Director of the FBI or the designee of the Director, any person making or intending to make a disclosure under (a) or (c) above shall identify to the Director or such designee the person to whom such disclosure will be made or to whom such disclosure was made prior to the request.

IT IS FURTHER ORDERED that service of this Order shall be by a method agreed upon by the Custodian of Records of Verizon and the FBI, and if no agreement is reached, service shall be personal.

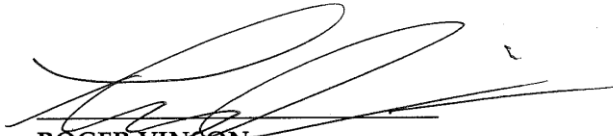
-- Remainder of page intentionally left blank. --

TOP SECRET//SI//NOFORN

TOP SECRET//SI//NOFORN

This authorization requiring the production of certain call detail records or "telephony metadata" created by Verizon expires on the 19th day of July, 2013, at 5:00 p.m., Eastern Time.

Signed 04-25-2013 P02:26 Eastern Time
Date Time



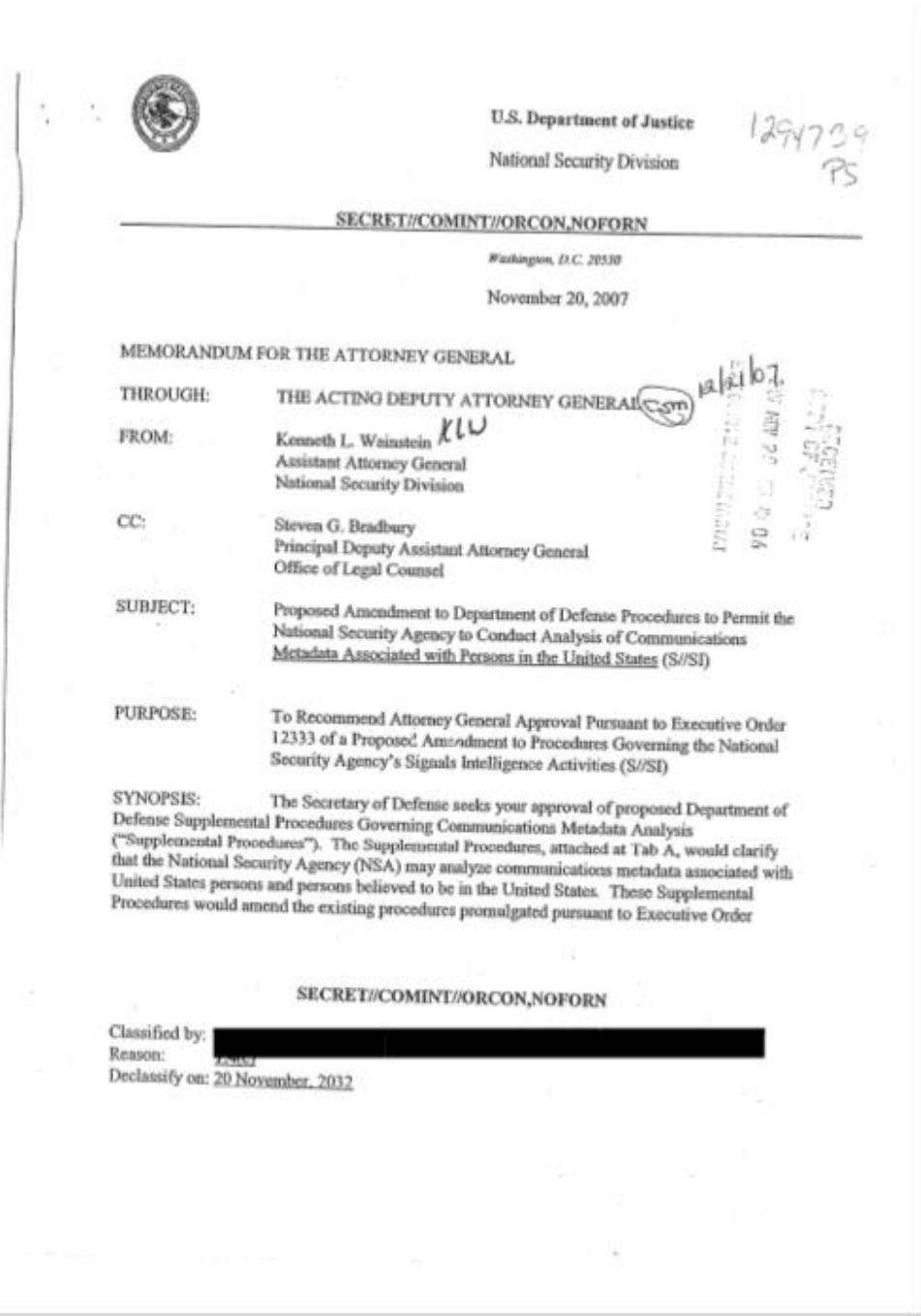
ROGER VINSON
Judge, United States Foreign
Intelligence Surveillance Court

I, Beverly C. Queen, Chief Deputy Clerk, FISC, certify that this document is a true and correct copy of the original. *BQ*

TOP SECRET//SI//NOFORN

This document states that “Telephony metadata does not include the substantive content of any communication, as defined by 18 U.S.C. § 2510(8), or the name, address, or financial information of a subscriber or customer.” However, I am not one to trust the government or big businesses enough, especially a system as big as the NSA, to not just go ahead and gather all of the information that they can.

Then there is this document, which I will only show one slide of for now, which is a proposal to broaden the powers for the NSA to collect data.



The statement that is the most worrisome is, “the NSA may analyze communications metadata associated with United States persons and persons believed to be in the United States.”

In a recent article, published by The Intercept (November 30, 2015), it is stated that “The USA FREEDOM Act, signed into law on June 2 earlier this year, gave the executive branch 180 days to

wind down the bulk collection program. According to the Tumblr of the Office of the Director of National Intelligence, the government is “prohibited from collecting telephone metadata records in bulk” starting November 29. The executive branch will now be able to obtain phone metadata by asking the U.S. Foreign Intelligence Surveillance Court to order telecommunications companies to turn over specific records.” This is no doubt a huge win for anti-NSA/Mass Surveillance activists. With that stated, I am not one to easily trust the powers that are in charge. If they (the US government/ the NSA) can and choose to directly spy on foreign governments and peoples illegally, then what makes you think that they will not continue to conduct mass surveillance of their own people?

Metadata

All of this information might be new to you though, and therefore you may not fully understand the terminology and the scope of it. Probably the most important and fundamental term that you will need to understand is metadata. Most simply explained, metadata is everything about a piece of information, apart from the information itself. So if someone or some organization is collecting metadata then they can easily find out that you called a specific number from a specific location for X amount of minutes. They directly have access to the content of the call, but that can easily be bypassed even though the legality of it is in question. Truthfully though, the legality of the US government or any of its large organizations does not matter much anymore. The governments of the United States, United Kingdom, and others have put themselves above standards of legality. You might not think that the collection of metadata is a big deal. They can just directly see who I contacted but not the information? That does not seem so bad. Unless you are calling an HIV specialist/doctor, a sex hotline, a suicide prevention hotline, or anyone else that you may not want people knowing about. Metadata is not just applied to your phones. Your credit card/debit card purchases, locations, emails, attachments, and just about everything else you do with your life is metadata.

I am sure most of you reading this carry a smartphone with you almost everywhere you go. That smartphone is constantly tracking your location whether it be from your GPS being turned on or from different radio towers pinging signals to it. Someone collecting your data would know when you are home, when you are at work, where you go in between, if you went to a certain store, how long you were at these places, etc. The best part for the people surveilling you is that you are doing all of the work for them! Smartphones are a mass surveillance dream. There are other people tracking and collecting your phone data other than the NSA. If you have the Facebook application installed on your phone then your location is always being monitored, even when you turn off the features which allow it to do so. Facebook also has access to your microphone, camera, images in your phones gallery, etc. Edward Snowden, in an interview with Brian Williams, even stated that it was possible for the government to turn your phone on when it was off, or even to be able to prevent it from being turned off completely. From this they would be able to constantly listen in to your conversations by using your phones microphone or even being able to use the camera on it. While Snowden does state that intrusions like these happen to those who are specifically targeted, I would rather be safe than sorry in a situation such as this.

Ultimately, the mass collection of metadata is an extreme intrusion of privacy and can also land people in quite a bit of trouble. Without looking at the direct information, many situations can be blown out of proportion or be seen in the wrong light. Activities that *might* appear suspicious to an NSA agent or to a program they use are probably not suspicious at all, rather just regular searches and messages that are misinterpreted. However, this could easily land you on a watch list and your information can and will be monitored closer than before. This is why metadata should matter to you.

Man in the Middle (MitM)

A MitM attack is a simple concept to understand. Say you email your friend through unencrypted channels and with no encryption in the email itself. If someone were to be monitoring you, or if they are just snooping in on your unencrypted Wi-Fi connection, they would be able to intercept that email without you or your friend ever knowing. The best way of preventing such attacks is to keep your information encrypted and secure while also sending this information through a secure channel. I will talk more about how to keep your information secure in later sections.

Your Rights to Privacy

If you live in the United States then you are (or at least should be) familiar with your Amendment Rights. If not, I have included them here:

Amendment I

Congress shall make no law respecting an establishment of religion, or prohibiting the free exercise thereof; or abridging the freedom of speech, or of the press; or the right of the people peaceably to assemble, and to petition the Government for a redress of grievances.

Amendment II

A well regulated militia being necessary to the security of a free state, the right of the people to keep and bear arms shall not be infringed.

Amendment III

No Soldier shall, in time of peace be quartered in any house, without the consent of the Owner, nor in time of war, but in a manner to be prescribed by law.

Amendment IV

The right of the people to be secure in their persons, houses, papers, and effects, against unreasonable searches and seizures, shall not be violated, and no Warrants shall issue, but upon probable cause, supported by Oath or affirmation, and particularly describing the place to be searched, and the persons or things to be seized.

Amendment V

No person shall be held to answer for any capital, or otherwise infamous crime, unless on a presentment or indictment of a Grand Jury, except in cases arising in the land or naval forces, or in the Militia, when in actual service in time of War or public danger; nor shall any person be subject for the same offense to be twice put in jeopardy of life or limb; nor shall be compelled in any criminal case to be a witness against himself, nor be deprived of life, liberty, or property, without due process of law; nor shall private property be taken for public use, without just **compensation**.

Amendment VI

In all criminal prosecutions, the accused shall enjoy the right to a speedy and public trial, by an impartial jury of the State and district where in the crime shall have been committed, which district shall have been previously ascertained by law, and to be informed of the nature and cause of the accusation; to be confronted with the witnesses against him; to have compulsory process for obtaining witnesses in his favor, and to have the Assistance of Counsel for his defense.

Amendment VII

In suits at common law, where the value in controversy shall exceed twenty dollars, the right of trial by jury shall be preserved, and no fact tried by a jury, shall be otherwise reexamined in any court of the United States, than according to the rules of the common law.

Amendment VIII

Excessive bail shall not be required, nor excessive fines imposed, nor cruel and unusual punishments inflicted.

Amendment IX

The enumeration in the Constitution, of certain rights, shall not be construed to deny or disparage others retained by the people.

Amendment X

The powers not delegated to the United States by the Constitution, nor prohibited by it to the states, are reserved to the states respectively, or to the people.

The Amendments here we will focus on are the First and Fourth Amendments. These have been hot topics in the media lately in terms of mass surveillance and government overreach. If you are not familiar with them then I suggest you take a closer read to them. (It's like two sentences, just go ahead and read them again.)

The First Amendment is strictly on the right of free speech. Free speech encompasses a persons right to say whatever they want, for the press to report on whatever they want, and the right to peaceful protests. However, if you have been paying any attention to the news lately you would know that all three of these aspects to free speech are being torn away from us. Now instead of having the right to say or report on anything you want, you can only say or report anything a long as it does **not scrutinize** the powers that be. These powers can be anything from a telecommunications company (AT&T for example) to the United States Government and the organizations it controls. The act of scrutiny is what is key here. If you break through the false veil of "power" and "superiority" that such organizations hold then they are not seen as strong. Rather, they have been embarrassed and now feel exposed and weak. If this does happen then be prepared to have people come after you. Ask any whistleblower or reporter who has come up with this information in the past 20 years. One person who does come to mind is Weev. I will not get into the backstory of who Weev is but rather what he did. Weev was on his iPad one day and went to log into his AT&T account. In the public domain web-address, something along the lines of www.att.com/login/query1547 (randomly thought up URL), Weev decided to merely add one to the end of the URL. So instead of www.att.com/login/query1547 he typed in www.att.com/login/query1548. What this ended up doing is revealing the email address of the person who last logged-in. The email would just show up in the "Please enter your email" field of the log-in

process. Weev noticed this as a major security flaw and proceeded to collect all of the emails of the people who logged in. He then sent the list, which contained email addresses from everyday citizens to those in the military and the White House, to a journalist who proceeded to publish a story about the security hole in AT&T's web servers. Instead of rewarding Weev with the find, or even thanking him, they decided to file criminal charges on him. Weev wound up spending 18 months in prison over addition as well as being labeled a "malicious hacker." I have provided information about his story at the end of the document in the **Sources** section.

So why did AT&T decide to punish Weev instead of thanking him? Because if they came out and acknowledged that such a security flaw existed it would undermine their "power" and "superiority." In reality, while usually having a great but of money and overall capital, such large corporations and organizations are not that strong. They throw up this disguise of being larger than life but it truly is all a facade. (Funny enough, one movie that accurately portrays the underbelly of such corporations is the 1995 classic, Hackers. If you have not seen this movie then I would highly recommend it just for the nostalgia and hilarious moments that were created around 90's cyber culture.) AT&T was able to circumvent any real scrutiny from the main stream media (MSM) by labeling Weev as a "hacker" as many people are often afraid of these stigmatic buzzwords. The same exact thing happened with Edward Snowden. Instead of actually explaining the programs and organizations in question, not to mention how Snowden was given free reign and access to all of these documents, the United States labeled him as a traitor which is was the majority of the MSM broadcasted after the first intial days of his leaks. And even now, in the wake of the Paris Attacks (11/13/2015) we see various governments blaming encryption software and Edward Snowden instead of accepting that they were unable to prevent the attacks. Mind you that the people who executed the Paris Attacks were using unencrypted communications anyways.

This conversation of exposing people or governments in power is not a new argument. If you wish to look up more on the relationship between those in power and those who are not then I have provided a reading list in the **Sources** section. It is an interesting and intriguing read if you wish to truly understand the relationship you have to your government or to other people who are in power over you.

On top of all of this, we are apparently not allowed to protest anymore when we are challenging the powers that be. There are numerous protests that have been broken up (many times in violent ways) in recent years. Some that come to mind are Occupy Wall Street and the most recent protest in Paris over the meetings on climate change. Many times these peaceful protesters are being arrested also.

Now comes the Fourth Amendment. As I have previously shown, the NSA, GCHQ, "Five Eyes", and others have all been intercepting our data, looking up information on our data through XKEYSCORE, and then storing that data for however long they choose to do so. Not even to mention the ability to listen in through the microphone on your smartphone or laptop, and even the camera on them as well. These countries have been in direct violation of our Fourth Amendment since 2001, most likely even prior to that. Obtaining all of this information without going through the proper channels to issue search warrants is strictly illegal. Yet, these governments and organizations will face no true backlash because they have been able to ascend themselves above continental laws. The only way to ensure that your Fourth Amendment Right is being kept true is to encrypt your data and stay safe, which is pretty fucked up that you have to actively fight and use alternative routes in order to keep your rights.

Why not Windows –

<http://www.getgnulinux.org/en/windows/>

Restrictions

A legal copy of Windows is expensive, but what do you get? Windows and Office are licensed, not sold.

By using these products, we have to agree to **a number of harsh restrictions**. For most Windows licenses, you can't keep the software when you change the hardware. You sometimes can't even give your software away. Who can run the software? On which computer? What can you do with it? The list of restrictions is long and some items are outrageous.

The small print in the contract

- Windows and Office are licensed, not sold.

No one can buy Windows or Microsoft Office: instead users **purchase a permission to use** them. The license describes the terms of this permission. It is the restrictive legal text you have to click "OK" to upon install.

- You must abandon many rights to use the software.

There are a number of restrictions that you must accept by law. Restrictions on who can use the software, what kind of revenue you may earn with it, on how you choose to install it, restrictions on your privacy, even on whether you can give it away: the list is long. [Reading the license and enumerating your remaining rights](#) is itself a difficult task.

- An OEM (Original Equipment Manufacturer)-distributed software cannot be transferred to another computer.

If you bought your computer with Windows or Office pre-installed (the so-called OEM licenses, or the "shrink wrap" Windows discs), **if you change computers you must buy software again**. The license is linked to one computer, and expires when the computer dies. It is then illegal to transfer the software on another computer.

If you go to most of the free software sites you can click down a couple of levels and find the [GNU] ,GPL (General Public License) the X license, the Apache license, whichever terms and conditions you have to accept in order to use that software.

Now with a proprietary software company, the license is buried so you can't read it until after you have paid for the product, then they're asking you to **turn off part of your brain**, they're asking you to turn off part of your ability to work with other **people and to do business, when you use their software**.

Don Marti, [2005 interview](#)

The meaning behind

Companies like Microsoft like to assimilate their software to physical products, [when mentioning copyright infringement](#) for example. Yet, proprietary software is very different because of the restrictive license -such restrictions would be unthinkable on a car or bicycle, for example.

Restrictions on the use of Office and Windows are so harsh, that many violations occur everyday around us. People are tempted to buy only one version of Microsoft Office and install it on two computers. Others keep their version of Windows when they throw their PC away. Other people give away their second-hand Windows software when they stop using it.

You have an alternative to breaking this law, or feeling very restrained when you abide to it. GNU/Linux is [Free Software](#) (much better than just freeware): its [GPL](#) license is designed to protect your rights.

More details on restrictions

- It is unclear who can use, receive or buy your software.

The license is particularly unclear as to who may or may not use your version of Windows or Office. Several sentences in the Microsoft Office license suggest it would be illegal to let your neighbour type a letter with your version of Word on your computer.

It is however clear in the Microsoft Windows license that you may *only* give or sell your copy of Windows software to anyone **if you are the first buyer**. This means, that if you buy it from the user who initially purchased it from Microsoft, then *you* are not able to sell or give the software away to yet another user, even if you do not use it anymore, even if you buy the latest software version with your new computer.

- You may not lease, lend or provide commercial hosting services with the software

You cannot let professional users use your software, whether they pay for the service or not, and whether you use a "Professional" license or not.

- The upgrade is only valid for the first license you use it on

If that original license expires (for example because the computer it came on stops working), so does the upgrade. If you purchase an upgrade, you are not allowed to use the original software version anymore.

- Educational versions are crippled

If Microsoft accepts to define you as a student or an academic, you are allowed to buy an academic license, and install the software on three computers. But it is illegal to use it

for any commercial purpose "*or in any way related to the operation of any business enterprise or revenue-generating activities*".

- The components of the software may not be separated

It is illegal to buy Microsoft Office, then install only Word on one computer, and only Excel on another. The Office suite is one single product.

- Private information is collected

The license explicitly states that "Microsoft and its affiliates" collect technical information gathered on your computer by the software. They "*may use this information solely to improve products or to provide customized services or technologies*".

In order to activate most Microsoft products, as part of the license agreement, your computer connects to the Microsoft servers and sends "technical" information. It can "solely" be used for pretty much anything.

Poor support for open standards

There exist open file formats and protocols, which are standards to store and carry many types of information, respectively. They are openly specified: they can be read by anyone on any computer, now and in the future. Proprietary programs, however, rarely support them.

For a long time Microsoft Office only had support for Microsoft formats. There are many ways to write work documents, but Microsoft Office users were bound to Microsoft Office files. People who opted without expensive Microsoft products cannot write and read .doc files very reliably.

There are of course other ways of writing and exchanging office files, the most well-known being [the OpenDocument Format](#). But Microsoft isn't keen on letting Office users exchange files anyone can read and edit. As of Microsoft Office 2007 SP2, Microsoft finally added support for the ODF format after being pushed by the European Commission.

Standards that change all the time

Sticking to Microsoft standards is not an easy job – unless you can afford to upgrade very often. Ever tried to work on the same .doc file with both an Office 95 and an Office 97 computer? You'll know what we mean.

Microsoft owns the Office file formats: **they change them with every new Office version** and have no obligation to keep them backwards-compatible. Saved your presentation as a .ppt file? If you give up using Microsoft Office next year, you'll **have to rely on other communities to reverse-engineer the format, to be able to access and modify your own work.**

Default programs you can't uninstall

Don't want Windows Media Player on your computer? Don't use Internet Explorer anymore? **You can't uninstall these programs.** They previously worked on a standalone basis, but have been intrinsically linked with Windows – so they come in with every PC and no one can get rid of them.

Monopolistic practices

Microsoft has a tight control over OEMs (computer manufacturers) who ship their computers with Windows installed.

This means that no program competing with Microsoft products, be it [multimedia player](#), [web browser](#), [office suite](#), [instant messaging program](#) or other, will come pre-installed on most computers you can buy.

It is a good thing to propose a wide range of software and services like Microsoft do; however, designing and combining them to shut users from non-Microsoft peers is unethical. It's not technically hard to adopt more open formats. But it means your customers are *free to choose what they do with their work* – Microsoft isn't there yet.

What about source code?

The source code details the way a program works (it is effectively what programmers write). Without it no one can understand how the software is built. It does not matter if you can't read code: whether or not it is available **directly affects any user.**

No one can look inside

Windows comes without its source code. More than that: all users must abide to the license term that says:

You may not reverse engineer, decompile, or disassemble the software.

As such, it is illegal to work out how Windows or Microsoft Word are designed. It is even illegal to try. You are also forbidden to modify the program for any purpose.

This restriction in the license makes sure that **Microsoft remain the sole organization that understand how their products work.** Windows is very much like a car that only the original manufacturer is allowed to service.

You might say, "How do I change this recipe to take out the salt?" and the great chef would respond, "How dare you insult my recipe, the child of my brain and my palate, by trying to tamper with it? You don't have the judgment to change my recipe and make it work right!"

Richard Stallman, [Why Software Should Be Free](#)

Insecurity means business

Whenever a flaw in Windows is discovered, it is exploited, which results in trojans, viruses, spyware and the likes. **Such nuisances delight the Windows security industry**, including Microsoft, which develops various anti-virus protections with subscriptions.

It is a lucrative process to release flawed software *that no one is allowed to change*, and then selling protection services over it (anti-virus software doesn't correct flaws: it merely prevents viruses from exploiting them, if it is active and updated).

Microsoft will today happily [sell you their own protection over their own insecurities](#) – it sells the poisoned apple and its antidote, separately.

You can trust free software

It comes as no surprise that proprietary software is severely lacking in terms of security, compared to software whose source code is freely available (including GNU/Linux).

Free software means **programmers can change the code** to repair flaws. It means you can hire someone to really check how secure your software is. It means you can benefit from the contributions of a world-wide community to improve safety and reliability. Fifteen million users run GNU/Linux without an anti-virus, in complete safety. And the servers behind search engines and banks run on it too.

Source code is the recipe for software. How could one improve on a meal with too much salt, if forbidden to look at the recipe used to cook it?

Using Windows and Office requires us to **not ask** nor search for the source of the software. Only Microsoft developers can modify your program.

Which would you rather trust: the package you are forbidden to study, or the package with the recipe?

Professor Norm Matloff's Beginner's Guide to Installing and Using Linux

Norm Matloff
Department of Computer Science
University of California at Davis
matloff@cs.ucdavis.edu
c 1999-2013
January 4, 2013

1 Background Needed

I have tailored the material here to beginners. No special sophistication in computers is needed. Any typical Microsoft Windows user should be able to understand the instructions here and install Linux in less than an hour's time. (Do not be intimidated by the length of this document; you probably will not have to use most of it.) Don't worry about the length of this document. You'll probably only need a small part of it. For some background on the history and significance of the Linux operating system, go to Appendix A.

2 Install to Where?

I recommend that you install Linux on your hard drive, so that you will dual-boot either your old OS (I'll assume Windows from now on) or Linux. After installation, each time you boot up, you will be given a choice of whether to boot Windows or Linux. Another alternative is to install Linux on a USB key (memory stick) or external hard drive. Still another choice is to run Linux as a virtual machine. This is not covered in this document, but if you are a UCD student, see <http://csifdocs.cs.ucdavis.edu/documentation/archives/csif-fedora-on-a-for> the easy steps. By the way, if you have a Mac, you may have the capabilities you need without Linux, since both the Mac OS and Linux are Unix systems.

3 Which Linux Distribution Is Best?

Linux comes in various distributions, called distros by Linux aficionados—but they are all Linux in terms of functionality. Some of the most popular are Ubuntu, Red Hat, Fedora, Linux Mint, SuSE, MEPIS, PCLinuxOS and so on. Remember, there are tons of good distros out there. Any of the above would be fine, as would many others, but here is my short answer: Use Ubuntu (or one of the many Ubuntu derivatives, such as Linux Mint). It is arguably one of the most user-friendly of the distros, and it has a large user community you can access in the Ubuntu forum on the Web, probably the most active one out there. I now use Ubuntu myself on my home computers, as well as on my office computer, after years of using various other distros.

If you have an old machine, especially one with limited memory (i.e. RAM), you may wish to give Puppy Linux or Damn Small Linux a try. I installed them (one at a time) on an old 1998 laptop with only 64M of memory! And they take as little as 50M of disk space.

4 Installation

Here is the short way to install Linux on your hard drive, dual-booting with Windows.

4.1 The Short Answer

Here you will install the Ubuntu distro, using UNetbootin as your installation tool.

For simplicity, I'll assume you wish to install Linux to your laptop.

1. Download UNetbootin from its home page <http://unetbootin.sourceforge.net/> to your hard drive. (For further informat on UNetbootin, see <http://sourceforge.net/apps/trac/unetbootin/wiki/guide>.)
2. Insert a USB key (memory stick). It needs to have FAT32 format. It probably came that way, but if not then check the Web for how to fix that using your OS.
3. Run UNetbootin.
4. Click Select Distribution, and choose Ubuntu. Then choose the latest Live version in the window to the right.
5. For Type, choose USB Drive, and for Drive, choose the drive in which your USB key is inserted.
6. Click OK.
7. After the installation to your USB key finishes, choose Exit.
8. Leaving your key in the drive, restart your laptop.
9. Select the choice labeled something like Try Ubuntu.
10. Once Ubuntu boots up, try to use the WiFi: Click on the proper icon at the top right of the screen, and select your wireless network. If none appears, then for now, connect your machine to an Ethernet jack, say at a public library or copy shop.
11. Follow directions. If asked whether you want third-party software to be installed, say yes.
12. Reboot (remove the USB key when the screen goes dark).
13. If WiFi didn't work above, it should work now. You may have to click a pop-up window that asks If you want to use the proprietary drivers.

During the installation process, there may be some mention of disk partitions. You should not have to take action, but if you wish to know about partitions (very useful!), see Appendix B.

4.2 Installing Linux to a USB Key or External Hard Drive

You can install Linux to a USB key or external hard drive, and boot up Linux from there whenever you want to use Linux. (This is not the same as the USB key created from UNetbootin, which is only temporary.)

Unfortunately, UNetbootin does not produce Mac-bootable USB keys.

4.2.1 Installation Method I (for Slax Linux)

Slax is a nice, colorful and small version of Linux, at <http://www.slax.org>. Click on "Get Slax" to download, and on "Read Manuals" to see how to install onto a USB key or external hard drive. It is extremely easy!

In short:

1. Download the Slax .tar package.
2. Go to the directory (or folder, in Windows) for your USB key.
3. Unpack the .tar file from that directory.
4. Go to the boot subdirectory, and run either bootinst.sh (from Linux) or bootinst.bat (from Windows).

In the Linux case, you may need to precede your command by sudo.

4.2.2 Other Methods

You can use UNetbootin (Section 4.1), but you'll need to make your USB installation persistent; see <http://sourceforge.net/apps/trac/unetbootin/wiki/guide>. There are methods to construct your USB installation "by hand" from an ISO file. This is complicated, and will not be pursued here.

5 Post-Installation Configuration

This section describes some further steps I recommend taking after your installation is finished.

5.1 Configuring Your Search Path (“Why can’t I run my a.out?”)

Most Linux distros do not include your current directory, ‘.’, in the PATH variable. Thus if for example you compile a program and then type

```
a.out
```

the shell may tell you that a.out is not found. You are expected to explicitly specify the current directory: ./a.out

If you consider this a problem, as I do, to remedy it in the case of the BASH shell (the default shell for most distros), edit the file /.bash profile In the line which sets PATH, append “:.” (a colon and a dot) at the end of the line, with no intervening spaces. Then log out and log in again, or do source ~/.bash_profile

5.2 Configuring a Printer

Your Linux distribution should have some program to help you configure your printer if something went wrong during installation. For example, if you are running the GNOME GUI, select System Administration j Printing.

It’s now easy to connect to a remote printer elsewhere on your network (even if it is on a Windows machine), using Samba.

5.3 Switching from GNOME/Ubuntu Unity

I personally don’t like the Unity window manager in GNOME. Many others feel the same way. So, Ubuntu gives us other choices. To set them up, do

```
sudo apt-get install gnome-shell
```

At your next start, the login screen will show a symbol next to choices of login names; choose Gnome Classic (No Effects) or whatever you like; experiment to find one that suits you.

5.4 Configuring KDE/GNOME for Convenient Window Operations

5.4.1 Autoraise Etc.

You should find that windowing operations are generally easier in Linux systems than in Windows, in the sense of requiring fewer mouse clicks, if you set things up that way. Personally, I find it annoying in Windows that, when I switch from one window to another, I need to click on that second window. In most Linux windowing systems, I can arrange things so that all I have to do is simply move the mouse to the second window, without clicking on it. The term for this focus follows mouse, and we can configure most Linux windowing systems to do this.

Also when I move from one window to another, I want the second one to “come out of hiding” and be fully exposed on the screen. This is called autoraise, and can be configured too.

You can arrange this configuration in less than one minute’s time. Again, the exact configuration steps will vary from GNOME to KDE, and from one version to another within those systems, so I can’t give you the general steps here but here is how it works on GNOME in Ubuntu 12.10 or later:

Open a terminal window (ctrl-alt-t), and type

```
sudo gsettings set org.gnome.desktop.wm.preferences auto-raise true
```

```
sudo gsettings set org.gnome.desktop.wm.preferences focus-mode 'mouse'
```

Then log out and back in. You only need do this once.

You get check these settings using `get` instead of `set`, or use `reset` to revert to the original values (`false` and `'click'`), e.g.

```
sudo gsettings reset org.gnome.desktop.wm.preferences auto-raise
```

6 Some Points on Linux Usage

6.0.2 Ubuntu Root Operations

Ubuntu works like any other Linux distro, except for one important point: Ubuntu does not have a root user account in the classic Unix sense. Instead, whenever executing a command which requires root privileges, one precedes the command by the term `sudo` (“superuser do”). One is then prompted for a password, which is the password for the first user account created at the time of installation. If you have a lot of root-type work to do in a session, type

```
$ sudo -s
```

to create a new superuser shell, and do your work there.

6.1 More on Shells/Terminal Windows

In Microsoft Windows, most work done by most users is through a Graphical User Interface (GUI), rather than in a command window (Start | Run | cmd). In Linux, a lot of work is done via GUIs but also it is frequently handier to use a command window, called a terminal window. You should always keep two or three terminal windows on your screen for various tasks that might arise. You can start a terminal window in GNOME by typing `ctrl-alt-t`.

When you type commands in a terminal window, the program which reads and acts on those commands is called a shell. (Thus a terminal window is sometimes called a “shell window.”)

I have an introduction to Unix shells, at <http://heather.cs.ucdavis.edu/~matloff/UnixAndC/Unix/ShellIntro.html> and <http://heather.cs.ucdavis.edu/~matloff/UnixAndC/Unix/CShellIII.html>. These are based on the T C-shell, `tcsh`, but at least in the case of the first tutorials, most of the material also applies to the more popular `bash` shell.

6.2 Cut-and-Paste Window Operations

To do a cut-and-paste operations, hold down the left mouse button and drag it to highlight the text you wish to copy. Then go to the place you wish to copy that text, and simultaneously push both the left and right buttons. Generally, more things are cut-and-pastable in Linux than Windows, so this is a big convenience.

6.3 Mounting Other Peripheral Devices

This section explains how to use DVDs, USB devices and so on under Linux. You may wish to review Section B before continuing.

6.3.1 Mount Points

Each I/O device that contains a file system must be mounted, i.e. associated with some directory. That directory is called a mount point. The files then appear in that directory.

These days most Linux distributions have a designated directory for mount points for DVD/CD-ROMs, USB devices, floppy disks, etc. This will vary from one distribution to another, but typical directory names are `/mnt`, `/media` etc. You can check what is currently mounted by running the `df` command from

a shell window (another good Linux learning experience). The mount points are listed along with the /dev files. Also, to list the /dev files for all your operating drives including USB flash drives and including drives not mounted, type

```
sudo fdisk
```

- For more detailed information, such as file system types, just run mount without any arguments. Your machine's internal hard drives, and possibly other devices, will be mounted automatically at boot time. This is controlled by the entries in the file /etc/fstab. The details are an advanced topic, but even without understanding everything, you might find it worthwhile to take a quick look at that file. Here is a line from the file on my office machine,

```
/dev/sda3 /usr/home ext3 defaults 0 2
```

Here /dev/sda3 is the third partition ('3') on my first SATA hard drive ('a'). The entry says that this partition has an ext3 type filesystem in it, and is to be mounted at the directory /usr/home. The remaining entries concern things such as backup and file system checks.

When you attach a device to your machine after bootup, your system will probably recognize it immediately, and maybe pop up a window showing the device's contents. If you have trouble, you can use the Unix mount command. This is an advanced command, but just to give you an idea, a typical usage would be

```
mount -t iso9660 /dev/hdc /mnt/yyy
```

This tells Linux that the I/O device corresponding to /dev/hdc, our CD-ROM, should be mounted at the directory /mnt/yyy. If that directory doesn't exist, you must create it first, using mkdir. The field -t iso9660 says that the file system type is ISO9660. This is standard for CD-ROMs, and you can probably omit it. Use umount to unmount. It's not safe to remove a USB device without running this first.

6.3.2 Using USB Devices

USB drives, including memory sticks, should have their filesystems mounted automatically when you attach them. Use the df command to check where they've been mounted (it could be in the directory /mnt/ /media etc.). USB mice should become automatically usable when you attach them.

²This might not work in some cases. If fdisk doesn't recognize your device, try viewing the file /proc/partitions. Your device may appear there, say as sdb1. Then run mount as shown below, on /dev/sdb1.

7 Linux Applications Software

7.1 GUI Vs. Text-Based

Most people prefer to use GUI-based applications. If you are one of them, rest assured that there are tons of them available for Linux.

I do wish to mention, though, that many "super hard core" Linux users prefer to use text-based applications, rather than GUI ones. For instance, I and many others like the mutt e-mail utility (Section ??), which is text-based. Here's why, at least in my view:

- I often access my Linux machine remotely, while traveling. I might be at a university library, for instance, or at the business center in a hotel, and be "stuck" with a Windows machine, and logging in to my Linux machine via an SSH connection.⁴ This limits me to text.

- It's very important to me that I use the same text editor for all my computer applications—e-mail, programming, word processing, etc.—so that I can take advantage of all the abbreviations, shortcuts and so on which I have built up over the years. This saves me huge amounts of typing. But most GUI applications, e.g. e-mail utilities, have their own built-in text editors, so I can't use mine.
- I find that text-based applications often have more features, are better documented, etc. For example, I often wish to automate certain processes, such as uploading files to another machine, and typically text-based programs do this better. However, in listing my favorite applications in Section 7.2 below, I've made sure to list both text-based and GUI programs.

7.2 My Favorite Unix/Linux Apps

In Ubuntu, one downloads new apps using apt-get, which I'll use in my examples here. The same is true for other distros derived from Debian. In Fedora, use yum.

7.2.1 Text Editing

I use a modern extension to the vi editor, vim. This is the version of vi which is built in to most Linux distros. See my tutorial at <http://heather.cs.ucdavis.edu/~matloff/vim.html>.

Note: In some Fedora distros, somehow the version of vim that is linked to vi isn't configured fully correctly. I suggest using /usr/bin/vim directly. Even though vim is text-based, it does have a GUI version too, gvim. This comes with nice icons, allows you to do mouse operations, etc. Unfortunately, most Linux distros seem to have only the text-based program. To get the GUI, you can download it yourself. In Ubuntu, do

```
sudo apt-get install vim-gnome
```

For this, you may need to edit /etc/apt/sources.list and uncommented the lines for Canonical's 'partner' repository.

7.2.2 Web Browsing and Java

Your Linux distro will come with a Web browser, probably Firefox, and possibly Konqueror in addition. I usually use Firefox. Chrome is nice, but I really like the plugins available for Firefox. But believe it or not, sometimes I use the famous text-based browser, lynx. In some cases, it is just plain quicker and easier. Moreover, you can do cool tricks, such as recording keystrokes for later playback, thus enabling one to do certain Web operations automatically.

If you use Ubuntu, your system may not be configured for Java in Web browsing. If so, do

```
sudo apt-get install openjdk-7-jre
sudo apt-get install icedtea-7-plugin
```

7.2.3 HTML Editing

I usually use Vim, along with some macros I've written for HTML editing, but I sometimes use Amaya, which is a full-featured GUI HTML editor, written by the Web policy consortium. One nice feature is that you can actually use the embedded Web links, good for testing them. See my tutorial at <http://heather.cs.ucdavis.edu/~matloff/amaya.html>. There are many newer and more powerful packages, such as Quanta+, Bluefish and NVu.

7.2.4 Compilers

Some distros come with the GCC suite. Ubuntu, for example, does not, but it can be downloaded via

```
sudo apt-get install build-essential
# may need to do this separately:
sudo apt-get install libc6-dev # C library
```

7.2.5 Integrated Software Development (IDE)

For programming work, I rarely use IDEs, as they are slow to load, take up too much space, and often don't allow me to use my own text favorite editor. I find that the vim editor (cited above) and the ddd GUI interface to the gdb debugging tool, work great together. For example, in vim I can type: make (which I have aliased to just M, or with gvim click on the make icon, and the source code I'm debugging will be recompiled. And as I've mentioned, it's important to me that I use the same text editor for all applications, which most IDE would not allow me to do. I use either GDB (try CGDB!) or DDD for my debugging tool. See my tutorials at <http://heather.cs.ucdavis.edu/~matloff/vim.html> and <http://heather.cs.ucdavis.edu/~matloff/debug.html>.

DDD is also usable with my favorite programming language, Python.

However, if you love IDEs, try Eclipse. I've got a tutorial that is more complete than most, at <http://heather.cs.ucdavis.edu/~matloff/eclipse.html>. It can be used with C, C++, Java, Perl, Python and many others.

Another system that has become quite popular is NetBeans. For R programming, RStudio and StatET are both first-rate.

7.2.6 Word Processing

I use L^AT_EX because of its flexibility, its beautiful output, and its outstanding ability to do math. You may like Lyx, which is a great GUI interface to L^AT_EX which is especially good for math work. See my tutorials at <http://heather.cs.ucdavis.edu/~matloff/latex.html> and <http://heather.cs.ucdavis.edu/~matloff/lyx.html>.

Install by running

```
sudo apt-get install texlive
# you may also need:
sudo apt-get install texlive-fonts-recommended
```

If you wish to work with files compatible with the Microsoft Office environment, there is a free suite of programs, OpenOffice, which provide Microsoft compatibility. It is packaged with most Linux distributions. If you would like something that quickly converts an Office file to rough text form, say to use with e-mail attachments, try Antiword. In Ubuntu, install via

```
sudo apt-get install antiword
```

7.2.7 Playing Movies, Music, Etc.

MPlayer is free and outstanding. Its capabilities are amazingly broad.

The documentation is extensive, and hard to navigate, but here are a couple of things to get you started: Installation: It's easy in Ubuntu:

```
sudo apt-get install mplayer
sudo apt-get install mencoder
```

Otherwise, build it yourself, as follows

One downloads the source code, MPlayer-1.0pre7try2.tar.bz2 and the codecs, essential-20041107.tar.bz2, from www.mplayerhq.hu/design7/dload.html.

Unpack the codecs file first,

```
tar xjf essential-20041107.tar.bz2
```


This creates a new directory. Copy the contents of that directory to the directory `/usr/local/lib/codecs` (use `mkdir` to create it if necessary). (Note: There may be legality issues with some codecs. When in doubt about a particular codec, you should obtain it from a site like Fluendo that offers it for a nominal fee, See a discussion at <http://fedoraproject.org/wiki/CodecBuddy>.

Now, unpack the source code file, and go into the directory it creates. Then go through the usual sequence for building open-source software from source:

```
configure
make
make install
```

Note that if you want to use the GUI, the `configure` command should be

```
configure --enable-gui
```

After `make install` is done, you will probably get a message something like

```
*** Download font at http://www.mplayerhq.hu/dload.html
*** for OSD/Subtitles support and extract to
/usr/local/share/mplayer/font/
*** Download skin(s) at http://www.mplayerhq.hu/dload.html
*** for GUI, and extract to /usr/local/share/mplayer/skins/
```

The fonts are needed for the subtitles (and for the GUI, if you use it). Just the iso1 font is needed. Download the font package, go to the indicated directory (`/usr/local/share/mplayer/font/` in the above example), and then do the unpack operation. This will produce a subdirectory, e.g. `font-arial-iso-8859-1`.

Viewing a video:

To play a video or audio file, say `x.avi`, type

```
mplayer x.avi
```

If you specify several files, as a playlist, it will play them all. Hit the Enter key if you want to skip the rest of the current file and go to the next one.

You have the following controls:

- right and left arrow keys to go back or forward 10 seconds
- down and up arrow keys to go back or forward 1 minutes
- PgDown and PgUp keys to go back or forward 10 min
- left- and right-bracket keys to decrease/increase speed by 10%, or left- and right-brace for 50%;
- Backspace key to return to normal speed
- Space bar to pause, then `.` to go forward frame by frame, Space bar to resume play
- `f` to go full screen
- `q` to quit

You can use `mplayer`, actually `mencoder`, which comes with the package, to do format conversion, e.g. AVI to MPG, change aspect ratio, and even do some primitive editing.

There are many, MANY,MANY different options. You may wish to try other players, e.g. VLC.

7.2.8 Video Editing

Try Kino, Cinelerra, LiVES and many others.

7.2.9 Image Viewing, Manipulation and Drawing

I use xpdf to view PDF files, though Acroread for Linux is available. I like the fact that xpdf allows me to copy ASCII text from the file. Others popular with Linux are evince, okular and MuPDF. For collections of JPEG files and the like, I use xzgv, gqview and gwenview; for viewing a single image, I use qiv.

Want something like Adobe Photoshop? The GIMP program is quite powerful, and free. It's included with most Linux distributions.

You can use GIMP to draw, but for “quick and dirty” tasks, I would suggest Dia, at <http://www.gnome.org/projects/dia/>.

7.2.10 FTP

I usually use the text-based ftp and sftp, the latter being an SSH version for security.

If you do frequent uploads/downloads to/from a particular site and wish to automate them, another text-based program, yafc, is excellent. A very nice GUI program, though, is gftp, which you can download from the Web if your Linux system doesn't already have it. In addition to the GUI, this program also has some functionality which ordinary FTP programs don't have.

7.2.11 Statistical Analysis

Use the statistical package that the professional statisticians use—R!

In my opinion from the point of view of someone with a “foot in both camps”—I'm a computer science professor who used to be a statistics professor—the R statistical package is the best one around, whether open source or commercial. It is statistically modern and correct, and it also is a general-purpose programming language.

I have a tutorial on R at <http://heather.cs.ucdavis.edu/~matloff/r.html>.

Install via

```
sudo apt-get install r-base
```

7.2.12 Video Chat

Currently, this is an area in which many Linux distros need work. Ubuntu comes with Ekiga, which works best if the person you're chatting with has Ekiga too. Skype has a Linux version, which many people use, but some have found to have problems. As of this writing, Google does not offer Google Talk for Linux. However, Empathy can be used. There are driver issues for some Webcams. Those using the UVC protocol are supposed to work on Linux, with the uvcvideo driver that comes with Linux. But again, there may be problems.

7.2.13 Running Windows Applications from Within Linux

I am simply not a Windows user, but on occasion there is a Windows program I need to run from within Linux. The simple way, if it works, is the WINE Windows emulator. Your distro may include it (type which wine in a terminal window to check); if not, download it, with for example the Ubuntu command being

```
udo apt-get install wine
```

For more involved applications, you may wish to try one of the virtual machine packages. See <http://heather.cs.ucdavis.edu/~matloff/vm.html> for a brief introduction.

7.3 Downloading New Software

There is a vast wealth of free software for Linux on the Web. Here's how to obtain and install it.

7.3.1 How to Find It

These days most downloads and installs are done automatically, say with yum or apt-get, as seen in Section 7.3.2 below. That helps you find it too. If you want to find application Z, instead of plugging “Z” into Google, plug “yum install Z” or “apt-get install Z” so as to narrow down the volume of response.

^sIn some respects, it’s even better than S, the commercial product it is based on.

7.3.2 Automatic Download/Installation

In recent years, most Linux distros have made it very easy to download and install new software. In Fedora, for instance, one uses the yum command. For example, to download the program yafc mentioned above, one simply types yum install yafc. In Ubuntu, there is the apt-get command, which works similarly. For instance, to download the xpdf PDF viewer, I typed

```
sudo apt-get install xpdf
```

(See Section 6.0.1 for an explanation of sudo. Ubuntu may ask you to install from your CD-ROM, but yours may be incomplete. If so, comment out the first line of /etc/apt/sources.list; this is the line telling Ubuntu to install from the CD-ROM.) For those who prefer GUIs, Ubuntu offers the Synaptic package manager. With both yum and apt-get, one can direct where to download from, by making the proper entries in the file etc/apt/sources.list. For instance, for the R statistical package above, apt-get may not find it on its own, in which case we can add a line

```
deb http://cran.stat.ucla.edu/bin/linux/ubuntu gutsy/
```

to etc/apt/sources.list, telling apt-get that here is an alternative place it can look. (This is for the Gutsy edition of Ubuntu.)

By default apt-get will try to retrieve your requested program from your installation CD/DVD. You can change this by commenting-out the line in etc/apt/sources.list that begins with

```
deb cdrom:
```

Sometimes it may not be clear which package name to use with yum or apt-get. For instance, to install the GCC compiler, C library and so on, the command is

```
sudo apt-get install build-essential
```

How did I learn this? I did a Web search for “apt-get GCC.” To install the curses library (and include file), do

```
sudo apt-get install libncurses5-dev
```

7.3.3 Debian/Ubuntu .deb Files

The Debian distro of Linux uses its own packaging for downloaded programs, which you’ll see as files whose names have a .deb suffix. Ubuntu, as a derivative of Debian, uses this too.

Usually you will not need to work directly with these files, since you will use apt-get or Synaptic. But if you do download such a file directly from the Web, use gdebi to install it; the GUI version is gdebi-gtk.

7.3.4 Using RPMs

Though the methods in Section 7.3.2 have now made RPMs less important, you may find that the software you want comes in an RPM package, with a.rpm suffix in its name. To install such a package,

type `rpm -i package_file_name` If you later wish to remove, i.e. uninstall a package, you can use `rpm -e` ('e' stands for "erase"). You do NOT have to have the RPM file present to do this. Some packages will have different versions for different C libraries. Red Hat uses glibc. Type `ls -l /lib/libc*` to see which version you have. You may find that you need some library files for a program you download, and that you are missing those files. You can usually get these from the Web too. If a program complains about a missing file, try the `ldd` command (e.g. `ldd x` if the name of the program which needs the library is `x`); this will tell you which libraries are needed, where they were found on your system, and which ones, if any, were not found.

8 Learning More About Linux

The only way to really learn Linux is to use it on a daily basis for all your computer work—e-mail, word processing, Web work, programming, etc. As you do this, the expertise you'll want to pick up includes: file, directory and mount operations; process operations; roles of system directories (`/usr`, `/etc`, `/dev`, `/sbin` and their various subdirectories, e.g. `/usr/lib`; search paths; network operation and utilities such as `netstat`; and so on. Don't try to do this all at once. Instead, take your time, and learn these naturally, as the need arises. As you use Linux more and more in your daily computer application work (e-mail, word processing, etc.), the needs will arise as you go along. And remember, there's lots of help available if you need it.

_ If you are running Ubuntu or one of its offshoots, the Ubuntu Forums, <http://ubuntuforums.org/> is an excellent resource.

- Linux home page, at <http://www.linux.org/> Lots and lots of information is available here. www.linux.com. Chock full of information and links.
- Google's excellent set of links to various Linux sites, http://directory.google.com/Top/Computers/Software/Operating_Systems/Linux
- Another good set of Linux links, <http://www.linuxjunior.org/resources.shtml>
- If you are having trouble with specific hardware in your Linux installation, an excellent place to go for detailed information is the Linux HOW-TO documentation. (For the same reason, if you are about to purchase a machine and suspect that some of the hardware is nonstandard, you can check the corresponding Linux HOW-TO to see if there are any problems with that hardware.

The HOW-TO documents are available at many sites, such as the one at linux.org.

There are Linux Users Groups (LUGs) in virtually every city. You can join if you wish, or just get to know them casually. They are great sources of help! And by the way, many of them hold monthly Linux Installfests, where you can see Linux being installed or have it installed on your own machine.

9 Advanced Linux Usage

9.1 Dual-Boot Issues

You may wish to change some parameters of your dual-boot process, e.g. change the default OS. You can do this by editing the configuration file for your bootloader.

Most distros today use GRUB as their bootloader. Its configuration file is `/boot/grub/menu.lst`. By the way, note that GRUB's notation for partitions is (drive ID, partition number), so that for instance `(hd0,1)` means the second partition in the first hard drive.

9.2 Live CDs or USB-Key Based Linux As Rescue Tools

Among other things, Knoppix has developed a reputation as being useful as an OS rescue/repair tool, including for Windows! And now, most of the live CDs or USB-key based Linux installations can be used this way. A common usage is to either fix broken files or at least make copies of important user files. It may be, for instance, that Windows is not bootable due to corruption, but by using a Linux

rescue CD/USB key, we can access individual files. Here is a typical pattern. One brings up a terminal window and then:

```
sudo -s # get root privileges; could try su root instead
fdisk -l # check where the partitions are
# say /dev/sda1 is of interest
mkdir x
mount /dev/sda1 x
cd x
# now you have those files at your disposal
```

In one case, I forgot my password on an Ubuntu netbook. I could fix it as root if I could boot up in Ubuntu recovery mode, but unfortunately the GRUB bootloader was configured with a timeout value of 0 seconds, giving me no way to choose recovery mode. So, I booted up Linux from a USB key (Section 4.2), mounted my Ubuntu file system as above, and then edited the GRUB startup file, `/boot/grub/menu.lst`, changing the timeout value to 5 seconds.

The preceding operations can be done by booting almost any Linux distro, but Knoppix is nicer as it comes with two very nice utilities (both can be obtained separately as well):

- `testdisk`: This does a lot of diagnostics on your hard drive, recover lost partitions, undelete deleted files, fix boot sectors and so on.
- `ntfsfix`: May be able to fix your broken NTFS partition.
- `photorec`: Quite a program! It bypasses your (possibly broken) file system, and looks for files by going through your hard drive literally bit by bit, looking for bytes that encode any of 180 known file types, e.g. `.jpg`, `.avi`, `.pdf` etc.

9.3 Troubleshooting

One of Linux’s biggest strengths is its stability. If you are tired of getting Windows’ infamous “blue screen of death,” then Linux is the OS for you. (It is also subject to far fewer virus and other attacks than Windows.) So emergencies are rare, but they can happen. Here are some tips for such cases.

9.3.1 Tools

Here are some commands you can run in a terminal window that you can use to investigate:

- `ps`: Tells you what processes are running. Typically one uses this with something like the `ax` option.
- `dmesg`: Tells you the major events that have occurred on your machine ever since it was last booted up.
- `lsmod`: This tells you what OS modules are installed, i.e. device drivers and the like.
- `lpq`: Lists the current printer queue.
- `lsusb`: Lists what USB devices are currently plugged in.
- `ifconfig`: Lists network interfaces.
- `iwconfig`: Lists currently operating wireless devices.
- `iwlist`: Lists wireless access points in range.
- `netstat`: Lists current network connections.

9.3.2 WiFi Networking

The newer versions of the major distros handle WiFi configuration pretty well without your intervention. But if you have problems, the material in this section may be helpful.

9.3.3 General Information

Below is a five-minute crash course in WiFi. Even if you don’t understand all of it, even partial understanding may be helpful.

- Recall that in Unix-family operating systems, I/O devices are represented as “files” in the directory /dev. Your WiFi device is probably eth1 or wlan0.
- Your WiFi device needs a driver. Many, if not most, laptops use Broadcom WiFi hardware, and in older Linux distros, they needed some fiddling to work, but now it’s much easier (see below).
- The names of wireless access points are called ESSIDs.
- If you are connected to a router or a wireless access point, your machine is probably assigned an IP address via DHCP, rather than statically. An error message like “no lease offered” means that the DHCP process failed.
- DNS servers convert an “English” address like www.google.com to a numerical address like 209.85.171.103. So your OS needs to set up a connection to a DNS server.

9.3.4 Network Management Tools

If you are running the GNOME windows manager, select System j Administration j Network. There is also an icon you can click in the toolbar; it looks like two black monitors when you are not connected, and is a set of blue bars indicating signal strength when you are connected. Note that left- and right-clicking gives different results, so try both. In KDE, select System j Network Device Control. You can activate/deactivate your network card during a session. In GNOME, this is done via System j Administration j Network. The network managers included with most Linux distros are rather primitive. An excellent alternative is WiFi Radar. In Ubuntu, install via

```
sudo apt-get install wifi-radar
```

9.3.5 Individual Linux Network Commands

Useful commands from a terminal window include:

- iwlist: You can determine which ESSIDs are within range of you by typing the command

```
$ sudo iwlist eth1 scanning
```

say if your wireless device is eth1.

- ifconfig: Shows information about all your network interfaces, i.e. their hardware addresses, IP addresses and so on. Lack of IP address on your wireless port, e.g. wlan0 or eth1, may indicate that DHCP has failed. This command can also be used to set the IP address and other parameters “manually,” deactivate/reactive a network interface, etc.
- iwconfig: Shows information about all your wireless connections. Also can be used by you to specify which access point you wish to use. For example, to select a particular wireless access point named X, type

```
sudo iwconfig wlan0 essid "X"
```

(assuming wlan0 is your wireless interface)

- dmesg: Shows a record of your last bootup. This may show error messages regarding your WiFi card. It’s pretty long, so either run it through more, i.e. run

```
dmesg | more
```

or save it to a file, say dmesg.out, and then explore the file at your leisure with a text editor.

- _ route: Displays the current packet routing table.
- _ ethtool: Running

```
ethtool eth0
```

will give you information about your Ethernet link, e.g. link speed. To get statistics on recent usage, run

```
ethtool --statistics eth0
```

Some of these must be used with root privilege. For example, running

```
iwlist eth1 scanning
```

may produce no access points, while

```
sudo iwlist eth1 scanning
```

will show you all of them.

The file `/etc/resolv.conf` lists the IP addresses of the DNS servers. You can add more nameserver lines if you know of some, say from your ISP (of for that matter, other ISPs).

9.3.6 If You Have a Problem

These days, Linux generally does well with WiFi, and it might work for you “right out of the box,” with no configuration on your part. If not, this section is for you. Some wireless network cards typically sold with PCs today do not have direct Linux drivers available. A common example is the Broadcom BCM43XX series. However, you can still operate as usual after some preparation, as explained below.

Ubuntu: BCM43XX Series

Ubuntu handles Broadcom cards well, as long as you have Linux kernel 2.6.15 or newer. (Run `dmesg` if you want to check this.) You simply need to take the following action once:

First establish an Ethernet connection to the Internet, to enable download. For example, if you have a router at home, even a wireless one, connect your machine directly to the router with an Ethernet cable. Then Select System Administration Hardware Devices (the last might be labeled Additional Drivers). It will ask you if you want to download the Broadcom firmware, so say yes. Check the Enable box for Firmware for Broadcom 43 Wireless Driver. You will be asked whether you want the firmware to be downloaded from the net; say yes. Then check Enabled after the download. Know YourWiFi Card You first need to determine which wireless card you have. On the laptop I use now, I determined this by running `dmesg` and `lspci` under Linux. Sure enough, it turned out to be a Broadcom BCM43XX series card. Other Cards/Kernels

For other cards, go to the `ndiswrapper` home page, <http://ndiswrapper.sourceforge.net/>.

The program `ndiswrapper` allows Linux to use Windows drivers.

9.3.7 A Program Freezes

If an application program freezes up and you invoked it from the command line within a shell, you can in most cases kill it by hitting `Ctrl-c` in the terminal window from which invoked it. If this doesn't work, run the “`processes`” command by typing

```
ps ax
```

in another terminal window, and noting the process number of your program. Say for concreteness that that number is 2398. Then type

```
kill -9 2398
```

to kill the program. If you have a program named, say, xyz, the command

```
kill -9 xyz
```

kills all running instances of the program.

9.3.8 Screen Freezes

What if your entire screen freezes up? Again, this should be quite rare, but it is possible. I recommend the following remedies, in order:

- _ In Gnome hit Alt F2, which will bring up a little window in which you can run a command, say kill as above.
- _ In Gnome, hit Ctrl Alt T, which will create a new terminal window, from which you can kill the offending program.
- _ Try going to another screen! Linux allows you to switch among multiple screens. In Gnome, for instance, you can switch to the second screen from the first via Ctrl Alt Right, and go back via Ctrl Alt Left. Then open a terminal window in the new screen, find the process number of the program and kill the program, as described above.
- _ In Gnome, try hitting Ctrl Alt Del). This should cause an exit from Linux's X11 windowing system but not an exit from Linux itself. You would then get an opportunity to log in again.
- Try NOT to simply poweroff the machine, as that may do damage to your files. It may not be permanent damage, as the OS will try to fix the problems when you next reboot, but don't just pull the plug unless you have no other recourse.

9.4 Accessing Your Windows Files from Linux

At this point, most Linux distributions, except Fedora/Red Hat, give you access (at least read access) to your Windows partition from Linux. For some of them, they may do this automatically, in which case your Windows partition, say /dev/hda1 should be visible in the file /etc/fstab. If not, mount it yourself:

```
mkdir /dosc
mount /dev/hda1 /dosc
cd /dosc
```

You should now see your Windows files, and should be able to access them on at least a read basis. For more information, including concerning write access, see the Linux-NTFS Project, <http://www.linux-ntfs.org/>.

A What Is Linux?

Linux is a form of the Unix operating system. Though originally Unix was used mainly by engineers and scientists and thus was not very familiar to the general public, a lot of what you take for granted on computer systems today began in Unix. A notable example is the Internet—the first major operating system to implement the TCP/IP protocol at the heart of the Internet was Unix, and that led to the general acceptance of the protocol. The Apple Macintosh operating system is based on a form of Unix, and the Android system is based on Linux.

In the early 1990s, computer science student Linus Torvalds decided to write his own version of Unix, which he called Linux. Other “homegrown” versions of Unix had been written, such as MINIX, but what distinguished Linux was the scale of worldwide participation involved. Torvalds innocently put a message on the Internet asking if anyone wanted to help, and he got a torrent of responses. There are a several reasons why Linux is mainstream today. First, it became known as a very reliable, stable operating system, with one result being that Linux has become a major platform for large corporate

Web servers. Another reason is that it is free, as is the vast majority of the software associated with it developed elsewhere. Many companies have found that it is cheaper to run Linux on their PCs, both for this reason and because of reduced maintenance costs. There are several good reasons for you to use Linux:

- As mentioned, Linux is becoming one of the “hottest” software systems. Virtually all of the major companies are promoting it, and as mentioned Linux is a leading corporate choice for Web servers. Linux is the main operating system used at , and in fact they developed their own version of Linux, Goobuntu (a play on Ubuntu, one of the most popular versions of Linux).
- Linux is also starting to make inroads in large desktop markets, such as businesses, schools and so on, due to its high reliability, far lower rate of infection by viruses compared to Windows, and its low cost.
- The Linux community shares. That means that people online are much more willing to help you, and more open source software is available. If you are a university computer science student, there are some very important additional advantages:
- Many CS courses make specific use of Unix, and thus their work cannot be done on Windows platforms.
- Since it is a full Unix system, Linux allows students to do their homework in the comfort of their own homes. If you are new to Unix, click here for my Unix tutorial Web page at <http://heather.cs.ucdavis.edu/~matloff/unix.html>, which will introduce you to Unix file and directory commands, and so on.
- In installing and using Linux, students learn many practical things about computers which they do not learn in coursework. This practical experience can also help you in job interviews, both for permanent jobs after graduation and for summer jobs and internships/co-ops during your college years. Even if the job you interview for does not involve Linux, you will definitely impress the interviewer if, for example, you discuss various things you have done to use and customize your Linux system.
-

B What Is Partitioning?

It is probably not necessary for you to know the material here, and it is rather detailed, but you may find it useful at some point. I do recommend that you take a few minutes and read this section. A hard drive (not just for Linux) will consist of one or more partitions. A partition is a set of contiguous space (sequential blocks) on the disk, and is treated as an independent disk.

So, assuming you want your system to include both Windows and Linux (termed a dual boot situation, since you can boot either system), you will need at least one partition for Windows and one (actually two) for Linux. It’s important to understand how the naming works: In Linux systems, all I/O devices are treated as “files.” If your first hard drive is of the IDE type, the entire drive is probably called /dev/hda, i.e. the “file” had within the directory /dev. In the case of SATA-type hard drives, the notation is /dev/sda etc. Your first CD-ROM/DVD drive may be /dev/hdc (your third “hard drive”), your first USB port may be /dev/sdf1 and so on.

Partitions within, say, /dev/hda, are called /dev/hda1, /dev/hda2 and so on.

Your original Windows single partition was probably /dev/hda1 or /dev/sda1. Within a partition you’ll have some type of file system. The disk consists simply of a long stream of bytes, with no structure, so the OS needs to have a way of organizing them into files, recording where in that stream each file has its bytes. But you don’t need to know the details. Windows XP and Vista use the NTFS file system. The standard Linux file system is ext2 (number 0x83, sometimes called Linux native), or possibly ext3, for your main Linux partition and of type swap for your swap partition (number 0x82, used for temporary storage during the time the OS is running). PCs were originally designed to have up to four “real” partitions, called primary partitions. After people found that to be too constraining, logical or extended partitions were invented. You should install Linux in a primary partition, for recovery reasons, but it is not necessary.

B.1 Partitioning Using GParted

Today most distros will invoke a partitioning program to do your partitioning. This could be the famous GParted program, or one that the authors of your distro wrote themselves.

You can use GParted on your own by downloading and booting a GParted live CD (or USB key), but I'll assume here that your Linux installation program invokes either GParted or another program written specifically for your distro. Since every distro will handle this a bit differently, what I will do here is just give you an understanding of what operations need to be done, with the specific mouse clicks needed varying from one distro to another. I'll assume that you want your Windows and Linux systems to coexist on the same hard drive. So when your distro's installer program asks you whether you want to use the entire disk, be sure to say no! Of course, if you do want to erase Windows, or if you are installing Linux on a separate drive from Windows, you can go ahead and use the whole drive.

Here are the main steps in GParted, roughly stated (you may see some variation):

- Select the disk you wish to repartition. If you have only one disk, it will be something like `/dev/hda`. (See Section B.)
- Select the partition where Windows resides. This will typically cover the entire disk, and will almost certainly be of file system type NTFS. I'll assume that here.
- Decide how much space you want to remove from the Windows partition in order to make a partition for Linux.
- Now resize, in this case shrink, the Windows partition. The partitioner will ask you how much room to make.
- Adjust the partition size according to your desired value.
- You'll need to make the main Linux partition primary, of type `ext2` or `ext3`, and set to be bootable.
- You'll need a smaller partition of type `linux-swap`. This is not used for files, but rather as "scratch space" by the OS, for virtual memory and for storage when your machine is in hibernate mode.
- You'll then have to commit, i.e. save, the changes to the partitions. This might take a few minutes, so be patient.
- The next time you boot Windows; you will be asked if you want a disk consistency check. Definitely say yes.

TOR – The Onion Relay

(Links route to www.torproject.org)

Tor: Overview

Topics

- [Overview](#)
- [Why we need Tor](#)
- [The Solution](#)
- [Staying anonymous](#)
- [The future of Tor](#)

Overview

The Tor network is a group of [volunteer](#)-operated servers that allows people to improve their privacy and security on the Internet. Tor's users employ this network by connecting through a series of virtual tunnels rather than making a direct connection, thus allowing both organizations and individuals to share information over public networks without compromising their privacy. Along the same line, Tor is an effective censorship circumvention tool, allowing its users to reach otherwise blocked destinations or content. Tor can also be used as a building block for software developers to create new communication tools with built-in privacy features.

Individuals use Tor to keep websites from tracking them and their family members, or to connect to news sites, instant messaging services, or the like when these are blocked by their local Internet providers. Tor's [hidden services](#) let users publish web sites and other services without needing to reveal the location of the site. Individuals also use Tor for socially sensitive communication: chat rooms and web forums for rape and abuse survivors, or people with illnesses.

Journalists use Tor to communicate more safely with whistleblowers and dissidents. Non-governmental organizations (NGOs) use Tor to allow their workers to connect to their home website while they're in a foreign country, without notifying everybody nearby that they're working with that organization.

Groups such as Indymedia recommend Tor for safeguarding their members' online privacy and security. Activist groups like the Electronic Frontier Foundation (EFF) recommend Tor as a mechanism for maintaining civil liberties online. Corporations use Tor as a safe way to conduct competitive analysis, and to protect sensitive procurement patterns from eavesdroppers. They also use it to replace traditional VPNs, which reveal the exact amount and timing of communication. Which locations have employees working late? Which locations have employees consulting job-hunting websites? Which research divisions are communicating with the company's patent lawyers?

A branch of the U.S. Navy uses Tor for open source intelligence gathering, and one of its teams used Tor while deployed in the Middle East recently. Law enforcement uses Tor for visiting or surveilling web sites without leaving government IP addresses in their web logs, and for security during sting operations.

The variety of people who use Tor is actually [part of what makes it so secure](#). Tor hides you among [the other users on the network](#), so the more populous and diverse the user base for Tor is, the more your anonymity will be protected.

Why we need Tor

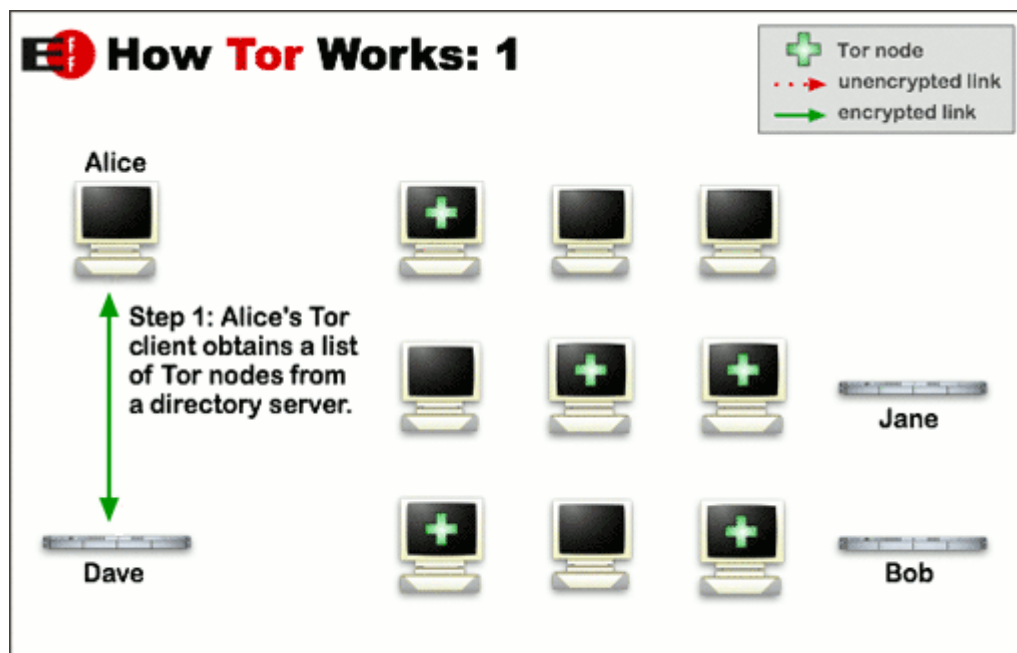
Using Tor protects you against a common form of Internet surveillance known as "traffic analysis." Traffic analysis can be used to infer who is talking to whom over a public network. Knowing the source and destination of your Internet traffic allows others to track your behavior and interests. This can impact your checkbook if, for example, an e-commerce site uses price discrimination based on your country or institution of origin. It can even threaten your job and physical safety by revealing who and where you are. For example, if you're travelling abroad and you connect to your employer's computers to check or send mail, you can inadvertently reveal your national origin and professional affiliation to anyone observing the network, even if the connection is encrypted.

How does traffic analysis work? Internet data packets have two parts: a data payload and a header used for routing. The data payload is whatever is being sent, whether that's an email message, a web page, or an audio file. Even if you encrypt the data payload of your communications, traffic analysis still reveals a great deal about what you're doing and, possibly, what you're saying. That's because it focuses on the header, which discloses source, destination, size, timing, and so on.

A basic problem for the privacy minded is that the recipient of your communications can see that you sent it by looking at headers. So can authorized intermediaries like Internet service providers, and sometimes unauthorized intermediaries as well. A very simple form of traffic analysis might involve sitting somewhere between sender and recipient on the network, looking at headers.

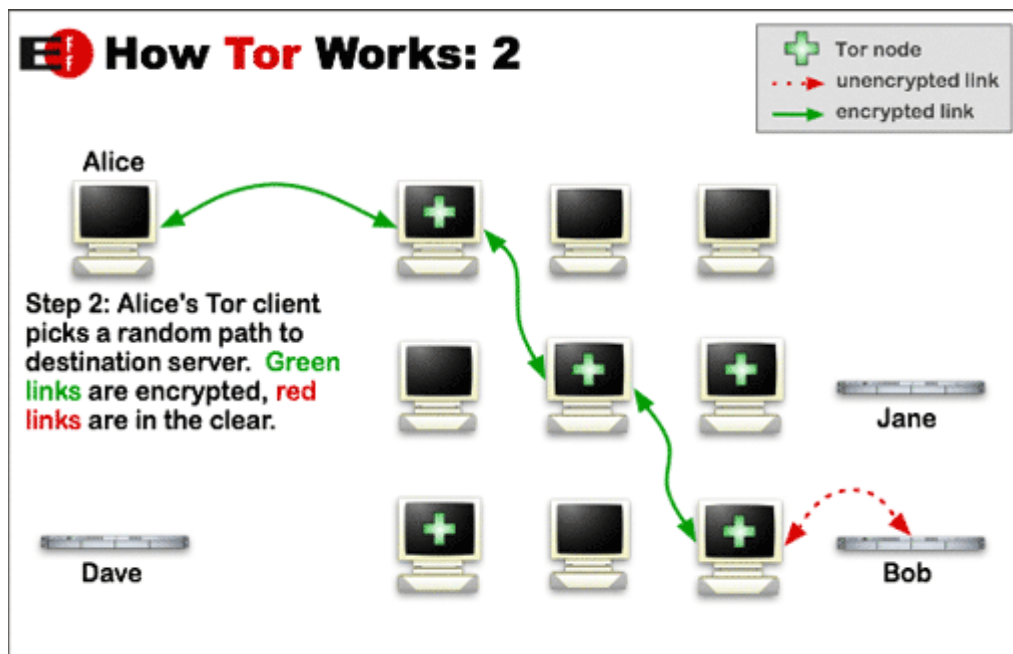
But there are also more powerful kinds of traffic analysis. Some attackers spy on multiple parts of the Internet and use sophisticated statistical techniques to track the communications patterns of many different organizations and individuals. Encryption does not help against these attackers, since it only hides the content of Internet traffic, not the headers.

The solution: a distributed, anonymous network



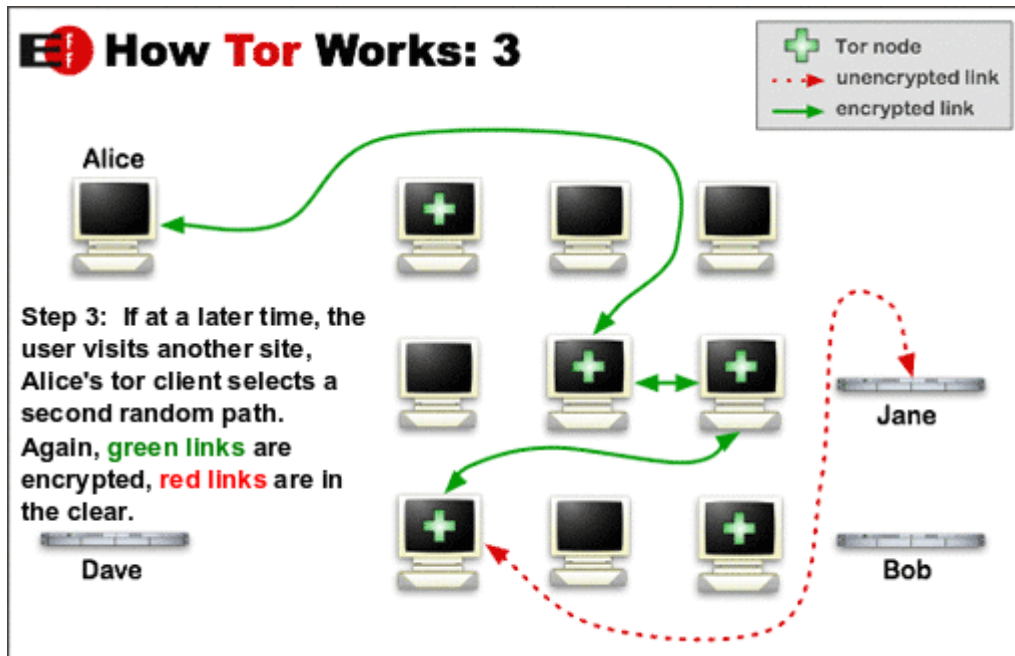
Tor helps to reduce the risks of both simple and sophisticated traffic analysis by distributing your transactions over several places on the Internet, so no single point can link you to your destination. The idea is similar to using a twisty, hard-to-follow route in order to throw off somebody who is tailing you — and then periodically erasing your footprints. Instead of taking a direct route from source to destination, data packets on the Tor network take a random pathway through several relays that cover your tracks so no observer at any single point can tell where the data came from or where it's going.

To create a private network pathway with Tor, the user's software or client incrementally builds a circuit of encrypted connections through relays on the network. The circuit is extended one hop at a time, and each relay along the way knows only which relay gave it data and which relay it is giving data to. No individual relay ever knows the complete path that a data packet has taken. The client negotiates a separate set of encryption keys for each hop along the circuit to ensure that each hop can't trace these connections as they pass through.



Once a circuit has been established, many kinds of data can be exchanged and several different sorts of software applications can be deployed over the Tor network. Because each relay sees no more than one hop in the circuit, neither an eavesdropper nor a compromised relay can use traffic analysis to link the connection's source and destination. Tor only works for TCP streams and can be used by any application with SOCKS support.

For efficiency, the Tor software uses the same circuit for connections that happen within the same ten minutes or so. Later requests are given a new circuit, to keep people from linking your earlier actions to the new ones.



Staying anonymous

Tor can't solve all anonymity problems. It focuses only on protecting the transport of data. You need to use protocol-specific support software if you don't want the sites you visit to see your identifying information. For example, you can use [Tor Browser](#) while browsing the web to withhold some information about your computer's configuration.

Also, to protect your anonymity, be smart. Don't provide your name or other revealing information in web forms. Be aware that, like all anonymizing networks that are fast enough for web browsing, Tor does not provide protection against end-to-end timing attacks: If your attacker can watch the traffic coming out of your computer, and also the traffic arriving at your chosen destination, he can use statistical analysis to discover that they are part of the same circuit.

The future of Tor

Providing a usable anonymizing network on the Internet today is an ongoing challenge. We want software that meets users' needs. We also want to keep the network up and running in a way that handles as many users as possible. Security and usability don't have to be at odds: As Tor's usability increases, it will attract more users, which will increase the possible sources and destinations of each communication, thus increasing security for everyone. We're making progress, but we need your help. Please consider [running a relay](#) or [volunteering](#) as a [developer](#).

Ongoing trends in law, policy, and technology threaten anonymity as never before, undermining our ability to speak and read freely online. These trends also undermine national security and critical infrastructure by making communication among individuals, organizations, corporations, and governments more vulnerable to analysis. Each new user and relay provides additional diversity, enhancing Tor's ability to put control over your security and privacy back into your hands.

Tails

(Links route to <https://tails.boum.org/about/index.en.html>, [debian.org](https://www.debian.org), or to [torproject.org](https://www.torproject.org))

Tails is a live system that aims to preserve your privacy and anonymity. It helps you to use the Internet anonymously and circumvent censorship almost anywhere you go and on any computer but leaving no trace unless you ask it to explicitly.

It is a complete operating system designed to be used from a DVD, USB stick, or SD card independently of the computer's original operating system. It is [Free Software](#) and based on [Debian GNU/Linux](#).

Tails comes with several built-in applications pre-configured with security in mind: web browser, instant messaging client, email client, office suite, image and sound editor, etc.

1. [Online anonymity and censorship circumvention](#)
 1. [Tor](#)
 2. [I2P](#)
2. [Use anywhere but leave no trace](#)
3. [State-of-the-art cryptographic tools](#)
4. [What's next?](#)
5. [Press and media](#)
6. [Acknowledgments and similar projects](#)

Online anonymity and censorship circumvention

Tor

Tails relies on the Tor anonymity network to protect your privacy online:

- all software is configured to connect to the Internet through Tor
- if an application tries to connect to the Internet directly, the connection is automatically blocked for security.

Tor is an open and distributed network that helps defend against traffic analysis, a form of network surveillance that threatens personal freedom and privacy, confidential business activities and relationships, and state security.

Tor protects you by bouncing your communications around a network of relays run by volunteers all around the world: it prevents somebody watching your Internet connection from learning what sites you visit, and it prevents the sites you visit from learning your physical location.

Using Tor you can:

- be anonymous online by hiding your location,
- connect to services that would be censored otherwise;

- resist attacks that block the usage of Tor using circumvention tools such as [bridges](#).

To learn more about Tor, see the official [Tor website](#), particularly the following pages:

- [Tor overview: Why we need Tor](#)
- [Tor overview: How does Tor work](#)
- [Who uses Tor?](#)
- [Understanding and Using Tor — An Introduction for the Layman](#)

To learn more about how Tails ensures all its network connections use Tor, see our [design document](#).

I2P

You can also use Tails to access [I2P](#) which is an anonymity network different from Tor.

[Learn how to use I2P in Tails in the documentation.](#)

To know how I2P is implemented in Tails, see our [design document](#).

Use anywhere but leave no trace

Using Tails on a computer doesn't alter or depend on the operating system installed on it. So you can use it in the same way on your computer, a friend's computer, or one at your local library. After shutting down Tails, the computer will start again with its usual operating system.

Tails is configured with special care to not use the computer's hard-disks, even if there is some swap space on them. The only storage space used by Tails is in RAM, which is automatically erased when the computer shuts down. So you won't leave any trace on the computer either of the Tails system itself or what you used it for. That's why we call Tails "amnesic".

This allows you to work with sensitive documents on any computer and protects you from data recovery after shutdown. Of course, you can still explicitly save specific documents to another USB stick or external hard-disk and take them away for future use.

State-of-the-art cryptographic tools

Tails also comes with a selection of tools to protect your data using strong encryption:

- [Encrypt your USB sticks or external hard-disks](#) using [LUKS](#), the Linux standard for disk-encryption.
- Automatically use HTTPS to encrypt all your communications to a number of major websites using [HTTPS Everywhere](#), a Firefox extension developed by the [Electronic Frontier Foundation](#).
- Encrypt and sign your emails and documents using the *de facto* standard [OpenPGP](#) either from Tails email client, text editor or file browser.
- Protect your instant messaging conversations using [OTR](#), a cryptographic tool that provides encryption, authentication and deniability.
- [Securely delete your files](#) and clean your disk space using [Nautilus Wipe](#).

[Read more about those tools in the documentation.](#)

What's next?

To continue discovering Tails, you can now read:

- the [warning page](#) to better understand the security limitations of Tails and Tor,
- more details about the [features and software](#) included in Tails,
- our [documentation](#) explaining in detail how to use Tails,
- some hints on why [you should trust Tails](#),
- our [design document](#) laying out Tails specification, threat model and implementation,
- the [calendar](#) that holds our release dates, meetings and other events.

Installing Tails

Here is the link you can use to find the Tails .iso you need in order to manually install Tails onto your drives. <https://tails.boum.org/download/index.en.html>

Manual Installation using Linux

Find out the device name of the device

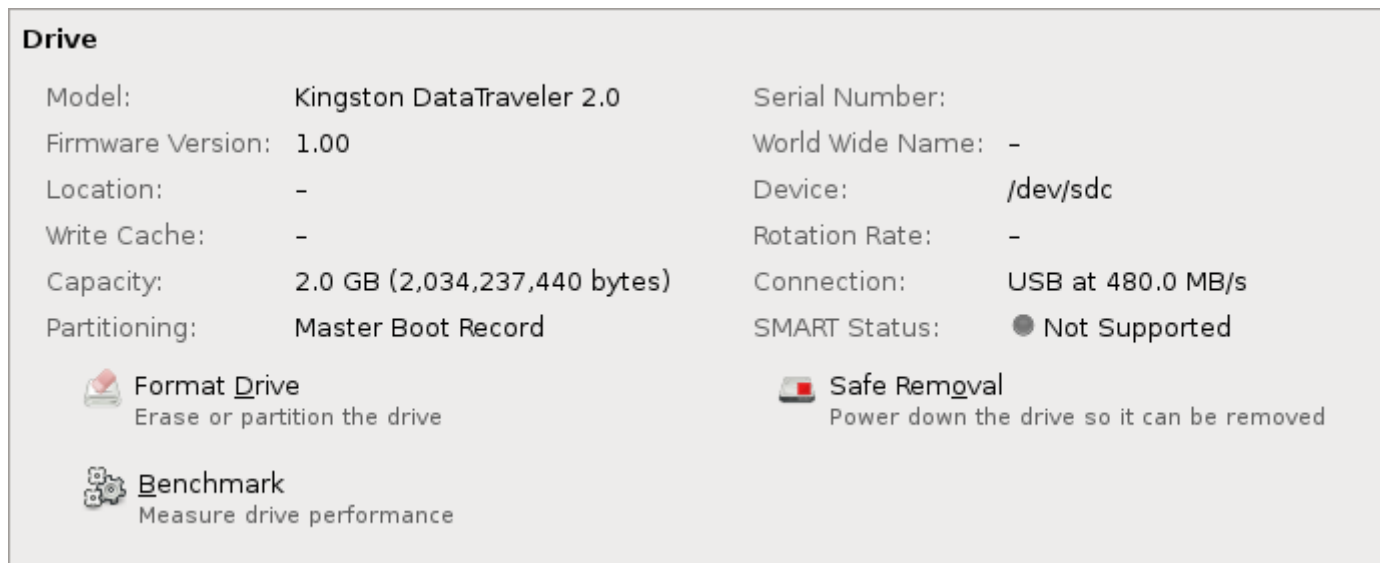
The device name should be something like `/dev/sdb`, `/dev/sdc`, etc.

If you are not sure about the exact device name, with GNOME, do the following:

1. Make sure that the USB stick or SD card onto which you want to install Tails is unplugged.
2. Open GNOME Disk Utility from the menu Applications ► Accessories ► Disk Utility
3. Disk Utility lists all the current storage devices in the left pane of the window.
4. Plug the USB stick or SD card onto which you want to install Tails.

A new device appears in the list of storage devices. Click on it.

5. In the right pane of the window, verify that the device corresponds to your device, its brand, its size, etc.



On this screenshot, the USB stick is a Kingston DataTraveler of 2.0 GB and its device name is `/dev/sdc`. Yours are probably different.

If you are not sure about the device name, you should stop proceeding or **you risk overwriting any hard disk on the system.**

Do the copy

All the data on the installed device will be lost.

Execute the following commands, replacing `[tails.iso]` with the path to the ISO image that you want to copy and `[device]` with the device name found in step 1.

```
dd if=[tails.iso] of=[device] bs=16M && sync
```

Here is an example of the commands to execute, yours are probably different:

```
dd if='/home/amnesia/Desktop/tails-0.6.2.iso' of=/dev/sdc bs=16M && sync
```

If you are not sure about the path to the ISO image or if you get a No such file or directory error message, you can first type `dd`, followed by a space, and then drag and drop the icon of the ISO image from a file browser onto the terminal. This should insert the correct path to the ISO image in the terminal. Then complete the command and execute it.

If you don't see any error message, Tails is being copied onto the device. The whole process might take some time, generally a few minutes.

Once the command prompt reappears, you can shutdown your computer, and [start Tails](#) from this new device.

Troubleshooting

`dd: /dev/sdx: No such file or directory`

Then double-check the name of the device you found in [step 1](#).

`dd: /dev/sdx: Permission denied`

You might also have committed a mistake in the device name, so please double-check it. If you are sure about the device name, this could be a permission problem and you could need to gain administration privileges before running the commands in the terminal. That could be:

```
sudo dd if=[tails.iso] of=[device] bs=16M && sync
```

`dd: tails.iso: No such file or directory`

Then you surely have committed a mistake on the path to the ISO image in [step 2](#).

Manual Installation using Windows

This technique uses the Universal USB Installer, for more info or more help visit <http://www.pendrivelinux.com/>.

Insert a USB stick with at least 2GB of free space

[Download the Universal USB Installer](#)

You will need version 1.9.5.4 or later.

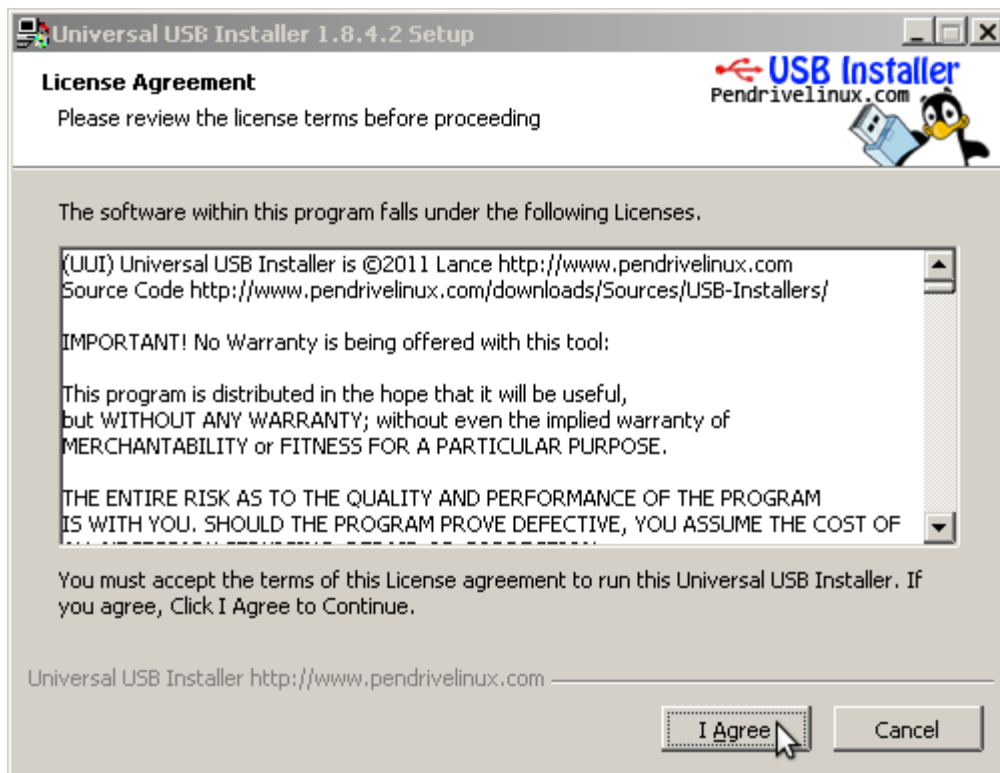
Click 'Run' when prompted



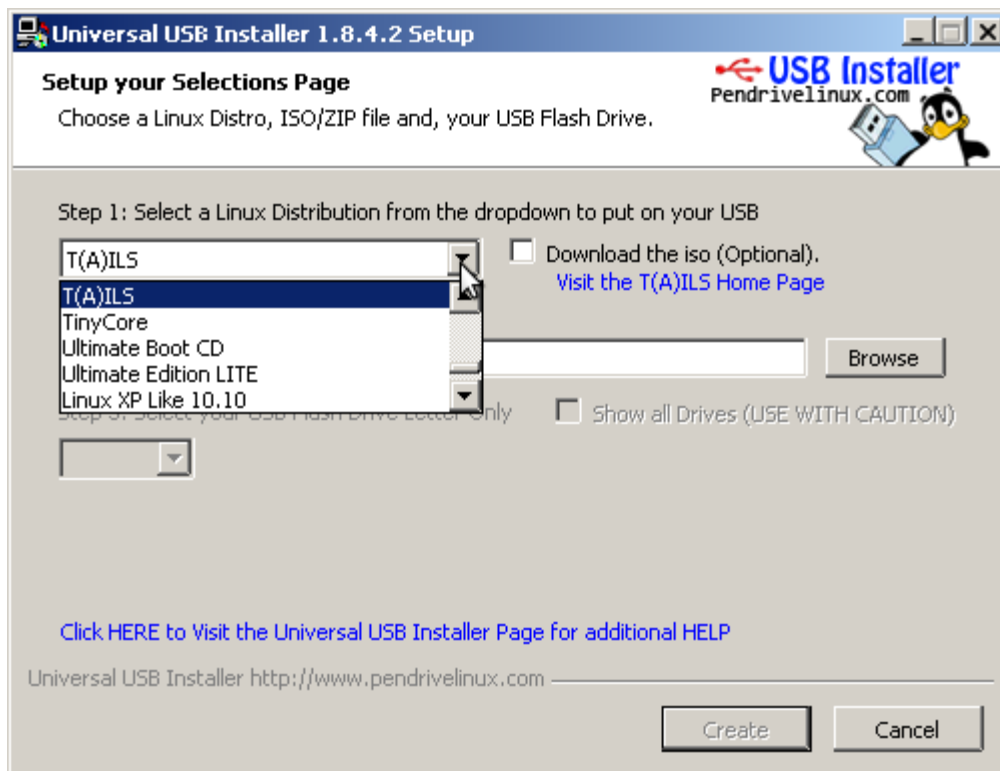
If the security dialog appears, confirm by clicking 'Run'



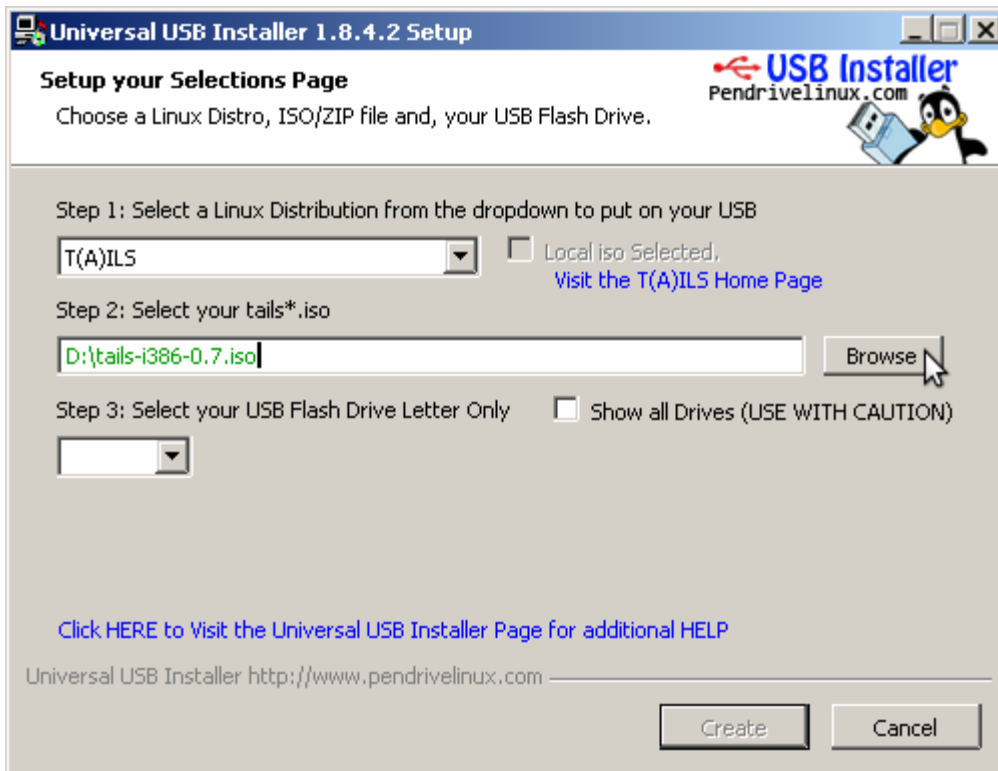
Read the license agreement and choose 'I Agree' to continue



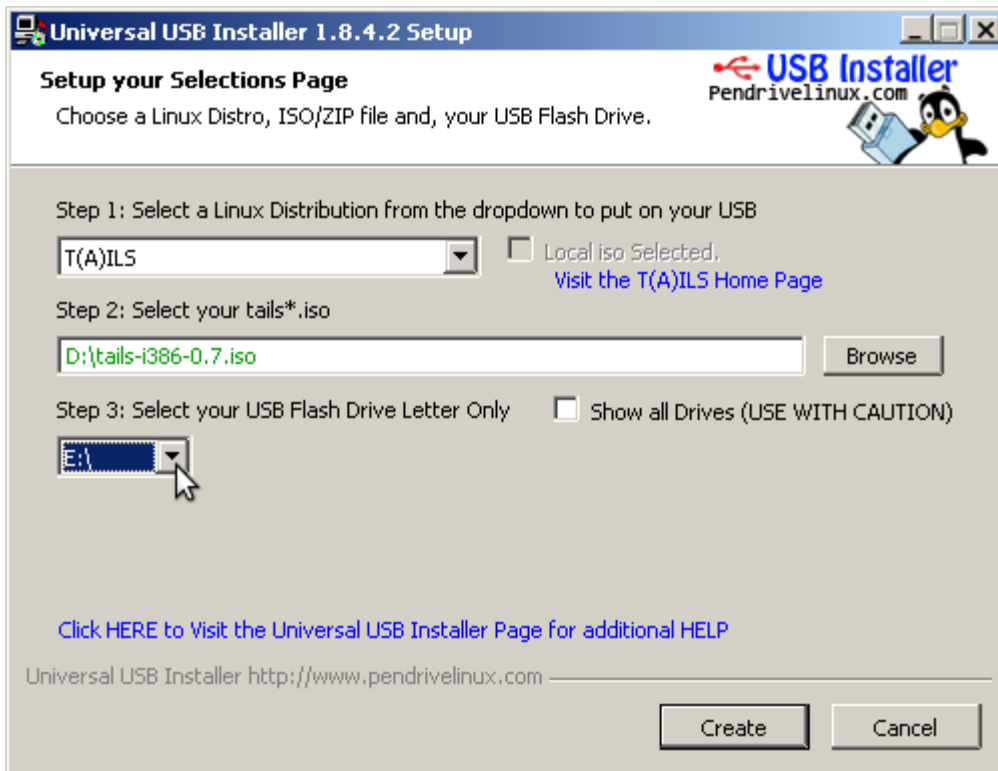
Select Tails from the dropdown list



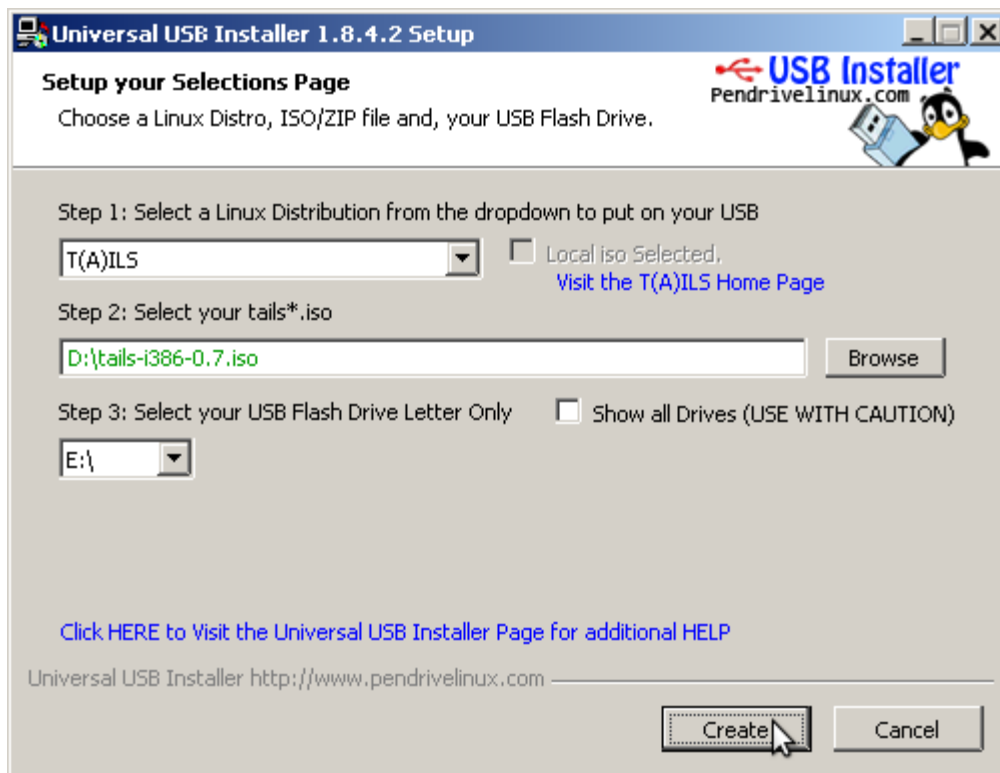
Click 'Browse' and open the downloaded ISO file



Choose the USB stick



Click 'Create'



Then safely remove the USB stick from the computer.

After the installation completes, you can [start Tails](#) from this new USB stick.

Manual Installation using Mac

Find out the device name of the USB stick

The device name should be something like `/dev/disk8`, `/dev/disk9`, etc.

If you are not sure about the exact device name, do the following:

1. Unplug the USB stick.
2. Open Terminal from Applications ► Utilities ► Terminal.app
3. Execute the following command:
4. `diskutil list`

This returns a list of all the current storage devices. For example:

```
$ diskutil list
/dev/disk0
#:  
0:      GUID_partition_scheme      *500.1 GB   disk0  
1:      EFI                        209.7 MB   disk0s1  
2:      Apple_HFS MacDrive          250.0 GB   disk0s2  
3:      EFI                        134.1 GB   disk0s3  
4:      Microsoft Basic Data BOOTCAMP 115.5 GB   disk0s4
```

1. Plug back the USB stick and run the same command as before:
2. `diskutil list`

A new device should appear in the list of storage devices. Check that the size of the device corresponds to the size of your USB stick.

```
$ diskutil list
/dev/disk0
#:  
0:      GUID_partition_scheme      *500.1 GB   disk0  
1:      EFI                        209.7 MB   disk0s1  
2:      Apple_HFS MacDrive          250.0 GB   disk0s2  
3:      EFI                        134.1 GB   disk0s3  
4:      Microsoft Basic Data BOOTCAMP 115.5 GB   disk0s4
/dev/disk1
#:  
0:      FDisk_partition_scheme      *4.0 GB     disk1  
1:      Apple_HFS Untitled 1          4.0 GB     disk1s1
```

In this example, the USB stick is 4.0 GB and the device name is `/dev/disk1`. Yours are probably different.

If you are not sure about the device name you should stop proceeding or **you risk overwriting any hard disk on the system.**

Unmount the USB stick

Execute the following command, replacing `[device]` with the device name found in step 1.

```
diskutil unmountDisk [device]
```

Do the copy

Execute the following command, replacing `[tails.iso]` by the path to the ISO image that you want to copy and `[device]` by the device name found in step 1. You can add `r` before `disk` to make the installation faster.

```
dd if=[tails.iso] of=[device] bs=16m && sync
```

You should get something like this:

```
dd if=tails-i386-1.3.iso of=/dev/rdisk9 bs=16m && sync
```

If you are not sure about the path to the ISO image or if you get a No such file or directory error, you can first type `dd if=` and then drag and drop the icon of the ISO image from a file browser onto Terminal. This should insert the correct path to the ISO image in Terminal. Then complete the command and execute it.

If you don't see any error message, Tails is being copied onto the USB stick. The whole process might take some time, generally a few minutes.

If you get a "Permission denied" error, try executing the command with `sudo`:

```
sudo dd if=[tails.iso] of=[device] bs=16m && sync
```

Be careful, if the device name is wrong you might be overwriting any hard disk on the system.

The installation is complete when the command prompt reappears.

Start Tails

After the installation completes, follow the instructions to [start Tails on Mac](#).

Notes

This method was successfully tested on the following hardware:

- MacBook Pro Model A1150 with OS X 10.6.8, 2006
- MacBook Pro Retina 15" Mid-2012 (aka MacBookPro10,1)

The method worked on some hardware but a bug in the video support prevented Tails to start successfully:

- MacBook Pro Retina with OS X 10.8.3, December 2012
- Macbook Pro model A1150

Note that Tails developers are in general not very knowledgeable about Mac. Any additional information is welcome.

Jolly Roger's Security Thread for Beginners

Last Updated 2014

By: Jolly Roger

Active Source

[http://bm26rwk32m7u7rec.onion/index.php?PHPSESSID=8i5jin3i1ufu6dhm7ned59jdm6&topic=2107.](http://bm26rwk32m7u7rec.onion/index.php?PHPSESSID=8i5jin3i1ufu6dhm7ned59jdm6&topic=2107)

0

INTRODUCTION TO SECURE COMMUNICATION - TOR, HTTPS, SSL

Greetings comrades.

Through my research I have put together some security measures that should be considered by everyone. The reason I put this together is mainly for the newbies of this forum. But if I can help anyone out, then I am grateful for this. I would like to start out by saying, if you are reading like, you are likely a Silk Road user. If this is the case, then the #1 thing you must be using to even access this form is **Tor**. Tor will provide you with a degree of anonymity by using an 128-bit AES (Advanced Encryption Standard). There has been some debate as to whether or not the NSA can crack this code, and the answer is likely yes. This is why, you should never send anything over Tor that you aren't comfortable sharing with the entire world unless you are using some sort of PGP encryption which we will talk about later.

Communication from your computer, to the internet relies on an entry node which basically "enters your computer" into the Tor network. This entry node communicates with your computer, this entry node knows your IP address. The entry node then passes your encrypted request onto the relay node. The relay node communicates with the entry node and the exit node but does not know your computer's IP address. The exit node, is where your request is decrypted and sent to the internet. The exit node does not know your computer's IP, only the IP of the relay node. Using this model of 3 nodes it makes it harder, but not impossible to correlate your request to your original IP address.

The problem comes obviously when you are entering plain text into TOR because anybody can set up an exit node. The FBI can set up an exit node, the NSA, or any other foreign government, or any malicious person who may want to steal your information. You should not be entering any sensitive data into any websites, especially when accessing them over TOR. If any of the nodes in the chain are compromised, and some likely are, and the people in charge of those compromised nodes have the computing power to decrypt your request, then you better hope it wasn't anything sensitive.

So what can we do to fix this? Well, luckily we are now having more and more servers that are offering something called Hidden services. You can easily recognize these services by the address **.onion**. These services offer what's called end-to-end encryption. What this does is take the power out of the compromised exit nodes and put them back in your hands. The web server of the hidden service now becomes your exit node, which means the website you are visiting is the one decrypting your message, not some random exit node ran by a potential attacker. Remember, the exit node has the key to decrypt your request. The exit node can see what you are sending in clear text once they decrypt it. So if you are entering your name and address into a field, the exit node has your information. If you are putting a credit card, a bank account, your real name, even your login information, then you are compromising your identity.

Another step you can take, is to only visit websites that use something called HTTP Secure. You can tell if the website you are visiting is using HTTP Secure by the prefix at the beginning of the address. If you see **https://** then your website is using HTTP Secure. What this does is encrypts your requests so that only the server can decrypt them, and not somebody eavesdropping on your communication such as a compromised Tor exit node. This is another form of end-to-end encryption. If

somebody were to intercept your request over HTTP Secure, they would see encrypted data and would have to work to decrypt it.

Another reason you want to use HTTPS whenever possible, is that malicious Tor nodes can damage or alter the contents passing through them in an insecure fashion and inject malware into the connection. This is particularly easier when you are sending requests in plain text, but HTTPS reduces this possibility. You must be made aware however, that HTTPS can also be currently cracked depending on the level of the key used to encrypt it. When you visit a website using HTTPS, you are encrypting your request using their public key and they are decrypting it using their private key. This is how cryptography works. A public key is provided to those who want to send an encrypted message and the only one who can decrypt is the one with the private key.

Unfortunately, many websites today are still using private keys that are only 1,024 bits long which in today's world are no longer enough. So you need to make sure you find out which level of encryption the website you are visiting uses, to make sure they are using at a minimum 2,048, if not 4,096 bits. Even doing all of this unfortunately is not enough, because we have another problem. What happens if the web server itself has become compromised? Maybe your TOR nodes are clean, maybe you have used HTTPS for all your requests, but the web server itself of the website you are visiting has been compromised. Well then all your requests are again, as good as plain text.

With that being said, this will conclude the first post in this series of the steps we can take to protect our privacy online, to remain anonymous and maintain our freedom.

PGP, TAILS, VIRTUAL BOX

So keep in mind that if you are a user of Silk Road, or any other form of activism, you never want to enter any identifying details about yourself online. Make it so that even if the NSA intercepted and decrypted, or compromised Silk Road that the only information they have against you is your username and password. How safe is that username and password? Does your password contain any identifying information? Is it the same password that you use for your personal email? Does it contain a name of somebody you know personally? Always keep all of these factors in mind.

Another step you must take, especially when communicating with other users on sites such as Silk Road is using PGP encryption. This is not always possible, such as in cases when you are logging into a website, filling out a form, logging into an email, etc.. Consider any type of information you enter into a website using plain text possibly compromised. Never put anything sensitive in any type of plain text format online. PGP comes into play because it uses a very strong method of encryption called cryptography. PGP stands for **Pretty Good Privacy**, and it is used for encrypting, decrypting and signing texts, e-mails, files, directories, and whole disk partitions and to increase the security of e-mail communications.

For the more technical users, it uses a serial combination of hashing, data compression, symmetric-key cryptography, and finally public-key cryptography. For the less technical users, the process of encrypting messages using PGP is as follows. You create a private key and a public key. The

public key is the key you give out to people you want to send you encrypted messages. Your private key, is kept privately by you. This private key is the only key that can unlock messages that were previously locked with your public key.

If you are still confused, think about it like this. Think about a public key that can go around locking boxes that are intended for you. Anyone can lock a box that is intended for you, but you are the only one with the key to unlock the box. Either if the person who sent you a message locked a box (message) with your public key, they themselves can not unlock it. Only the person possessing the private key can unlock it. If you wish to respond to this person, you must use their public key to encrypt the message you intend to send to them. And they themselves, use their own private key to decrypt the message you sent them.

If you are still with me, I am glad I haven't lost you yet. This is called cryptography and was designed so that anybody intercepting your message could not decrypt the message without your private key. Even if you yourself, lose your private key, there is no method of key recovery. You can consider that message locked forever. So how do you use PGP?

Well before we get to that, I want to introduce you to a **Live Operating System**, which makes using PGP encryption and decryption very easy. A live operating system is an operating system that you can run on top of your current operating system. So for example, if you are a Windows user, you have 2 choices. You can download the live operating system, burn it to a CD or DVD and then boot your computer from that DVD or CD. This will make sure your computer run as if you have this operating system installed on your computer. However, if you remove the CD or DVD and reboot, then your computer will boot as normal. You can also use a USB drive to perform this same feature.

Secondly, you can run this live operating system in what's called a Virtual Box. The benefits of this are that you can run Windows simultaneously as you run this other operating system and you can easily switch back and forth between them without rebooting the computer. Both methods have their pros and cons. The pros of running a live CD boot, are that reduce the risk of having your computer compromised by viruses, malware and keyloggers that rely on Windows vulnerabilities to run.

If you are going to run this OS from a Virtual Box, I suggest downloading Virtual Box from Oracle. Note the **https://**

<https://www.virtualbox.org/>

Next, the live operating system I would encourage you to use is **Tails**. Tails can be found at the following website.

<https://tails.boum.org/>

The reason I choose Tails, is because it has many of the security features that you require to stay anonymous already installed. Some users are not happy with Tails, but it really is a great operating system loaded with security features. Many I will talk about in this series on security including PGP encryption and decryption. Make sure you download the Tails ISO file from the official Tails website

and you can either load it into Virtual Box or burn it to a DVD or load it onto a USB and booting your computer from that drive.

There are plenty of tutorials on how to load Tails into Virtual Box, so I won't go into much detail other than, make sure you run Virtual Box and Tails from a USB drive or SD card. I would suggest a USB drive however for reasons I will explain later. But basically when Virtual Box runs directly on your hard drive, it creates a virtual hard drive that is used as a temporary hard drive while Tails is running. Once Tails is closed, this virtual drive is deleted, but it's not permanently deleted. As we know from the power of recovery tools, deleted files are easily recoverable with the right tools. I will talk about how to protect your files from data recovery tools in future posts but for now, just keep Virtual Box and Tails OFF of your hard drive, and load it either on a USB drive or SD card.

The same goes when booting your computer directly into Tails from a DVD or USB stick. Your hard drive will be used to store files used by Tails, so make sure any files that are saved or accessed using Tails are done from a USB stick or SD card, otherwise they will be recoverable. This is why I prefer using a Virtual Box and running both the Virtual Box and Tails inside of it, off of a USB stick. Keep as much as possible off of your actual hard drive. It is possible to shred files beyond recovery, but it's much easier to do this on a 16gb flash drive, then it is a 1 TB hard drive.

Next post we will get back on topic and start learning how to use PGP. The reason I have to take a detour to using Tails is because we will be using Tails for many of the features from here on out, including PGP.

PGP CONTINUED

Ok, so by now I am assuming you have Tails running. Let's learn how to use PGP within Tails. First thing you are going to want to do is create your own personal key, which consists of your public key that you can give out to people or post in your profiles online. As mentioned before, this is the key people use to encrypt messages to send to you. Your personal key also consists of your private key which you can use to decrypt messages that are encrypted using your PGP public key.

If you look up to the top right area, you will see a list of icons, and one of them looks like a clipboard. You need to click on that clipboard and click **Manage Keys**

Next click **File** -> New

Select PGP Key and click Continue

Fill out your full name (I suggest you use your online name, not your real name)

Optionally fill out an email and a comment as well.

Next, click Advanced Key Options.

Make sure Encryption type is set to RSA and set key strength to 4096.

Once you have done this, click Create and it will generate your key.

Once you have done this, you can view your personal key by clicking the tab **My Personal Keys**. You

have now created your personal key! To find your PGP public key, you right click on your personal key and click Copy and it will copy your PGP public key to your clipboard, in which you can paste anywhere you wish. A PGP public key will look something like this.

-----BEGIN PGP PUBLIC KEY BLOCK-----

```
mQINBFLWDcBEADEzn3mnLsezUDDAS5Q0Im1f6JdkI534WPuRIAN8pnuQsCSwUQU
hPEAgNCUNhxN4yCJ1mDt9xpXpX8QzsMlcofCHE9TMLAnHzbmXLLi+D8sPZpLpDN
6jElFvmBD4dvp5adimvRI8Ce49RpO345VUz8Ac0qLSmsv2u+kQviDQXZkrrxXHnA
lalvgDopXTISA9Sh7J3HHYYQazOZt9mfAjjuuRdaOqmAAAtEe9dl43nrx+nSd/fqH
13XvMKhqJhloJ02CBFFRbM86vtx5yiXqHZX438M9kbASqU0A2jAfRd+IZG5Z9gCI
W6FTTrror+F4i+bEdAuGTG1XFsQSgjtIG0vgYiTJ93C2MZxrLvNnJp0g2zD0URyk8
Y2ldyCDfIL10W9gNMqLmjD0z/f/os66wTJkflSGaU9ZsrKHUKFN5OSfOZtNqktWn
fCpY4bigkJ8U/5C8mtr9ZE3Tv+RV4rPY0hAOtZucnhlRmYKVFNjvbS0MjqA1188c
wzBNG0XcpCNTmM5UsSvXwnDoUaEMXe50Hikxdk3d+CJzqYnor72g/WmIDROCIxI6
2D9rJ2JuLpI9bQLM+KCbXJf3kUSvzszZGXL/AwmynvqlruaXqr5975sCdfqXVexx
1sxsLofOzE01xSDEJRWwHQPlxTKPZFnXD709Xumjdinjv1w4onLk04Z96wARAQAB
tC5Kb2xseSBSb2dlciAoVGhleSB3b3VsZCBsaXZlIGFuZCBkaWUgdW5kZXlgaXQp
iQI3BBMBCgAhBQJSy1g3AhsDBQsJCAcDBRUKCQgLBRyCAwEAAh4BAheAAAoJEPuh
6tSg81nyzNsP/2ayrAz4InCK/ZnyRnnsjSHIXMv7t2uDtBomA/0B6v/S6wHMNZX
G6+sYg41mfMuZEimgavNb0Uc2r6mI7UyWy5lp1Gd/D+all81X7bm5EBpvl1isPgJ
EqjehEdh9FQjrTiRiJafM1m254hIAaZ1RvAphI0tM2lpudk+tNKq+ivV8PpsN9TP
Omg5ZAu1lIKtG9k5vS9HAQ0grJ01TFMEjlifrf7eRyJ1+dmRJ+Xtoy2js8UwS+wM
Rrli3G39P2BfEZFQka3EmQ2JgN4pDWFoI0hODGhTba8Z0XSnVtabOTi1TOWIFmFu
yqA9bNtuOt3KhIC/O+mEATRsc/VPbTY+80kf45LwlDBfKO3PcOXSOg7ygibzEqXn
Ms/Rfe1kNEBeR9Wx2NMJSdxypqGij17CLJwNLC3KypTIQrhzy3YAndeDG4TadW2P
v/FJxhz+MX+s+9VeX2fGC0Fsfp8JbeWMAznp8Rf6O/tzEYW+pbLoLRPdi/DvFBZV
yWGPspzt3Qspm+BHbeW9iFjvCyvP2/DrKmQM7ABuRh/TMZr7uQ5na11L8rf3nzs
Al/ISul42xLzxG+h9mDixXd1Vh6rVGMbCjL7wO25TUneFo13U5J+klo1blQWV/DL
FZUwhh2utWNCMCtcdRW0HYa14Wdyy7H68WmsJqBWUsbyD9PZ2gSawBy7uQINBFL
WDcBEACg3IOme+sg0OZN349UYRr9/O6uW2vC5x9/azZrFNSNYh/LFJTt3XI/FsjN
gCj6NxRxbfdyLjL1gxSlJyFtclFGS0lCOGlz7lINvemkewjde/bHXChz2Ilalli
L2A6Z6w3fP4jIQCw8NoGGJ360WMkZVTDDakYYkb50BrZSx4TVLjrHfFuLMXTE255
gQrId02jYO6240EDihHITuiSwUQvHtXlOrHSohN83TD1I4H7iH/FLae9gYh4C/lx
VLkzLUqvpf72Q/xogCZAJI4WEMmWD6dXufvyvhCXQnbjiLuAdQas0ef/t652LPw/
vJFDSdmguw9PXWpv3vFoE13UNU//+nw3klGxaVWGvazXk8lFiDv9USgEGjcNn4zo
8HqIQrYz9/gyl3XojGV6L8ieCWpHSweqR3NxKJmWKWEG1wwnWPL8M+z6OwEvRdxV
spy+eG0Zs+6igbw3tk6gJ4cq5ehdlmD6py27AhRhIj7uLlZxmK3uFV19QjtX/Dyt
73ZNX16krXqufI0HAJRd1PwhITPctSviW3L2qKF2Pdak3j97A656EclnCcAyOUC/
mUNUDtXJik6uwFgFFn9/pnFr+acY7ppsWPG5rr7jRj+Lgjnjkckpkjo8jN1hZE17
CfJyrYrSqdglCclgTHteIEZdPfPUMnbbSoyeufkyEW1AolKatQARAQABiQIlfBBgB
CgAJBQJSy1g3AhsMAAoJEPuh6tSg81ny4nIP/2IVfODTp1n5xPEBZEUlgzcMNeh5
FTIS3J44g5a+OlkRVgHFtu7K/MUsftlUzkvMMa0sXllhKc6syxctoD7LAt9tbQh
62yEzijTliU2QFgWJSS6lftC2lyRouAns3KD6XouKTFUs/iOn/QpwhnM+Ya/SAg
c/oroM7SE/T4g+v6EeRCq7In/TMgc74j+25zUF1rVScenbZKkYezxqz33cXLwl7l
```

```
IUBcK2uNHDBUB5G853NR0OkBm5i+KC8vM3K1/MZ+P/IK0xOcTGXZH/A7GrEsl4FJ
nw5i6zJZb8gmDt44Tp/1Ujxnm5xhVWgnOQeSVSyiRsHQ/gTCL1PqsZhW7yulwL05
yxZgN+oYVx4pNtLJMigRjoCY9IKEmZhY75cWXXA19j14Wnxu8lrwwSk1WyzMQcjj
7onP4OEhbPuotqWqVAc0M/+MV5oMGIG0Qepy6XpZOCCpZw/p1rDrZSYP5eQMd/4x
LB7xch6GjbWsnKhA1wGdjdclBodixorVfCRn4s5jTgXx7wWz/opM4ix/CPAkify7
4Sf0BdJ5YtFILZc5StED4WC5pljJbdEWWsb9rn6egvFn7W/ZIDJAerS6Mt5LJGAh
Aude0Kz2HJwDtOBF4nXeTzRCK5BrBnCYPHAtO2aqfowirzjMTd9A/ADoPmlbIJAm
04mA6krRiH909Bnx
=Az2N
-----END PGP PUBLIC KEY BLOCK-----
```

Next, you are going to want to save the private key on a secondary USB drive or SD card. If you are running Tails from a USB drive, then you must use a separate drive to store your key on. If you are running Virtual Box, you want to **right click** on the icon in the bottom right corner that looks like a USB drive, and select your separate drive that you will be using to store your keys on. Again, never store your private keys on your hard drive, keep them OFF your computer.

To save your private key, you are going to right click on your personal key and click Properties. I know you probably saw where it says Export, but this is not what you want to do. Clicking export will ONLY export your public key and will not save your private key. If you lose your private key, you can never recover it even if you create another personal key using the exact same password. Each private key is unique to the time it was created and if lost, is lost forever. So once you have clicked **Properties**, go over to the tab **Details** and click **Export Complete Key**.

Once you have done this, you have saved your personal key for future use once you restart Tails. Remembering that Tails is not installed on your hard drive, so every time you restart Tails you lose all your keys. By saving your keys onto a USB drive or SD card, you can import your keys for use every time you restart it.

Next you are going to want to learn how to encrypt and decrypt messages using your key. Well, luckily for me, Tails has already made a tutorial on how to do this, so I will refer you to their webpage. But before I do that, I need to mention that you need to find somebody else's PGP public key, or you can practice by using your own. Needless to say, the way you import other people's keys into what's called your **key ring** is by loading them into a text file. You do this with the program called **gedit Text Editor**.

Click Applications -> Accessories -> gedit Text Editor and enter in someone's public key and hit save. Next you can return to your key program from the clipboard icon and click File -> Import and select that file. It will import that person's public key into your key ring. To add future public keys to your key ring, I suggest reopening the same file and just adding the next key below the previous key and each time you open that file it will load all keys within that file. This way you can keep all the PGP public keys together in one file and save it on your SD card or USB drive for future use.

Finally you can use the following 2 pages to learn how to encrypt and decrypt messages using PGP.

https://tails.boum.org/doc/encryption_and_privacy/gpgapplet/public-key_cryptography/index.en.html

https://tails.boum.org/doc/encryption_and_privacy/gpgapplet/decrypt_verify/index.en.html

Until next time. Have fun with your new found ability to communicate in PGP!

WHOLE DISK ENCRYPTION AND FILE SHREDDING

Welcome back again!

Now that we have PGP figured out, hopefully, I want to remind you that using PGP whenever possible, is very very very important. One of the pitfalls of Silk Road 1, is that some of the administrators, including Ross himself did not always communicate using PGP encryption. Once Ross was busted, they had access to his servers and his computers and anything that wasn't encrypted was wide open for them to look at. Most users on Silk Road 2 believe that Ross had stored personal information about some of Admins and Moderators on his computer in plain text that was used to make 3 more arrests of Silk Road users.

One of the reasons why I would suggest for you to store your PGP keys and other sensitive data on a SD card, is that if that day comes when you are compromised and you get a knock at your door, you have time to dispose of that SD card or USB drive quickly. Even better, if you have a micro SD card that plugs into an SD adapter, then you can snap it with your fingers or at the very least hide it. USBs would need to be smashed into pieces and it might not be easy to do this in the heat of the moment, so do what you feel best about. But always prepare for the day they might come for you.

But our next topic brings us to something called Whole Disk Encryption or Full Disk Encryption. From here on out I will refer to it as FDE (Full Disk Encryption). Tails has a FDE feature built into it, which is another reason why I encourage the use of Tails. It has many of these features to protect you. Essentially FDE will protect your drive, whether SD or USB from the people who may come for you one day. The method in which it does this is it formats your drive and rewrites the file system in an encrypted fashion so that it can be only be accessed by someone who has the pass phrase.

If you lose your passphrase, just like in PGP, there is no recovery. Your only choice is to format the drive and start over again. So make sure you remember it! And please for the love of God, Allah, Buddah, etc... don't store the passphrase on your hard drive somewhere. The tutorial on how to do this is located at the following webpage.

https://tails.boum.org/doc/encryption_and_privacy/encrypted_volumes/index.en.html

Again, always prepare for the day they come knocking, encrypt everything. Use PGP when communicating with others and always shred your files when finished with them. Which brings me to my next topic. **File shredding**.

File shredding is extremely important and here is why. If you delete a file from your computer, you are only deleting where it is located on the drive. It is still on the actual drive, just it's location data has

been removed. If you take a file recovery tool you can recover virtually any file that you have recently removed. File shredding combats this by overwriting files instead. The idea is that instead of removing the file's location, you need to overwrite the file with random data so that it becomes unrecoverable.

There are a lot of debate happening on whether you can overwrite a file once, or if you need to do it multiple times. Supposedly the NSA recommends 3 times, supposedly the Department of Defense recommends 7 times, and an old paper by a man named Peter Gutmann written in the 90's recommended 35 times. Needless to say, I personally think between 3-7 times is sufficient, and several people out there believe 1 time will get the job done.

The reasoning behind this is that some people believe the drive may miss some files the first time it over writes them and to be more complete, you should do multiple passes. Do what you feel most comfortable with, but I even think 3 passes would be sufficient, although it wouldn't hurt every now and then to run 7 passes and just leave it overnight.

The programs that can do file shredding are ones you will want to run from Windows or whatever operating system your computer is running. These programs can delete your files from your Recycling Bin, delete your temporary internet files and even Wipe your free disk space to make sure everything gets cleaned up. You always need to think, did I have any sensitive material on my hard drive? If so, maybe I need to shred my free disk space. When emptying your Recycle Bin, you should always use a shredder. When only deleting under 1gb at a time, you can easily do 7 passes pretty quickly.

To put this in perspective, the leader of a group called LulzSec name Topiary has been banned as part of his sentence from using any type of file shredding applications so that if the FBI wants to check up on him, they can. File shredding keeps your deleted files actually deleted.

Here are some file shredding applications you can use.

<http://www.dban.org/>

<http://www.fileshreder.org/>

<https://www.piriform.com/ccleaner>

Next we're going to talk about removing harmful metadata from files, and some other topics as well.

JAVASCRIPT VULNERABILITIES AND REMOVING PERSONAL METADATA FROM FILES

Welcome Back.

Before I get into removing harmful meta data from your files, I want to talk about another vulnerability to our browsing capabilities called Javascript.

In mid 2013, a person in Ireland was providing hosting to people that hosted hidden services including a secure email platform called Tor Mail. Unfortunately, they busted him on an unrelated charge relating to child pornography and seized all his servers. Whether or not he was related to child

porn or not, is unknown to me, or it could be a silly charge the feds slapped him with but either way, the feds ended up injecting malicious Javascript into his servers so that when users would visit certain sites, this malicious code would execute on their computers and reveal information about their computers to the feds. I suggest you read the following article to learn more about this.

<https://openwatch.net/i/200/>

With that being said, you may want to disable Javascript in your browsers, especially when visiting certain websites like Silk Road that may become compromised one day. Many users refuse to visit the original Silk Road website and forums with Javascript enabled because the feds likely injected it with malicious Javascript to identify users.

In Tails, the browser is called Iceweasel and when Tor is ran in Windows, it uses Firefox. Both browsers can disable Javascript using the exact same method. Open up a Window and type the following command in the address bar, "about:config" and click the button that says "I'll be careful, I promise."

This will bring up a bunch of settings including a search bar at the top. Enter javascript in the search bar and look for the following two entries, "javascript.enabled" and "browser.urlbar.filter.javascript". Right click on these and click "Toggle" and you will see the Value changed to false. If you want to enable Javascript again, just click Toggle again and you will see the value change back to true.

Again, remember that every time you restart Tails you will have to do this again, so get into a habit of doing this every time. You never know when your favorite website could become compromised.

Moving onto meta data. There is a bit of a famous story about an online hacker named w0rmer that would take pictures of his girlfriend and post them online after he would deface a webpage. What he either forgot, or didn't know was that photos taken with the iPhone and other smart phones save the GPS coordinates of where the picture was taken and store it in the meta data of the picture. Check out this article below.

<https://encyclopediadrastica.es/W0rmer>

You need to remove this meta data! Otherwise you could end up in federal prison with w0rmer. Luckily Tails has a solution for this! See why I love Tails?

Applications -> Accessories -> Metadata Anonymisation Toolkit

Please get a more clear idea of how this works by reading the following page.

<https://mat.boum.org/>

Please note the currently supported formats. In terms of pictures, jpg, jpeg and png. But unfortunately MAT is not perfect and I wouldn't solely rely on it, so a better idea would be to never

upload pictures of yourself or your significant other online, especially bragging about a hack you committed. Please read the site provided above for more information.

GENERAL SECURITY PRECAUTIONS WHEN POSTING ONLINE, LEARN FROM OTHERS' MISTAKES

Next I want to talk about good practices when using TOR, Tails and other hidden services.

First of all, it is highly recommended that you use multiple identities online for different things. Perhaps if you are a buyer and a seller on Silk Road, you may want to have separate logins for this. And then possibly a third login for the forums. Then maybe you want to be part of another marketplace, then you might want a fourth login.

Well, Tails has another good program offered by Tails is called KeePassX. When you have multiple logins, it is hard to keep track of them all, so it might be a better idea to keep them all in 1 document that is encrypted with a strong password. KeePassX can help you with this.

https://tails.boum.org/doc/encryption_and_privacy/manage_passwords/index.en.html

You never want to use nicknames or locations, or anything else that is related to yourself online when you post or create usernames. And another thing you need to adopt are new ways of conducting yourself. If you are generally a messy typer, who makes the same grammar mistakes, or the same spelling mistakes all the time, this can be used to identify you. Always proof read anything you post publicly, or privately because the feds will always find ways to correlate things to you.

With Ross Ulbricht, they found an old post he posted on a forum when he first started Silk Road asking people if they had heard of a marketplace called Silk Road. Obviously this is an old trick used by people trying to spread awareness about a new project of theirs. Later he identified himself by saying he was looking for programmers and gave out his private email address on the same forum under the same name.

But if you always misspell the same words, if you always use the same slang terms, capitalize the same words, use a certain amount of periods after an etc.... or always use the same number of !!!!! then all of these things give them reasonable suspicion and it becomes easier to tie things to you. Once they have you under their radar, like they had Ross, it only took a few slip ups and he was theirs. Remember, you only have to make one mistake. So talking about your local election is a really dumb idea, get it?

Think about the time you use your computer. Is it easy to correlate your timezone based on the time you go online? Or is it more random? Do you have patterns that are predictable? Always think about these things when you post online. Always think about what type of personality you are putting out there about your online name.

Expect that every single word you type online is being read by the Feds. To them, this is much easier than tracking drug lords on the streets. They sit in an office and read forum posts and try and make

connections. Don't underestimate the feds. Always treat everything as compromised, always treat everybody as compromised and don't ever think anybody will ever go to jail for you. If somebody can avoid 10-20 years by ratting you out, they will do it in a heart beat.

The perfect example is Sabu from LulzSec. After he was busted and facing 112 years in jail, they made him a deal to help them rat out his friends and he ended up getting many of his "friends" arrested. Even people who are your friends will turn their backs on you when it comes down to their freedom.

EXIF DATA

I forgot to mention above when talking about metadata, that when it comes to photos, there is another risk involved called EXIF data, this is another form of meta data specifically related to images and may not be properly removed by Metadata Anonymisation Toolkit mentioned before.

EXIF data stands for **Exchangeable image file format** and affects JPG, JPEF, TIF and WAV files. A photo taken with a GPS-enabled camera can reveal the exact location and time it was taken, and the unique ID number of the device - this is all done by default - often without the user's knowledge.

In December 2012, anti-virus programmer John McAfee was arrested in Guatemala while fleeing from alleged persecution in Belize, which shares a border. Vice magazine had published an exclusive interview with McAfee "on the run" that included a photo of McAfee with a Vice reporter taken with a phone that had geotagged the image. The photo's metadata included GPS coordinates locating McAfee in Guatemala, and he was captured two days later.

To avoid this, only take photos that use PNG because it does not store EXIF data. To check if your photo has any revealing EXIF data attached to it, check out this site.

<http://www.viewexifdata.com/>

or you can download a tool by doing a quick search online to see what EXIF data may be contained in your photos before you upload them. Be very careful with any files that you upload online, because you never know what type of harmful data could be attached in them. It helps to use Tails, but always consider everything you put online as a potential piece of evidence to be used against you and always prepare for the day the feds come to your door.

RETAINING A LAWYER, HOW TO HANDLE GETTING CAUGHT OR INTERROGATED

Next entry into the series on security is how to handle getting caught.

Let us face it. We are all human and we make mistakes. Unfortunately, you only need to make one mistake, and the Law Enforcement, commonly referred to as LE on these forums can bust you. Maybe they will wait for you to do something more serious before they nab you, but if you slip up and they feel you are worth going after, you can expect them to get you no matter where you live, with rare exception.

The first thing I want to do is link you to another thread I just came across on these forums.

<https://silkroad5v7dywlc.onion/index.php?topic=13093.0>

The main question is, should I keep an emergency lawyer fund on hand? And how much should it be. The response I think was most appropriate for this question was the following.

Quote from: VanillaRoyale on January 02, 2014, 05:33:49 am

Give your lawyer 50k and put him on a retainer.

Don't have a emergency fund 'stash' lying around if that is what you mean.... you should already have your lawyer paid + plus extra in case he needs to post bond for you and they seize the majority of your drug funds.

Once you get arrested by LE, they can seize your money based on the assumption that it is drug related. So you need to have a lawyer paid for ahead of time. That way, in the unfortunate case that you get a visit from the feds, you have a lawyer ready to go. The agreed upon amount was around \$50,000.

Next I want to talk to you about what to do in case you get interrogated by LE. There is a great thread about this.

<https://silkroad5v7dywlc.onion/index.php?topic=4461.0>

The take homes from this thread are basically. Keep your moouth shut. The feds are going to try all types of tactics on you to get you to admit to guilt of the crimes you are being accused of. They will likely use the good cop, bad cop on you. First they will tell you that they want to help you, and that they are after the big guys. They just need your help to put away the big guys. Do not listen to this, I have never cooperated with a good cop LE and have it end up working in my favor. Once you admit to being guilty, you can kiss your freedom good bye.

Secondly, if you refuse to cooperate, their attitude will change to bad cop. They will say, "OK fine, you do not want to cooperate? I tried to help but now you are going to be in a lot of trouble. Do you have any idea what kind of charges you are facing? You are going away for a long time unless you start talking."

They are going to try and scare you into admitting guilt. Again, keep your mouth shut and continue to ask for a lawyer, hopefully the one you put on a \$50,000 retainer prior to this happening. Never speak without a lawyer present and never do anything you do not have to do legally. If you have the right to remain silent, then exercise that right. I know there are some circumstances in which you do not have that right, but unless that is the case, you are better off staying quiet.

Third, drop the attitude. Do not argue with the cops about having nothing on you, or something for that matter. Act scared, anxious and confused. Act like you have no idea what is going on and that you

are scared for your life. Tell the cops they are scaring you and you want to see your lawyer because you do not know what this is about. They need evidence, and solid evidence at that, to charge you with a crime.

They are going to try and correlate posts you made on forums, phone numbers you called, perhaps a package shipped to your home, all forms of communication, bank transfers, and so forth, until they can find a way to link you to the crime you are being accused of. But the biggest piece of evidence will always be your willingness to admit your guilt for a lesser sentence.

When Sabu found that he was facing 112 years in federal prison, he quickly spilled everything and started working for the feds. Again, talk to your lawyer, find out the evidence against you and only answers questions your lawyer advises you to answer, and answer them in a way your lawyer advises you to answer them.

Try and be as honest as possible with your lawyer. Your lawyer can not and will not share any admittance of guilt you have with the prosecutors or LE, this is called Attorney-client privilege. Please note there are a few instances where this does not apply.

https://en.wikipedia.org/wiki/Attorney%E2%80%93client_privilege#When_the_privilege_may_not_apply

COMBINING TOR WITH A VPN

Welcome back readers!

Today I want to talk about a greatly debated topic.

Should I use a VPN with TOR?

Should I use TOR to connect to a VPN, or use a VPN to connect to TOR?

Let me say first of all, that when you are browsing the internet without TOR, you should probably be using a VPN regardless of whether or not you are using TOR. And make sure that the VPN uses some form of encryption as well. For those of you who are very beginner, think about when you connect to a public wifi network at a coffee shop, or an airport and you get all these warnings that your requests sent over this network are vulnerable.

All networks, but especially public wifi networks are vulnerable to traffic analysis. Put this together with the fact that some internet service providers monitor your activity to some level, and you can see why it might be a good idea to always use an encrypted method of using the internet. At the very least to protect your personal information when you are entering credit cards, usernames and passwords, as well as other personal data online. Again, especially if you are using a public wifi network.

Choosing a VPN that uses at least 128 bit encryption like TOR is good practice, and will stop the majority of eavesdroppers. But if you can get 256 bit encryption, you are even safer. Before we get into whether or not we should be using a VPN together with TOR, I want to give you a few warnings

regarding how you should be using a VPN.

If you are going to be using a VPN for any type of freedom fighting, make damn sure that your VPN does not keep logs. This is actually a lot harder than you might think. Many VPN providers will claim to not keep logs of your activity in order to gain you as a customer, because they have to compete with the other providers out there. Customers are going to trend towards providers who offer no identifying data retention. Unfortunately, this claim of theirs is not always the real case and I will give you an example.

There is a well known VPN provider named HideMyAss that previously claimed not to keep logs of its users. Unfortunately, when met with a court order from their government in the UK, they handed over evidence of a suspected hacker from an internet group LulzSec which helped lead to his arrest. The story can be found below.

http://www.theregister.co.uk/2011/09/26/hidemyass_lulzsec_controversy/

One of the take home quotes from this article is the following.

Quote

We are not intimidated by the US government as some are claiming, we are simply complying with our countries legal system **to avoid being potentially shut down and prosecuted ourselves.**

A very smart man that goes by the online handle The Grugq, said when doing your freedom fighting online that nobody is going to go to jail for you, and he is 100% correct. When it comes down to it, no VPN provider is going to risk jail to protect a \$20 a month subscriber. No matter how tough they sound, no matter how much they claim to care about protecting their customers, when faced with a choice to give you up or go to jail, they will always choose freedom.

Another thing to consider however, is using a VPN does hide your internet activity from your internet service provider. It can also hide the fact that you are using TOR, which may flag some suspicion when the feds start asking ISPs to provide data about their users. This may or may not be relevant, since many people use TOR and you can argue there are many legitimate reasons to use TOR and nothing suspicious about TOR. But it is just another factor to arouse suspicion that may or may not come into play and should be considered.

If you choose to use TOR over a VPN, the benefits are that you would be again, hiding from your ISP the fact that you are using TOR. Also, your VPN would only be able to see that you are connecting to TOR nodes and that you are sending encrypted data. The VPN would not be able to see what data you are sending over TOR unless they decrypted it, because remember, all information relayed over TOR is encrypted.

The downsides of course, as mentioned are that VPN providers may or may not log everything that you do in the form of meta data or even content if they have the storage capacity, and keep those logs on hand for a long time. In this case, it is no better than connecting to TOR through an ISP. Another thing to mention to those who will use VPNs when not using TOR, but also use VPNs when using TOR is

remember when you are, and are not connected to your VPN. Sometimes VPNs can unexpectedly drop connections and you may not even be aware of it. If the reason you are using a VPN is to hide TOR activity from your ISP, then if your VPN drops, your ISP will start seeing your TOR traffic instead.

Or, maybe you forget that you are connected to your VPN and end up punching in your address on Google Maps to find directions somewhere. Well guess what Google does with all data entered into their system? They keep it. And they likely keep it indefinitely. So if one day the NSA identifies you on the TOR network by occupying a large number of nodes and using traffic analysis to identify you based on statistical analysis, it will link them to your VPN IP address.

At this point, they will likely ask the VPN to turn over data on their users, but if the VPN refuses to comply because they are not subject to US law, or the laws of other countries, they may check some of the big surveillance websites out there to see if you slipped up and used that IP address for anything else online. They will check logs from Google, Yahoo, Facebook, Twitter, Netflix and other big data collection companies to see who has been using that IP address to connect to their servers.

If you accidentally punched in your address on Google when connected to that VPN, you are now a suspect. So always keep things like this in mind. Just because you are covered behind a VPN does not mean you are not traceable by human error. The benefits of TOR, are that you get a new identity every time you connect. This may or may not be the case with your VPN, so please check and make sure.

Next post we will talk about the advantages and disadvantages of using TOR to connect to a VPN.
[/quote]

COMBINING TOR WITH A VPN CONTINUED

Ok, now let us talk about why you may want to connect to a VPN over TOR.

The data flow would look like this. You -> Tor -> VPN -> Internet

The benefits of doing that are as follows. You are more anonymous to your VPN in case they happen to keep logs, or if you do something using the VPN that you are not supposed to and a website or server grabs your VPN IP address. In the case of this happening, even if the VPN manages to keep logs of everything you do, they can only identify you as an anonymous TOR user as long as you did not purchase the service like an idiot with your credit card or Paypal account. If you use Bitcoin, and made sure the the Bitcoin trail is not easily traceable you should be okay. Some websites block TOR users from connecting to their websites or servers, by using your VPN to appear as the exit node, you are hiding your TOR activity from the website you are visiting and hopefully bypassing their filters.

Another advantage, is that if your VPN connection does drop, your fall back will be your TOR IP address instead of your real IP address. And finally, if you are passing through a compromised TOR exit node, your information will remain encrypted through the VPN's encryption protocol until it reaches the exit node of the VPN. This is a good thing if you are passing through a compromised exit node, but do not forget that the VPN could be logging everything you are doing anyways. **Do not trust anybody who has access to your unencrypted data!**

A few of the downsides of doing things this way, as mentioned in the previous post are that your ISP knows you are using TOR, when and for how long. This may or may not matter to you, but it is just something to consider. Second, you will be unable to visit hidden services websites. Remember those **.onion** sites we talked about in the beginning? You need to be connected to the TOR network to visit those hidden service websites.

But I am connected to TOR aren't I? Yes you are, but your final method of communicating with the internet does not come from the TOR network, it comes from your VPN. And your VPN is likely not configured for TOR. In order for you to be able to connect to a hidden services, you must either be connected directly to TOR, or use a VPN to connect to TOR. TOR must be your final node of connectivity in order to visit onion websites.

The choice is ultimately up to you, and every person in every state, province and country will have different reasons for wanting to do VPN to TOR or TOR to VPN, or just TOR, or just VPN. Whatever choice you make, please keep all the things mentioned in this post and the previous post in mind. None of these methods will save you if you enter anything identifying about yourself online. Do not log into your Facebook account using your VPN. Do not check your email or search a nearby address on Google using your VPN. In fact, stay away from Google altogether unless absolutely necessary.

There are two other search engines out now that do not store information about their users.

#1 - DuckDuckGo. They have both a clearnet URL and a hidden services URL for both types of users.

<https://www.duckduckgo.com>

<http://3g2upl4pq6kufc4m.onion/> - Please note the hidden services mirror is not HTTPS

#2 - StartPage. This server also does not store any information about its users.

<https://www.startpage.com>

Before we move on, I want to go back to how to choose a good VPN. When looking for a VPN provider, you will most likely come across two protocols to choose from. Find out which one your VPN provider is using before you sign up with them. PPTP and OpenVPN. At this time, I am going to highly recommend that you avoid PPTP and stick with OpenVPN providers. Check out this site for a quick comparison.

<http://www.goldenfrog.com/vyprvpn/openvpn-vs-pptp>

As you can see, PPTP uses a weaker encryption, 128-bit versus 160-bit to 256-bit for OpenVPN. It offers basic security versus a high level of security using something called digital certificates. This is basically a way to make sure they data coming in is sent from your VPN provider and not injected by some malicious third party because the incoming and outgoing data are signed using specially obtained certificates, similar to showing your ID to get into a a restricted area.

The only downside is that setting up OpenVPN can be a little challenging for the less technical users, but there are plenty of great tutorials online to set up OpenVPN providers and your VPN provider itself

will likely help you get set up as well. PPTP has been abandoned by those who demand the highest level of security, so I would recommend to avoid it. A third option for VPN providers is L2TP/IPsec, but many users now believe it has also been compromised by the NSA due to its weaker levels of encryption and should be avoided as well. **Stick with OpenVPN.**

Lastly, if you want to know how to connect to TOR over a VPN. If you are using OpenVPN like I recommended, then you it is really quite simple. Make sure you are connected to your VPN, check your IP address to on any website such as **WhatIsMyIpAddress.com** to make sure it has changed. Then, open TOR or open TAILS and start using TOR and you are now connected to TOR over a VPN.

Connecting to a VPN over TOR is a more tricky and currently above my skill set since OpenVPN reconfigures your network routes so Tor can't be running on the same host. As soon as I figure it out, I will post a tutorial, and if anybody can share an easy way to connect a VPN over TOR, then please share it with this thread.

UPDATE

A method of connecting to a VPN over TOR has been added to this thread but is currently only able to be used by Windows users. You can read it about it at the link below.

CONNECTING TOR -> VPN FOR WINDOWS USERS

After a long search, I have found a way you can connect TOR -> VPN. It is not perfect, and some might not agree with doing things this way, but it works and I am giving it to you as an option, but it only works for Windows users at this time.

If you look back at my previous posts regarding combining VPN and TOR then you will find the reasons why you would want to do so, and some of the reasons why you might not want to do it. But I was unable to provide you with a way to connect to a VPN using TOR so that the VPN does not know who you are. When it comes to TOR -> VPN, if you cannot trust your VPN, which you rarely should, then keeping your identity anonymous from your VPN is a good idea. Also, with more and more people using TOR, but with only around 4000 TOR exit nodes, many of the exit node IP addresses are being flagged as spammers on popular websites and limiting the usage of well meaning TOR users to post on message boards like Stack Exchange and so forth.

The way that I found you can do TOR -> VPN is by using a virtual machine, preferably Virtual Box and running another instance of Windows, preferably one that uses less memory than your current version. You also want to run TOR Expert and Tortilla on your host OS. I talk about how to do this in previous posts. Next set your Virtual Box to route all it's network traffic through Tortilla (bridge adapter), which routes it all through TOR. Currently Tortilla is only supported by Windows, which is why this option is only available to Windows users at this time. Doing this also makes it easier to do things like watch videos on YouTube.

Now that you have your Windows Virtual Machine running on TOR, you can install a VPN of your choice, preferably one using OpenVPN on your Windows Guest OS and connect to it. Check your IP

address before connecting and after and you should see a different IP address. If all went well, you now have a virtual machine running TOR -> VPN. Then if you want to add another layer, you can download TOR browser bundle onto your virtual machine and run that as well giving you TOR -> VPN -> TOR for another layer of security. Also you have the option using this method to use a VPN on your host OS, then Tor Expert with Tortilla, then another VPN on your guest OS, then TOR browser, giving you VPN -> TOR -> VPN -> TOR.

I am not advocating any which method, you need to make that decision on your own, I am just giving you the knowledge necessary to make an informed decision and you can ultimately choose which method you feel most comfortable with. Sometimes doing TOR -> VPN is necessary because of the spam filter reasons I mentioned above and other times having TOR as your last node to the internet is necessary like when accessing the onion network. It is completely up to you and I know that we are trying to shy away from Windows usage because of all the exploits and other reasons spoken about in the previous posts, but if you have no other way of staying anonymous from your VPN than this, then I think it is a good compromise until we have something like Tortilla that is compatible with Linux distributions.

TRACKING COOKIES

Next time I want to talk about is something that most people completely forget about. **Tracking Cookies.**

A recent article explains how the NSA uses things like Google Ads and other tracking cookies to identify users over TOR when doing so by other means is not possible.

<http://www.washingtonpost.com/blogs/the-switch/wp/2013/12/10/nsa-uses-google-cookies-to-pinpoint-targets-for-hacking/>

For those of you who do not know what I am talking about, let me ask you this. Have you ever noticed that certain ads seem to follow you around from website to website? Perhaps something you searched for on Google or Yahoo is now showing up in ads on other pages? This was originally designed to market things to you based on your preferences by installing tracking cookies into your browser.

Luckily TOR clears its cookies every time you restart the browser, and yes Tails does too, but that does not mean you are not vulnerable within the same TOR session. What I mean by this is, let us say you went and did some freedom fighting on a forum somewhere and then after, using the same Tor session, visited another website with Google Ads on it. Then you went to another site with more Google Ads on it. You would be surprised how many sites now have Google Ads on them, by the way.

Google can use these tracking cookies to learn about your browsing behavior. Your search terms, your preferred sites, and so forth. Some people are even stupid enough to use the same TOR IP address and go check their Facebook news feed or their email. Guess who is in bed with the feds? Google, Yahoo, Facebook, MSN, and all of their email providers as well. Remember, when you start leaving patterns behind, they will start looking for similarities that start with just a suspicion.

Perhaps they correlated the freedom fighting forum posts with you because you logged into your email, and now they start noticing that you always misspell the same words, make the same grammar mistakes, the same slang terms. Perhaps you visited a website belonging to somebody local to you with Google Ads on it. It is not entirely sure how they are able to use these tracking cookies to identify you, but the point is, they keep everything. And if you happen to do something stupid like Google a local restaurant or what movies are playing in your local area on the same IP address that you did something you should not have earlier on, then Google can put 2 and 2 together.

Once they are on your trail, you are screwed. So do not give them anything to correlate to you, ever! So then you might ask, can not I just disable cookies all together? Yes you could, but, cookies are required for things like login sessions. Without cookies, you are unable to maintain a state of being logged in on certain websites, because they use that cookie ID to identify the session on the server. Again, you can certainly disable cookies, but you will not be able to maintain a login anywhere.

LEARNING FROM OTHERS' MISTAKES. LIBERTAS, DPR, SABU, LULZSEC

A little change of pace for this next post. I want to talk about one of our fallen moderators Libertas.

It has finally been confirmed, what we all were hoping for that **Libertas**, one of the 3 arrested moderators was released on bail recently according to an article.

<http://techcrunch.com/2014/01/07/the-silk-roads-libertas-is-free-to-the-annoyance-of-us-authorities/>

Quote

The Silk Road moderator Gary Davis, aka Libertas, is officially free on bail and awaiting an extradition hearing on February 13.

The FBI flew to Ireland that night for the express purpose of taking Davis into custody and interrogating him in Ireland, with regard to his position and functions “being a moderator on a website allowing transactions to facilitate the sale of drugs online.”

So as you can see, just because Libertas was a moderator on the site, he is being charged with allowing transactions to facilitate sales of drugs. He is basically being charged as a drug dealer.

Quote

However, Davis was found in possession of illicit substances which could result in a minimum sentence.

He unfortunately was found with drugs on him at the time of his arrest, which made things much easier to keep him in custody. And it turns out that the alleged former owner of Silk Road, Ross Ulbricht is fully complying with law enforcement to attempt to identify senior vendors on Silk Road. According to the article, Ross communicated with the vendors frequently and likely in plain text (is my guess).

The reason I bring this up, is that we need to remind every user on here of the mistakes that were

made by Ross, and the other three moderators so that we can hopefully learn from them. We need to avoid these types of mistakes, never ever EVER give anybody any personal information about yourself online. The story goes, that Ross required moderators to give him copies of their IDs in order to become moderators of Silk Road, and he likely kept a record of these on his computer. Unfortunately, these are now in the hands of the FBI and 3 moderators have been arrested as a result since. And now, according to the article, they are after senior vendors as well.

A few take homes are; Always use PGP encryption in all your communications, which unfortunately in this case would not have mattered because Ross ended up giving up his private keys to the feds. But it is still another hurdle in their way to protect you from them taking away your freedom. Never give out any personal information to anybody online about yourself. Never put your trust in somebody else's hands, because at the end of the day, nobody is going to go to jail for you. Ross found an opportunity to possibly reduce his sentence and he is fully taking advantage of the opportunity.

This exact same scenario happened with Sabu from LulzSec was threatened with 112 years in prison, he quickly turned on all his friends and worked with the feds to get them all locked up to help reduce his sentence. Sabu has 2 kids and obviously decided he would rather snitch out his friends and have a chance at being a father rather than spend the rest of his life locked up in jail. Again, **nobody is going to go to jail for you.**

HOW FAR WILL LAW ENFORCEMENT GO?

Today we are going to talk about the lengths that law enforcement (LE) will go to try and catch you slipping.

The thread that inspired this post was the following SR thread.

<http://silkroad5v7dywlc.onion/index.php?topic=8788.0>

The first question is, can LE ship drugs to buyers to try and set them up for drug charges? Let us just say, that they have done it to a Silk Road user before who went by the name of Flush aka Chronicpain aka Curtis Green

<http://www.usatoday.com/story/news/nation/2013/11/07/vendor-administrator-plead-guilty-in-silk-road-case/3469751/>

Quote

In April 2012, a DEA undercover agent in Maryland posing as a drug smuggler began communicating with "Dread Pirate Roberts" on Silk Road about selling a large amount of illegal drugs. "Dread Pirate Roberts" instructed [Curtis] Green to help the smuggler find a drug dealer who could buy a large amount of drugs, court papers say. Green found a buyer and agreed to act as the middleman for a \$27,000 sale of a kilogram of cocaine. Green gave the DEA agent his address.

An undercover U.S. Postal Service inspector delivered the cocaine to Green's house in Utah on Jan. 17.

So as you can see, whether you view it as entrapment or not, once they have evidence against you, they will eventually figure out a way to get something on you and bust you for it like they did to Curtis Green.

The Secret Service posed as a vendor for fake IDs online for 5 years and actually shipped fake IDs that they made to buyers on an online Russian forum.

<http://www.tested.com/tech/456882-how-secret-service-sold-fake-ids-catch-identity-crooks/>

Quote

The US Government's "Operation Open Market" resulted in indictments against 55 defendants. According to Wired, Special Agent Mike Adams shipped out more than 125 fake IDs over about five years of activity while going by the username Celtic. Amazingly, the entire scheme started when the government arrested the real Celtic, a Nevada man who got caught shopping at a Whole Foods where he'd previously used a fake credit card.

Law enforcement discovered counterfeiting equipment among his possessions and learned about his online activities. Adams assumed his online identity and even improved Celtic's cred, shipping near-flawless IDs and becoming a trusted seller on Carder.ru.

As you can see in this article, the Secret Service again sold illegal items to people online in order to bust them. Several of the buyers used their real addresses and sent real photos of themselves to this officer to have their IDs made, resulting in being arrested by the feds.

And in this particular case, the feds charged all the defendants under something called the RICO act.

Quote

"The main indictment is noteworthy because, in addition to the usual mix of credit card fraud and false identification charges, the 39 defendants have been charged under the mob-busting RICO act – a first for a cybercrime prosecution.

Enacted in 1970 to help the FBI crack down on the mafia, the Racketeer Influenced and Corrupt Organizations Act **lets the feds hold every member of a criminal organization individually responsible for the actions of the group as a whole**. The losses collectively inflicted by the Carder.su members are easily enough to give every RICO defendant 20 years in prison."

When you commit crimes online, especially in an online community, the feds may be able to hold you accountable for the actions of other users on that same community. So make sure when you do your freedom fighting, or whatever you choose to do, that you take this into considering. Always weigh out the worst case scenario, should you get busted, because the LE will try and set you up.

One last example of how LE will try and set you up, but not relating to online communities is when they put together a fake sweepstakes in Los Angeles.

<http://www.nbclosangeles.com/news/local/La-Mirada-Inspired-by-the-Simpsons-to-Catch-Criminals-78093912.html>

Quote

Sheriff's deputies in La Mirada attempted a rope-a-dope on some alleged criminals by offering them a fake sweepstakes prize. Out of the 960 letters sent to these "people of interest" only eight showed up at the La Mirada Holiday Inn to collect their prize, according to the Whittier Daily News.

Posing as the "Pelican Marketing Group," deputies sent letters last week to people throughout the county wanted in connection with crimes ranging from misdemeanor warrants to murder.

According to the report, the suspects were advised to bring their letter and identification to the Holiday Inn, and told that they were guaranteed a prize worth at least \$100, and would be one of 200 people with a chance to win a 2010 BMW 238i sedan.

They were all smiles when they showed up to collect their prizes, Deputy Janet Ramirez told the newspaper. "Once they tell them they're under arrest, the smile fades quickly," she said.

So the reason I made this post, was for those of you who think that LE will not go to certain lengths to try and set you up for charges. They will do it if they want you bad enough, and if you fall for it, they might get you on some tough charges. Curtis Green is facing up to 40 years for the sting operation by the DEA on him and the users who purchased fake IDs on the Russian forum could face up to 20 years each since they can be charged under the RICO act. Always keep these things in mind when conducting activities online and always take the worst case scenario into account.

It only takes one mistake to get caught and the government has unlimited resources and super computers to try and catch you slipping. You may only have a few laptops, desktops, servers, but nothing compared to the what they have. Be careful everyone.

FRAUDULENT PRIVATE MESSAGES

Be careful with private messages (PM) online, because one thing that comes with anonymity, is plenty of scammers.

Silk Road users have been reporting suspicious and outright fraudulent messages from users posing as Moderators asking them to download files to their computers. Here is an actual message received by another member.

Quote

This message is to inform you that the version of Tor Bundle you are using may be vulnerable to a remote execution attack through a flaw in Javascript's onreadystatechange event. This vulnerability may disclose a users actual identity and other sensitive information transmitted over the tor network.

As of January 2nd 2014 the following vulnerability was found

Title: Execution of unmapped memory through onreadystatechange event

Impact: Critical

An attack that exploits a Firefox vulnerability in JavaScript has been observed in the wild. Specifically, Windows users using the Tor Browser Bundle (which includes Firefox plus privacy patches) appear to have been targeted.

Please note: If you are using Linux or Tails (bootable) this vulnerability does not apply to you, please disregard this message.

We are advising all of our community members to upgrade to the patched version Tor Bundle (3.5)

<http://www34.zippyshare.com/v/xxxxxxx/file.html> (Latest Tor Bundle 3.5)

Mirror: http://xxxxxxxxxxxxx.onion/files/torbrowser-install-3.5_en-US.zip

Note: You do not need to remove your current Tor Bundle before installing. This will overwrite the previous installation and upgrade you to the latest 3.5 version.

If you are unsure of which version you have it is best to upgrade anyways, it will preserve your bookmarks and preferences during the upgrade.

Also...Don't Forget to Click the "Forbid Scripts Globally" after clicking on the S

The rest....Do Not mess with....this is a relatively simple thing to do....you must do this all before accessing any DarkWeb Site. Point ...Blank & Period....

This is your Safety and Security that you're Dealing with here....TAKE THIS SERIOUSLY!!

I don't mean to sound harsh or an asshole...i believe we're all Family here....and from here on out if you cannot do as told to ensure that your security and safety is not compromised.....well then you don't need to be here....Period....

Any questions? Please feel free to message any mod and we will do our best to reply Asap

Happy New Year & Stay safe in 2014!

-SR Staff

They then provide a link for you to download an "updated" version of TOR, which has been removed for security purposes. But this message is not coming from any Silk Road staff, it is coming from a random account and the files are likely to be viruses or possibly even from law enforcement.

If you get any suspicious messages from anybody claiming to be a Silk Road moderator asking you to

download software to your computer, report it to a moderator immediately so that they can ban the accounts. Do not under any circumstances download any software to your computer unless it comes from an official website such as;

<https://torproject.org>
<https://tails.boum.org/>

Again, stay safe everyone!

LEARNING FROM OTHERS' MISTAKES. HOW THEY BUSTED SABU

This next post I want to focus on more mistakes that other hacktivists and freedom fighters have made which ultimately led to their arrests. This is more proof that you only need to screw up once.

You have probably heard me talk about somebody named **Sabu** multiple times and maybe you are new to the online communities and you have no idea who I am talking about. Sabu was the leader of a self proclaimed hacktivist group called LulzSec. They were responsible for taking advantage of security exploits in online servers and posting the information online on a website called PasteBin. They had done this many times.

<https://www.informationweek.com/attacks/lulzsec-leader-sabu-unmasked-aids-fbi-hacker-sweep/d/d-id/1103214?>

Quote

The men have been charged with hacking Fox Broadcasting Company, Sony Pictures Entertainment, and the Public Broadcasting Service (aka PBS).

During the time all this was happening, the members of this group maintained an online Internet Relay Chat (IRC) channel in which they regularly discussed and took credit for their attacks and exploits. The agreed upon ring leader for these attacks, and this group went by the online handle Sabu. Sabu had also been linked to selling stolen credit cards on Facebook through his online handle, not his real one, which carries a charge of aggravated identity theft.

The group had leaked identities of law enforcement, Sony users, and all wreaked all types of havoc online including DDos attacks on the CIA. The FBI wanted Sabu, they wanted the ring leader, who would eventually be facing charges that could lead to 112 years in prison. But as I mentioned in previous threads, it only takes one mistake to get caught. That is all they need.

<http://www.foxnews.com/tech/2012/03/06/exclusive-unmasking-worlds-most-wanted-hacker/>

Quote

Sabu had always been cautious, hiding his Internet protocol address through proxy servers. But then just once he slipped. He logged into an Internet relay chatroom from his own IP address without masking it. **All it took was once.** The feds had a fix on him.

However, this was not his first actual slip up, but it was his first slip up where the feds actually discovered his mistake. His identity was actually discovered, or "doxed" previously by another online hacking group called **Backtrace** who posted his identity and general location online weeks prior to this in an attempt to dox members of LulzSec.

<http://arstechnica.com/tech-policy/2012/03/doxed-how-sabu-was-outed-by-former-anons-long-before-his-arrest/>

Quote

Sabu occasionally mentioned ownership of a domain called prvt.org in his chats, including those in Backtrace's "consequences" document. Every domain registration is associated with corresponding information in the WHOIS database. This information is supposed to include the name and address of the domain's owner.

Often this information is incorrect (most domain registrars do nothing to validate it) or anonymized (many firms offer "proxy" domain registration, so the WHOIS database contains the details of the proxy registrar, rather than the person using the domain). Monsegur appeared to use one of these anonymizing services, Go Daddy subsidiary Domains By Proxy, for registering the prvt.org domain.

The registration for the domain was due to expire on June 25, 2011, requiring Monsegur to renew it. But for some reason—error on Monsegur's part perhaps, or screw-up by the registrar—the renewal was processed not by Domains By Proxy but by its parent, Go Daddy. Unlike Domains By Proxy, Go Daddy uses real information when it updates the WHOIS database, so on 24th June (the day before it was due to expire), **Monsegur's name, address, and telephone number were all publicly attached to his domain name.**

Monsegur quickly remedied the mistake, changing the WHOIS registration to use various other identities—first to that of Adrian Lamo (who reported Bradley Manning to authorities) and then to "Rafael Lima" and subsequently to "Christian Biermann". This attempt to mislead those relying on the WHOIS information successfully misled some would-be doxers. But not all: by August there were extensive dossiers on Sabu's true identity.

Two mistakes that we know of, is all that it took to bring down at one time, the World's Most Wanted Hacker. If you are familiar with the story of LulzSec, there was a time they were receiving mainstream news coverage and Sabu had gained a reputation of being this mystical untouchable hacker. Unfortunately for him, he made two small yet very costly mistakes which ended up putting him away. But we are not done yet on this story about Sabu.

Sabu had a weakness, that the feds used as leverage against him when he got busted.

Quote

An unemployed computer programmer, welfare recipient and **legal guardian of two young children.**

"It was because of his kids," one of the two agents recalled. "He'd do anything for his kids. He didn't

want to go away to prison and leave them. That's how we got him."

Monsegur was quietly arrested on aggravated identity theft charges and released on bail. On Aug. 15 he pleaded guilty to a dozen counts of hacking-related charges and **agreed to cooperate with the FBI.**

So when you are doing your freedom fighting online, you need to ask yourself. What do I have to lose? Do I have a wife? Children? What would happen if I were to lose everything and be thrown away for 10 to 20 years, could I handle that? If you decide that you are willing to risk all that, then you again need to learn from the mistakes of those who have fallen before you. Ask yourself, if put in a hard place, where you had to choose between life in prison, and cooperation, in order to see your own family, you may think you will not talk now, but you may start talking when the feds are threatening to take them away from you forever.

Once the FBI had the leader of the group LulzSec working for them, they wasted little time getting the former hacker to turn on his friends and aid in their arrests.

Continued next post.

LEARNING FROM OTHERS' MISTAKES. SABU BECAME FBI INFORMANT AND BETRAYED JEREMY HAMMOND

We are continuing the subject of how others were taken down after Sabu was compromised and started cooperating with the FBI. According to this article.

<http://arstechnica.com/tech-policy/2012/03/stakeout-how-the-fbi-tracked-and-busted-a-chicago-anon/>

Quote

The day after Christmas, sup_g had another online chat about the Stratfor hack and about some 30,000 credit card numbers that had been taken from the company. His interlocutor, **CW-1**, engaged in a bit of gallows humor about what might happen should they all get caught.

But the raid had, in fact, already happened. **CW-1 was "Sabu,"** a top Anon/LulzSec hacker who was in real life an unemployed 28-year old living in New York City public housing. His sixth-floor apartment had been visited by the FBI in June 2011, and Sabu had been arrested and "turned." For months, he had been an FBI informant, watched 24 hours a day by an agent and using a government issued laptop that logged everything he did.

So we see here Sabu is chatting with a user **sup_g** to try and engage him about the hacks that took place.

Quote

Sabu suddenly addresses sup_g by a new name, "anarchaos." It would turn out that sup_g went by many names, including "anarchaos," "**burn**," "yohoho," "POW," "tylerknowsthis," and "crediblethreat."

CW-1: if I get raided **anarchaos** your job is to cause havok in my honor

CW-1: <3

CW-1: sup_g:

@sup_g: it shall be so

Normally, the attempt to link his various names would have raised the hacker's guard; as he confided to Sabu, someone else had once tried to link the names "yohoho" and "**burn**," but the hacker "never answered... I think he picked up some language similarities I've worked with [REDACTED] on other ops in the past." But this was Sabu, a sort of hacker demigod in the world of Anonymous. If you couldn't trust him, who could you trust? Sabu had even provided a server to store the stolen Statfor data, so he couldn't be a fed (in reality, **he had done so at the FBI's direction**).

And more details on how they looked through copious amounts of logs to correlate this user **sup_g** to his real identity.

Quote

To identify sup_g, the Bureau first turned to the voluminous chat logs stored on Sabu's computer. They went through every comment that could be plausibly linked to sup_g or one of his aliases. The goal was to see if the hacker had slipped up at any point and revealed some personal information.

He had. On August 29, 2011 at 8:37 AM, "**burn**" said in an IRC channel that "some comrades of mine were arrested in St. Louis a few weeks ago... for midwestrising tar sands work." If accurate, this might place "**burn**" in the Midwest. FBI Chicago agents were able to confirm that an event called Midwest Rising was attended by Chicago resident Jeremy Hammond's twin brother. (Hammond had a history with anarchism and violent protest.)

"Anarchaos" once let slip that he had been arrested in 2004 for protesting at the Republican National Convention in New York City. Much later, "yohoho" noted that he hadn't been to New York "since the RNC," nicely tying both online handles to the same person. The FBI went to New York City police and obtained a list of every individual detained at the 2004 convention; they learned that Jeremy Hammond had in fact been detained, though he had not been arrested. The pieces were starting to fit.

"**Sup_g**" and "**burn**" both indicated later that they had spent time in prison, with "**burn**" indicating that he had been at a federal penitentiary. A search of Hammond's criminal records revealed that he had been arrested in March 2005 by the Chicago FBI and had pled guilty to hacking into a "politically conservative website and stealing its computer database, including credit card information," according to an FBI affidavit. Hammond was sentenced to two years in prison for the action.

In yet another chat, "**Anarchaos**" told Sabu that he had once spent a few weeks in a county jail for possession of marijuana. He also asked Sabu not to tell anybody, "**cause it could compromise my identity**," and he noted that he was on probation. Both matched Hammond, who was placed on probation in November 2010 after a violent protest against the Olympics coming to Chicago. When the FBI ran a criminal history check on Hammond, it also revealed two arrests for marijuana possession.

The FBI was so thorough that it even followed up on a "POW" comment saying "dumpster diving is all

good i'm a freegan goddess." ("Freegans" scavenge unspoiled, wasted food from the trash of grocery stores and restaurants.) The FBI went to Chicago authorities, who had put Hammond under surveillance when they were investigating him back in 2005. As part of that earlier surveillance, "agents have seen Hammond going into dumpsters to get food."

Now that they had a suspect, it was time to put him under surveillance.

This is why you all need to be extra paranoid with every single thing you say about yourselves on this forum. I have seen people talking about what country they live in, some even talking about which state they live in. If you think that the FBI will never put the pieces together, you may be sadly mistaken as Jeremy Hammond found out.

Quote

Watching the WiFi network revealed the Media Access Control (MAC) addresses of each device connected to the network. Most of the time there was only one, an Apple Computer—and sup_g had told Sabu that he used a Macbook.

On March 1, the agents obtained a court order allowing them to use a "pen register/trap and trace" device that could reveal only "addressing information" and not content. In other words, if it worked, agents could see what IP addresses Hammond was visiting, but they would see nothing else.

His Macbook's MAC address was soon seen connecting to IP addresses known to be part of the Tor anonymizing network.

And while this definitely sounded like their man, the Bureau went to even greater lengths to double-check their target. The main technique was to observe when Hammond left his home, then to call Sabu in New York and ask if any of Hammond's suspected aliases had just left IRC or the Jabber instant messaging system.

If this does not open your eyes to some of the mistakes many of you have been making online, then you need to reevaluate how you handle yourselves online. Read the entire article to get a better picture, but remember, I do not care if it is your best friend from elementary school, do not, under any circumstances ever admit anything online to anybody. Never under any circumstances take credit for any freedom fighting or hacktivism you have participated in online. And for christ's sake, NEVER log into a server, especially one that keeps logs with your real IP address!

WHERE YOU MIGHT CONSIDER RUNNING TO, IF YOU HAD NO OTHER CHOICE

In the case that you may have to run, here are some things to consider.

I am not an expert on evading extradition, or how to evade the federal government, NSA or other super powers, but I do have some recommendations that you might want to consider if you decide that

you have no other choice but to run. The following countries do not currently have an extradition treaty to the United States.

Quote

Afghanistan, Algeria, Andorra, Angola, Armenia, Bahrain, Bangladesh, Belarus, Bosnia and Herzegovina, Brunei, Burkina Faso, Burma, Burundi, Cambodia, Cameroon, Cape Verde, the Central African Republic, Chad, China, Comoros, Congo (Kinshasa), Congo (Brazzaville), Djibouti, Equatorial Guinea, Eritrea, Ethiopia, Gabon, Guinea, Guinea-Bissau, Indonesia, Ivory Coast, Kazakhstan, Kosovo, Kuwait, Laos, Lebanon, Libya, Macedonia, Madagascar, Maldives, Mali, Marshall Islands, Mauritania, Micronesia, Moldova, Mongolia, Montenegro, Morocco, Mozambique, Namibia, Nepal, Niger, Oman, Qatar, Russia, Rwanda, Samoa, São Tomé & Príncipe, Saudi Arabia, Senegal, Serbia, Somalia, Sudan, Syria, Togo, Tunisia, Uganda, Ukraine, United Arab Emirates, Uzbekistan, Vanuatu, Vatican, Vietnam and Yemen.

This does not mean that these countries will not extradite you, but if you are going to pick a country to flee to, it would be favorable to your chance to choose from this list. One notable country on this list, which is famous for extraditing one of the owners of the **Pirate Bay**, Gottfrid Svartholm to Sweden, is Cambodia. Although no treaty exists between the two countries, he was extradited by the government.

We all know that Edward Snowden fled to Russia from Hong Kong after leaving the US from Hawaii and has remained there since without being extradited by the government and was granted a 1 year temporary asylum. It is unclear if Snowden will be able to stay longer than his 1 year temporary asylum grants, but as of right now he is badly wanted by the US government, and Russia is refusing to hand him over.

Another person involved in the Pirate Bay named Fredrik Neij fled to Laos in Asia following being convicted of "assisting in making copyright content available" and was sentenced to one year in prison and ordered to pay damages of 30 million SEK (approximately €2,740,900 or US\$3,620,000). This is of course between Laos and Sweden, but Laos has not extradited Fredrik, so Laos may be a valid option.

I often hear people from the US claim that if "shit ever pops off" they would just flee to Canada. Do not even try it, you would not even make it through the border. Canada is like the baby brother of the United States. When the United States says jump, Canada says "how high?". Stay away from Canada if you are running from the United States. Even a pot activist named Mark Emery who was a Canadian citizen, lived in Canada, but sold marijuana seeds over the internet to people in the US was extradited to the US to serve a 5 year sentence. According to the other seed vendors in the area, those who only

sold within Canada had never been arrested, but because Emery sold to the US, he was arrested and extradited. And of course, we know that Ireland and Australia will likely be extraditing two of the moderators from Silk Road to the United States soon enough.

Although not on the list above, a woman, wanted in the US for parental kidnapping, named Chere Lyn Tomayko was granted asylum in Costa Rica.

http://www.usatoday.com/news/topstories/2008-07-25-3841863361_x.htm

Quote

Tomayko's claims that her actions were justified by domestic violence she suffered were taken into account by the Costa Rican authorities.

Assata Shakur was charged with murder, attempted murder, armed robbery, bank robbery, and kidnapping by the US and fled to Cuba. Cuba actually has an extradition treaty with the US, but the relations between the two countries have not been good since the cold war between the US and the Soviet Union and thus the requests were not honored, even for someone with such serious charges. Cuba may be an option for you, but again this is only something to consider as I am no expert in any way.

And finally according to a previous post of mine explaining how the Secret Service sold fake IDs online to people on a forum, several of the members of that forum were able to evade capture due to being in Eastern European countries, although not specified by the feds for obvious reasons, and remain at large to this day.

<http://www.tested.com/tech/456882-how-secret-service-sold-fake-ids-catch-identity-crooks/>

Quote

The government made its move in 2012, arresting dozens of fraudsters in the US and in countries where extradition is easy. But many more, including the founder of Cards.ru, remain at large. Those in Eastern European countries, especially, are largely out of the government's reach.

SECURING YOUR ACCOUNT FROM FBI MONITORING

I just had another realization that you may want to consider.

I noticed that certain some people on the forum were never shown as **Online**, even when they clearly were, and others were shown as online at times. I then realized to myself there must be a way to never show your status as **Online**.

The way you do this is to open up **Account Settings** and unselect the box that says **Show others my online status**.

So why would you want to do this anyways? For reasons we spoke about earlier, you do not want to give any law enforcement the ability to see when you log on and log off. It is bad practice, it can leave a trail, leave a pattern, and if you are a person of interest and they are able to connect the time you sign off on the forum with the time you leave your house, or go to sleep, it gives them more reason to be suspicious and more evidence to be used against you in court.

Consider disabling this option.

INVINCIBILITY MINDSET, FEDERAL GOVERNMENT BULLYING TACTICS

Some people have an invincibility mind set that nothing will ever be able to be tied to them or derived from their online communications.

Well guess what? They do not have to use your online communications to find out who you are. All that needs to happen, is for you to do something stupid and become a person of interest and they will be monitoring your activities online to the best of their abilities. Remember you only need to screw up once.

For example, maybe you become a person of interest and the FBI gains a subpoena to your Facebook account where you stupidly bragged to a friend of yours about participating in certain online activities. This happened to one of the members of **LulzSec** who transferred a data dump that he obtained through SQL injection exploits to a friend of his using his own Facebook in his own name. So do not ever talk about Silk Road or any of your online activities on any social media platform.

Even if a company does not currently keep logs, a court order may perhaps be used to force a company to start keeping logs. **Hush Mail** was forced to hand over 12 CDs worth of e-mails from three Hushmail accounts, following a court order obtained through a mutual assistance treaty between the U.S. and Canada. According to the following article.

<http://www.wired.com/threatlevel/2007/11/encrypted-e-mai/>

When it comes to being threatened by a court order from the federal government, 99.99% of all companies will comply to avoid either prosecution themselves, or shutting down their business as we saw previously with **Hide My Ass**.

But one company decided to stand up to this type of bullying that you may have heard of called **LavaBit** as seen in the following article.

<http://www.theguardian.com/world/2013/oct/03/lavabit-ladar-levison-fbi-encryption-keys-snowden>

Quote

The email service used by whistleblower **Edward Snowden** refused FBI requests to "defeat its own system," according to newly unsealed court documents.

The founder of Lavabit, **Ladar Levison**, repeatedly pushed back against demands by the authorities to hand over the encryption keys to his system, frustrating federal investigators who were trying to track

Snowden's communications, the documents show.

Levison is now subject to a government gag order and has appealed against the search warrants and subpoenas demanding access to his service. He closed Lavabit in August saying he did not want to be "complicit in crimes against the American people".

In July, the authorities obtained a search warrant demanding Lavabit hand over any encryption keys and SSL keys that protected the site. Levison was threatened with criminal contempt – which could have potentially put him in jail – if he did not comply. Such a move would have given the government access to all of Lavabit users' information.

The court ordered Levison to be fined \$5,000 a day beginning 6 August until he handed over electronic copies of the keys. Two days later Levison handed over the keys hours after he shuttered Lavabit.

You see what I am talking about? The federal government ordered this man to hand over all his encryption keys and SSL keys which compromised the privacy of 400,000 users just so they could gain more data on one man, Edward Snowden. And they used bullying tactics and attempted to bankrupt the owner of Lavabit by fining him \$5,000 per day until he handed over the keys. Unfortunately Levison had no choice but to hand over the keys or lose everything.

An interview on Reddit with Levison revealed what he claimed that other secure email providers who threatened to shut down were forced to stay up.

http://www.theregister.co.uk/2013/11/19/lavabit_analysis/

Quote

Lavabit's founder has claimed other secure webmail providers who threatened to shut themselves down in the wake of the NSA spying revelations had received court orders forcing them to stay up.

There you have it. Anyone who tries to stand up to the government, especially in the United States will be met with swift justice, court orders and outrageous fines unless they comply and on top of it, slapped with gag orders so they cannot tell anybody about what the government is doing.

HOW TO CONNECT TO TOR OVER TOP OF TOR

Here is another fun tip that may or may not interest you, but I figured I would throw it in for you anyways.

I figured this out while trying to figure out an effective way to do a TOR -> VPN connection. You can do TOR -> TOR connection with Tails by using a program called **Tortilla**, thus adding another layer for your adversaries to crack. Whether or not this is worth it, is completely up to you, but I am sharing in case it is something you want to do. This however currently only works for those using Windows because it was designed to be used by Windows users. Please note as well that this will noticeably slow down your connection since you are going through TOR twice. Here is the official homepage of Tortilla.

<https://github.com/CrowdStrike/Tortilla>

And the official download page for the prebuilt standalone exe below. There is a link to it on the home page if you do not trust me.

<http://www.crowdstrike.com/community-tools/>

The way you do this is very simple actually. You need to first download **TOR Expert Bundle** from the TOR Project download page and install it on your computer or better yet your USB drive.

<https://www.torproject.org/download/download.html.en>

Next open the **tor.exe** and just let it run until it says **Bootstrapped 100% Done**. Next you want to run the **tortilla.exe** file and make sure you run it with Administrator privileges. Also, if you are running Windows Vista or later, you will likely get an error that this program does not have a valid certificate, because it is actually signed with something called a test-signed certificate. In this case you need to allow test-signed drivers to run on your computer.

To do this, simply go to your Start Menu and type in the search box "command". When command comes up, you right click it, and click run as Administrator and it will open up a command prompt. Next type in the following command. **Bcdedit.exe -set TESTSIGNING ON** and this will allow Windows to install test-signed drivers. Restart your computer and you will see in the bottom right hand corner after you restart **Test Mode Windows**. Now you can run Tortilla. And let it connect to TOR. Remember to have **tor.exe** from TOR Expert Bundle open first.

Finally, you open up Virtual Box or whatever Virtual Machine software you are using and click **Settings** on the Tails virtual machine. Click on the **Network** tab and change the drop down menu where it says **Attached To:** to **Bridged Adapter** and in the drop down menu below it called Name: Select Tortilla Adapter. Now your Virtual Machine, in this case Tails, will always connect to the internet **through Tortilla**, which connects through TOR. And since Tails establishes its own connection to TOR, you will be running TOR over top of TOR. Again, you may or may not want to do this, but I am giving you the option should you want to.

If anyone is interested in learning more about the creator of Tortilla, he did a PowerPoint presentation at the 2013 Black Hat USA conference. Feel free to watch his talk at the YouTube link below. Please note however that YouTube is owned by Google and there are only about 57 views on the video, so the government will likely correlate users who watch that video with users from this forum. Make sure you do not watch the video on YouTube with your real IP address. At the very least use a VPN or find another site that has it hosted. Always be extra paranoid.

https://www.youtube.com/watch?v=G_jDPQU-8YQ

HOW TO VERIFY YOUR DOWNLOADED FILES ARE AUTHENTIC

I just had a realization about something that is pretty important and I wanted to share it with you, regarding security. **Verifying your downloads**

As a general rule of thumb, you should **always** download files from the home pages of their respective developers.

TOR: <https://www.torproject.org>

Tails: <https://www.tails.boum.org>

Virtual Box: <https://www.virtualbox.org/>

The reason this is so important, is that there are people who host maliciously modified versions of these programs and will host legitimate looking sites to try and get you to download their version, which can install things like backdoors into your computers, keyloggers, and all types of nasty surprises. Sometimes developers will offer mirrors for their projects, which are simply just alternative links to download from in case the main server is too slow, or down. Sometimes these mirrors can become compromised without the knowledge of the developers.

Maybe you do not have TOR or Tails on your laptop and you are traveling out of the country and the hotel that you are staying at has TOR's homepage blocked. There are times when you may need to find an alternative mirror to download certain things. Then of course there is the infamous **man-in-the-middle** attack where an attacker can inject malicious code into your network traffic and alter the file you are downloading. The TOR developers have even reported that attackers have the capability of tricking your browser into thinking you are visiting the TOR home page when in fact you are not.

So what do you do about it? You can verify that the file you downloaded is in fact legitimate. The best tool for this is **GnuPG**. The TOR developers recommend you get it from the following page (Windows Users).

<http://www.gpg4win.org/download.html>

You can install this program on your USB drive or on your actual computer, you will hear your actual computer's operation system referred to as your Host OS. So download it, run it, install it and we will start showing you how to use GnuPG.

If you remain on the GnuPG download page you will see something under the big green box that is called **OpenPGP signature**. Download that into the same folder as the GnuPG file, this is the file that the download was signed with. Basically someone's signature saying, I made this file. And you also need a PGP public key to verify the signature. So to sum it up so far, the signature is created from the PGP private key, and can be verified by the PGP public key. The signature file is used to verify the program itself. So let us grab the PGP public key for GnuPG as well.

If you look on the same download page, under the heading Installation, you will see a link where it says **verify** the integrity of the file. It will lead to you the following page.

<http://gpg4win.org/package-integrity.html>

Note where it says the following statement. **The signatures have been created with the following OpenPGP certificate Intevation File Distribution Key (Key ID: EC70B1B8)**. This is the link to the page that hosts the PGP public key file that you need to download, go there. On the page we just navigated to, go to the bottom right where it says **Intevation-Distribution-Key (public OpenPGP key for signing files)** and download that file. This is the PGP public key file, save it to the same place as your signature file for ease of use.

Okay, now that we have both the signature file and the PGP public key, let us now verify our download. First thing you need to do is navigate to the PGP public key file, called **Intevation-Distribution-Key.asc**, right click it and go to **More GpgEX Options** and down to **Import Keys**. This will import the PGP public key into your key ring, and now you can verify the file with the signature.

Right click your actual file you want to verify, in this case **gpg4win-2.2.1.exe** and go to **More GpgEX Options** and down to **Verify** and it should automatically detect the signature file where it says Input File, but if it does not, navigate to the signature file and make sure the box below it where it says **Input file is a detached signature** is checked. Look at the bottom and click Decrypt/Verify and you will likely get the following message.

Not enough information to check signature validity. Check details.

Believe it or not, this is completely fine. Click on show details, you are looking for a specific result.

Signed on 2013-10-07 08:31 by distribution-key@intevation.de (Key ID: 0xEC70B1B8). The validity of the signature cannot be verified.

If you navigate back to the page from Gpg4Win that says **Check Integrity** where you found the link to the page that contained the PGP public key you will see on that page.

Intevation File Distribution Key (Key ID: EC70B1B8)

Note the key ID from your decrypt result and the key ID from the Check Integrity page and note the email address ending in the same URL that we downloaded the PGP public key from. We have a match! I will explain the reason for this warning message later.

Now that we verified that our verification program is legit. Let us try and verify our Tails ISO file, since if we have a compromised Tails OS, then nothing we do will be anonymous. Let us get right to the Tails download page.

<https://tails.boum.org/download/index.en.html>

Scroll down to where it says Tails 0.22 signature and download that to your Tails folder where you have the ISO file that we already downloaded. Next scroll down to where it says Tails signing key, this is

our PGP public key. Exact same procedure, import the key, then click Verify and specify the signature file if it has not already been specified for you, exact same settings and you will get the same warning message. As explained by Tails

Quote

If you see the following warning:

```
Not enough information to check the signature validity.
Signed on ... by tails@boum.org (Key ID: 0xBE2CD9C1
The validity of the signature cannot be verified.
```

Then the ISO image is still correct, and valid according to the Tails signing key that you downloaded. This warning is related to the trust that you put in the Tails signing key. See, Trusting Tails signing key. To remove this warning you would have to personally sign the Tails signing key with your own key.

In other words, you need to basically promise that the PGP public key you downloaded is safe by signing the PGP public key with your own private key, but we do not really need to do that and I will not be including a tutorial on how to do that. Tails explains that if you are worried about a compromised PGP public key, just download the key from multiple sources and compare them, if they all match, it is a good chance you are using a legit PGP key. Now let us finally move on to TOR because this one will be a little less straight forward, but once you do this one, you should be able to figure out how to verify anything. Navigate to their download page and find the package that you want.

<https://www.torproject.org/download/download.html.en>

To keep things simple let us choose Tor Browser Bundle 3.5, and under the orange box you will see a link (**sig**). This is the link for the signature file, I hope by now you know what to do with it. Next we need the PGP public key right? Well it turns out that with so many developers working on TOR, there are multiple PGP public keys, and certain bundles were signed with different keys than other bundles. So we need to find the PGP public key that belongs to our Tor Browser Bundle. Check out this page.

<https://www.torproject.org/docs/signing-keys.html.en>

It has a list of all the signing keys that they use and you can certainly use these key IDs to get what we want by simply right clicking on the signature file and click verify. You will get a warning.

Not enough information to check signature validity. Show Details

And in details it will say the following warning.

Signed on 2013-12-19 08:34 with unknown certificate 0x416F061063FEE659

Keep this entire number in mind for later, it is called a fingerprint. But for now if you just compare the last 8 digits to Erinn Clark's key ID (**0x63FEE659**) provided on the above page, and since she is the person who signs the Tor Browser Bundles you will see they match. But we want to be a bit more

thorough, never settle for mediocrity.

Go to your task bar in Windows, and find the program called **Kleopatra**, it looks like a red circle with a small white square in it. Right click it and go to **Open Certificate Manager**. We are going to import the full keys using this manager. Also note, if you go to the tab that says **Other Certificates** you will find the Tails and Intevation (GnuPG) keys we used earlier stored for the future when you need to download a new version of those programs and verify them again.

We are going to be following the instructions from the **verifying signatures** page on the TOR Project website. Feel free to follow along from that page so you know what I am talking about and where I am getting my URL and numbers from.

<https://www.torproject.org/docs/verifying-signatures.html.en>

In order to import keys, we need to first add an online directory where they are stored. So let us first add the online directory where the PGP public keys are stored according to the TOR website. Click **Settings then Configure Kleopatra**. Next, click New and we are going to enter the following URL which I took right from the page above. **pool.sks-keyservers.net**, and leave everything else as default and click OK.

Finally, click the button that says **Lookup Certificates On Server** and we will be searching for Errin Clark's GPG public key by searching for her **fingerprint** provided on the TOR website page called **Verifying Signatures** above, remember, she is the developer who signs the Tor Browser Bundle. The fingerprint we are entering is **0x416F061063FEE659**, does this number look familiar? It should, it is the number we got back the first time we tried verifying but without the actual PGP public key. if you get any warnings that pop up when searching just click OK and it should bring up Errin Clark's key, select it and click **Import**. You should now have her key listed under **Imported Certificates**.

Now let us go back and verify that signature one more time and see what happens. You should get something like the following.

Not enough information to check signature validity.

**Signed on 201-12-17 12:41 by errin@torproject.org (Key ID: 0x63FEE659).
The validity of the signature cannot be verified.**

TOR also explains this warning message in their words in case you are still not happy with the warning message.

Quote

Notice that there is a warning because you haven't assigned a trust index to this person. This means that GnuPG verified that the key made that signature, but it's up to you to decide if that key really belongs to the developer. The best method is to meet the developer in person and exchange key fingerprints.

I do not know about you, but I am happy with the result here, and I am certainly not going to track down Erinn Clark to get her key fingerprint, and it looks like our TOR Browser Bundle is legitimate as well! Now you know what to do when the PGP public key file is not directly hosted on the site itself, you have no more excuses to not verify your downloads.

VERIFYING SIGNED MESSAGES WITH SIGNATURES AND SIGNING YOUR OWN MESSAGES

Since we just finished a section on verifying downloads with signatures and public keys, I figured we should do a quick post on verifying messages by using the same two things, signatures and public keys.

Now for those of you who are members of the Silk Road Forums, you will notice that some people, mainly Moderators like to sign their messages with signatures. Let us look at an example of a signed message from Dread Pirate Roberts. The last message he left before going on his leave of absence.

Quote

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA512

Silk Road has not been compromised even if the allegations are true. Neither had access to sensitive material. I will make an announcement later to address the concerns this has raised.

-----BEGIN PGP SIGNATURE-----

```
iQIcBAEBCgAGBQJStEgqAAoJEMyyOOR8/t+867AP/RpjCq1B3WSYgnsbcZU+UZOy
K0AGMM7tmu1DV1pr2S379YjVxQeUWeTbwDYhaYcWkDBDshnlpSd97fwAL1YVrBQx
jWE08tyo1sd1v5F/HajCx0DC2L5NeqD4UTDd6DI2AOeBI4pZ+Ah/Q4VoB9cOBQGw
lSbjBY2U4redqBeRd1mFR8N+f3XmxYXzmB4Mf8ddvQkl62HmkwRwA27uUExt73uj
f3/EYVc/XjPgKG345S8yUwcGxLQcfoRM7UosbSGeEaDvvWjfZ6qQw4p7CbqlimHu
IOT6dhFcPmoVdiZGDvjtM3jXfF2sTi5mclGp/4axsrvOWZlCbrobE9EuJnGvscU4
ekU90vtcviES9XEJAr9XGOGgzY/OBf1xpi0iRY7rBDHUqA/FjfSULxqanZYhh0Wn
webHldrjylBRKM0PsnQdPn1CVGj8ThwB6SLfd0WEN1FEQt0hXP3uK1zDori/flcJ
Pnvf3jxYNcw9Q+2OW6QpZ/7t+S2E0yifbNCobAMI18mrynuw3pk/xumg6t2WF/j
YHRpbTfFCCsbiPwR8P9CcUNQ5lqcc2ewq4GOPx053aL/Vo/nfPdu/9hrRpfF3U5E
J7rFvASTaejxH7/vNxZrRTiwrcc6njsFJHXWVAJjd+fHLI1efptbc8Kzwms9YlO
OnzLjAJPfZOv6y7gP8tG
=IDZd
-----END PGP SIGNATURE-----
```

So why should you care about this? What is the significance of signing a message? The reason is, in case somebody were to compromise DPR's account, due to having a weak password or possibly an exploit in the forum's coding, then the person would not be able to sign the messages without access to DPR's private key. So let us look at how we can verify this message left by DPR. First of all you need to visit Dread Pirate Roberts' profile page and grab his PGP public key. I am not going to post the key here for space reasons, but just visit his page at the following URL and import that key into your

keyring.

<http://silkroad5v7dywlc.onion/index.php?action=profile;u=1>

Next, highlight the entire PGP signed message from top to bottom and copy it to your clipboard (Right click, Copy). You will see your little Clipboard icon in the top right of Tails turn red. Click on that clipboard and select **Decrypt/Verify**. You should get the following results. One in the window on top and the other on the bottom.

Quote

Silk Road has not been compromised even if the allegations are true. Neither had access to sensitive material. I will make an announcement later to address the concerns this has raised.

gpg: Signature made Fri 20 Dec 2013 01:37:46 PM UTC using RSA key ID 7CFEDFBC

gpg: Good signature from "Dread Pirate Roberts <silkroad6ownowfk.onion>"

gpg: WARNING: This key is not certified with a trusted signature!

gpg: There is no indication that the signature belongs to the owner.

Primary key fingerprint: 5A48 F5D0 50E9 9052 62B4 799D CCB2 38E4 7CFE DFBC

Again we get the same warning we did when verifying our downloads, saying we have not verified that the PGP public key is authentic. We can see the signature name was made by Dread Pirate Roberts and the comment section has the Silk Road URL, so far so good. Now remember when we verified TOR? We wanted to check out the fingerprints to see if they matched. We do this by going to our key ring (Manage Keys), and selecting DPR's key, right clicking it and going to properties. Now move to the tab **Details** and look under where it says Fingerprint: and compare the numbers in there to the numbers we got when we verified the signature. They should be the following.

**5A48 F5D0 50E9 9052 62B4
799D CCB2 38E4 7CFE DFBC**

We have ourselves a match! So unless DPR's private key was compromised, we know that he himself was the one who wrote that message. So now you see why some people decide to sign their messages. It is a way of verifying that their account has not been compromised by verifying that the person in control of the account is the same person that is in control of the PGP private key.

Do you want to learn how to sign a message? It is very easy. Open up gedit Text Editor and type in a message. Next, select the message and copy it to your clipboard (Right Click - Copy) and then click on your clipboard icon up top and choose **Sign/Encrypt Clipboard with Public Keys**. Do not choose a key from your list of PGP public keys unless you want to encrypt the message. If you want to encrypt the message to send to somebody's inbox or so that only one person can view it, then select their name and it will encrypt it with their PGP public key. In our case, we just want to sign the message without encrypting it, but you can certainly do both at the same time if you wanted to.

If you look down near the bottom you will see where it says **Sign message as:** click on this and select your personal key. It will ask you for your passphrase because remember you are signing this with your

private key. Once you enter it correctly, the PGP signed message will be copied to your clipboard and you can paste it anywhere (Right Click - Paste) that you want to. Here it what mine looked like.

-----BEGIN PGP SIGNED MESSAGE-----

Hash: SHA512

This is my PGP signed message for demonstration purposes.

-----BEGIN PGP SIGNATURE-----

```
iQIcBAEBCgAGBQJS0GiWAAoJEPuh6tSg81nyqXAP/2mEqvk9RP0FEHZi3edH9faV
OmDoOostmzm90nGMGOOu4cuG0M6jgl7R3hfUZBE6zCh59MG8a9EDuUzptIT3U5nfd
zS0GWtzUQKGXPxfJ1OvWlsA6Sm7TsEsviBBz5DJxyVLcJGNU6OLUVm7onxBLwfTq
D1jAATIB43WJbDrq3XY9MF9GCoOLLcLeKNVa4m0JF582lvQJ05mSZXeXCueImvol
FaflpLW5MKyJJ92a8uheB0pLHUQTLr6jZn6TcfKY9dK8puOam5k2TGut/Sm47uqc
aMA1trXw4xntww/8X4QyL5SbSN7QVOFsy/g0b3Grp5OrConsfnsUoeRH5ArnxY0W
ijPI92aTbZazvXspW2REk3yq+fWjuGrYHw8m7/YVBig+OSMuBUXhSE5Pjq95fyM
bA1P7rF2fi7eRslz0qyETV3Bs1RltwvBUVIwj3SZNeVVoG5cHgpiPgGFq4S9Qke
unIFeHy3YpBk90kLA1n8n61VnkKAUy0Dt9AoTJloeOqPtcgeKHVsFzxdPCBcSwqd
XYnlx4lNeaw4OvHYgZsCMvFIUitSBGnFWLN9foQ8UybAUPGI9Z4sK2WmtYWK4fLI
cXnYY9zt56Ji4DiVsQrEUamNTQEDGxpvBL/kQKRMKN6HviEXW+qr57LAo6t6sTQw
KTV4uJkH1JxuOOhN9tle
=Nkox
```

-----END PGP SIGNATURE-----

And if you want to verify it, check out my PGP public key in my profile and verify my PGP signature against my key! It is really that simple. But you might be asking, cannot somebody just change the message and copy the signature? No, changing the message will change the signature because the signature depends on both the message and the PGP private key. So if you change one single character of my signed message you will get the following error.

gpg: Signature made Fri 10 Jan 2014 09:39:34 PM UTC using RSA key ID A0F359F2

gpg: BAD signature from "Jolly Roger (They would live and die under it)"

So when should you sign a message? And when should you not sign a message? Great question. The majority of users should probably not sign messages unless they have to because it gives you plausible deniability. It is easier to deny posting certain things or certain communications you may have had with vendors or other people including law enforcement if you do not sign your messages, because you can always claim somebody else gained access to your account. It is harder to do this if you signed the message with your PGP private key. If you are dealing with somebody who wants to verify your identity and make sure that your current signature matches the public key they had on file for you from 6 months ago, then maybe they might get you to send a signed message. But again, all they really need to do is send you an encrypted message with your PGP public key they had on file, and if you cannot decrypt it, you are not who you say you are.

In real world application, developers can use PGP signed messages in News Announcements or

perhaps new releases of their programs providing a download URL so that users can be sure the developer is the one posting the URL and not some malicious attacker who compromised the forum account of the developer and so forth. So for the average Silk Road forum user there really is not a lot of times when you should be signing messages unless you are a moderator or making a public announcement and so forth, but it is an option you now have in your arsenal, and now you can start verifying the signatures of the Administrators and Moderators in case you believe their accounts may have been compromised.

AN EXAMPLE OF REALLY BAD OPSEC - SMARTEN UP!

Guys, I am not going to post exactly who started this thread quoted below, but it belongs to somebody who is a senior member with 375 posts. And he posted some very personal details and probably did not realize how very revealing these details are.

Quote

Track Me If You Can...

Awesome bit I just watched on Netflix. This is not terribly new, done in 2010, but he is quite thorough in his demonstrating how to disappear in modern US culture.

I do have to add that some of the tech he introduced from the other side is quite alarming.

The alphabet cops have waaay too much discretionary income. Time to start defunding them.

So why is this revealing? Why is this bad you might be asking. Well, Netflix collects metadata on its users just like every other big data corporation. If you are a Netflix user, you likely have a profile which keeps track of every movie you have ever watched and what you rated it and so forth.

<http://www.usatoday.com/story/theoval/2013/12/17/obama-bidenapple-amazon-twitter-netflix-yahoo-facebook-microsoft-google/4049305/>

Quote

Electronic surveillance and the new health care law are on the agenda as Obama and Vice President Biden meet with a group that includes executives from Apple, Amazon, Twitter, **Netflix**, Yahoo, Facebook, Microsoft, and Google.

This user stated, that he just watched a specific movie, that he named. And also stated that this movie has been up since 2010. So how many people do you think watched this exact movie in the time frame that this guy stated he watched it? Probably not too many. Under 100 for sure since the movie has been up for almost 3 years. Well, now the federal government has a list of 100 or less suspects, one of which is this particular user on Silk Road.

But maybe he was using a VPN to connect to Netflix? Great.... does he use that VPN for anything else? Logging into his email, surfing the web, etc... Even if he used a VPN, maybe they keep logs? Maybe they are US based and are easily subject to subpoenas, maybe they will spill everything just like

HideMyAss did. We just do not know, but this is exactly the type of information you all should NOT be revealing about yourselves. This is extremely bad OpSec people. Smarten up!

But then I looked even further through this user's profile and looked at his posts. I know which country he lives in, I know which drugs he has imported into his country and I know which countries he has imported those drugs from. This guy has spoken about cooking drugs, he talks about being in a cold part of his country, which not all parts of that particular country ever even get cold which helps law enforcement narrow down that list of suspects they got from Netflix.

If you think that law enforcement is not interested in buyers you are sadly mistaken. Sometimes if they establish that a buyer has been purchasing from a vendor that they are after, then busting the buyer can help them get to the vendor. They may take over the user's online identity and start ordering things from vendors since he already has established a trust with these particular vendors. If the vendor slips up because of the trust built up with the buyer, the vendor is in trouble.

I want you all to learn a lesson from this! If you are going to talk about which vendors you buy off of, which country you live in and which countries you have imported drugs from, then you better make DAMN SURE you do not start giving away details like which movie you watched last night on Netflix. That is almost as bad as logging onto a server with your real IP address. Keep your mouths shut about your personal lives!
[/quote]

TOR CHAT

By now if you have been following this thread, you should know that any type of messaging system is likely compromised or storing your data for an unknown period of time, and if you ever become a person of interest can be looked back upon for 5+ years.

This means things like Gmail, Hotmail, Yahoo Mail, Skype Messaging, Facebook Instant/Private Message, Text Messages, and other forms of communication are all likely being monitored to some degree, at the very least logging the meta data. But you should always treat everything as if those who are monitoring it can read the content of the email as well.

We have talked about communicating with PGP, we have talked about using TOR and hidden services, and we have talked about good practices of OpSec. But some of us want to be able to instant message somebody else. The good news is, you can do this with something called TorChat.

TorChat is a decentralized anonymous instant messenger that uses Tor hidden services as its underlying Network, in other words it communicates over the Tor network through the .onion URL protocol. This provides end to end encryption that we talked about in previous posts. It provides cryptographically secure text messaging and file transfers for business dealings, and confidential communication between two people. The best news, is that you can use TorChat on your Windows, Linux and your smart phones. A French developer released a version for MAC users, but it still in beta and should be used at your own risk. You can get TorChat for the iPhone in the Apple store, you can get

TorChat in the Android Market as well, so you can even use it as a means of text messaging somebody else who also has TorChat.

In TorChat, every user has a unique alphanumeric ID consisting of 16 characters. This ID will be randomly created by Tor when the client is started the first time, it is basically the .onion address of a hidden service. TorChat clients communicate with each other by using Tor to contact the other's hidden service. For example, the first time you open TorChat your computer might generate d0dj309j94jfgf.onion and from here on out, d0dj309j94jfgf will be your TorChat ID that you give out to people that you want to be able to message you. Here is the home page of TorChat.

<https://github.com/prof7bit/TorChat>

<http://www.sourcemacs.com/?page=torchat> - MAC users

Unfortunately at this time, TorChat does not run properly in Tails, so you will either need to run it on your Windows, Linux or MAC system. It is pretty straight forward, download it, unpack it and run it and everything else should happen automatically for you. Once the avatar beside your TorChat ID turns green, you are online and same with your contacts. You can add contacts by right clicking and choosing Add Contact and just enter their TorChat ID.

At this time there is some people debate as to whether or not TorChat is completely safe, and I would say that TorChat is about as safe as Tor is, just make sure you practice the same good practices you are used to. Do not give out personal information, if you are sending sensitive information use PGP encryption and so forth.

Here is another article on how TorChat works going into a little bit more detail. You can access it over the onion network.

http://kpvz7ki2v5agwt35.onion/wiki/index.php/Hacking_TorChat

UPDATE

Another user had some additional input that I overlooked when writing this post that you should be aware of.

[Quote from: ldopa on January 13, 2014, 08:43:25 am](#)

Torchat's security is unknown. It has not undergone a proper security audit, professional or otherwise, that I know of. It creates a hidden service on your computer leaving you vulnerable to deanonymization attacks that apply to all hidden services. It also seems to be a very basic protocol that looks like netcat over Tor. There is no way to decline a file transfer. It automatically starts the transfer, writing the file to /tmp which is a RAM-mounted tmpfs on Linux. Then you are supposed to save the file somewhere. Theoretically an attacker could transfer /dev/urandom while you are away from your computer until it fills up your RAM and crashes your computer. This would be great for inducing intersection attacks. Not sure though. If the kernel is managing the system correctly, it may just stop the transfer when you run out of RAM.

Another thing is that once someone learns your Torchat ID there is no way to prevent them from

knowing you are online, even if you remove them from your buddy list. The reason is because your Torchat instance is a hidden service that publishes a normal hidden service descriptor which anyone can download. There's no way to stop that. If you want to cut off contact with someone, you have to get a new Torchat ID. So you should be very conservative about handing out your Torchat ID and only give it to extremely trusted associates.

OBTAINING, SENDING AND RECEIVING BITCOINS ANONYMOUSLY

This post was inspired by a user who posted the following on the Silk Road forums.

Quote from: dusttodust on January 12, 2014, 07:39:43 pm

BEST WAY TO OBTAIN BTC'S? AND HOW DO YOU PROTECT IDENTIY DONIG SO? i just would like to know so i can get over this bump i been learning all this stuff to do shit on these sites for a month now and this is my last obsticle i think?!!

We have talked about a large amount of ways to maintain your security, but we have not really talked about how to actually exchange currency. First thing I want to say as a disclaimer, is that I am not advocating that you do anything illegal. This is for educational purposes only and my recommendations are made assuming you are exchanging currencies anonymously as a means to protect your own privacy.

So you have found something online that you want to buy, and they are asking for Bitcoins as payment. How do you get the Bitcoins, and how do you get the Bitcoins to them? We are going to explore these options to a degree and hopefully by then you can make an educated decision on which method is best for your situation.

The options of buying Bitcoins are as follows.

1. Sign up at an exchange online. Some popular exchanges are **MT Gox, BTC-E, BitStamp and Coinbase**. The downside of purchasing Bitcoins at these exchanges, are that you need to verify your identity with them by means of submitting documents such as a driver's license or passport and a utility bill. If you are able to get past this first obstacle, then you need to find a way to get money into the account. Exchanges generally only accept wire transfers as a way to fund your account, but some of them offer a way of transferring money directly from your bank account. You can obviously see that by doing this you are exposing your true identity to the exchanges in one way or another, if not at the very least your location.

2. LocalBitcoins.com

LocalBitcoins offers a way for you to find a person in your local area, or if you want to go to another state or province to meet up with someone further away from you, you can choose where to look for people in that area selling Bitcoins either online (bank transfer or cash deposit) or meet them for cash in person. Traders have reputation lists, similar to a feedback score on eBay and you can find a trader who has a good reputation to buy off of. You send in a trade request and once the seller has received the money, he can release the Bitcoins from LocalBitcoins and they are sent to your wallet. Some people have expressed concern that law enforcement may act as buyers and sellers on LocalBitcoins, but it does not matter if this is the case in my opinion as long as you are not looking to buy large

amounts. You can also, if you want, communicate with the buyer over email, arrive from public transportation, wear a hat, and all sorts of secret agent type tricks to try and conceal your identity. Wear a wig if you are super paranoid.

3. Use a Bitcoin ATM

Currently there is only one ATM in the world that I am aware of, and it is located in Canada. If you do not live in Canada then this does not help you. Luckily according to the an article, the company who is rolling out these ATMs called Robocoin is launching ATMs in other countries as well coming soon.

<http://techcrunch.com/2014/01/02/robocoin-the-bitcoin-atm-is-heading-to-hong-kong-and-taiwan/>

Quote

The first shipping bitcoin ATM, Robocoin, is landing in Hong Kong and Taiwan as the company expands its reach this January. They are planning further releases in Europe, Canada, and the US but, given Asia's clout in the BTC markets, this is definitely an interesting development.

There will likely be some way to try and cut down on money laundering by getting you to verify your identification, but from what I understand, they currently only do this if you are selling Bitcoins for cash using the ATM, and not buying them for cash. The way that it works, is you choose the amount of BTC you want to buy, and you feed your cash into the ATM machine. You can at that point either print out a generated paper wallet, or choose a wallet of your own to send the Bitcoins to. This method may be another good way because it takes dealing with another human out of the transaction. Something you may need to be aware of is surveillance cameras, so maybe wear a hood, hat, wig, sunglasses, and so forth to disguise yourself if you are worried about your identity.

4. Craigslist

Believe it or not, there are a decent amount of people on Craigslist that you can meet up with in person and buy Bitcoins off of with cash. Your local area may not have a large number of listings, but you can always search in other nearby metropolitan areas and make a day trip out of it if you want. The same considerations about protecting your identity apply here as above.

5. Mine your own Bitcoins

I am not going to get into how to mine Bitcoins, or whether or not you should, but if you want to get Bitcoins without dealing with other people, this is one of the ways you can do it. Run your miners over Tor, stay anonymous and you will have yourself some untainted Bitcoins.

Okay, so now you have yourself some Bitcoins, how can you get them to somebody else that you want to buy something off of or trade with? As you probably know by now, every single transaction is tracked on BlockChain.info. My wallet address that I have set up for donations for the hours I have spent working on this thread is 1Pkj928QWC5BuQAsHoNQzRV5wfnveJSRCp. You can check out the transactions related to it by going to the following address.

<http://blockchain.info/address/1Pkj928QWC5BuQAsHoNQzRV5wfnveJSRCp>

So you have Bitcoins sitting in your wallet, and if you send them to somebody else, it will show up on

Blockchain exactly where you sent them. A couple of things to keep in mind.

1. You purchased your Bitcoins from somebody or something. They may have kept a record of the wallet those coins were sent to.
2. If you dealt with a law enforcement or somebody trying to track you, then they can track where the coins are sent after you forward them to somebody else.

Right now the best method of trying to lose this trail is using something called a mixer or a tumbler. You can think of this like throwing your Bitcoins into a giant pile of coins with other users and then withdrawing them at a later time from the mixer. If you threw in 1 Bitcoin and pulled out 1 Bitcoin, think of all the other people who did the exact same thing. Possibly thousands of others withdrawing 1 Bitcoin from the exact same pile of coins. It has now become much harder for you to be linked to those coins. Then on top of that, maybe you do not withdraw 1 Bitcoin, maybe you only withdraw 0.5 Bitcoin right now and leave the other 0.5 Bitcoin in the pile. It becomes even harder to link those Bitcoins to you.

One website that does this is called BitcoinFog and can be found on a cleartnet URL and a hidden services URL.

<http://www.bitcoinfog.com/>
<http://fogcore5n3ov3tui.onion/>

BitcoinFog has been around for a while now and most people seem happy with the service they provide, so I would come to think that they are a trustworthy service. The way they work is as I mentioned above, and on top of that the service takes 1%-3% (randomized for obscurity) fee on each deposit. So you may put in 1.0 Bitcoins and take out 0.97 Bitcoin after fees and it mixes things up. You can also decide when you might want to withdraw it, whether it is in a month, week, days, and so forth. This is a good service to use and definitely mixes things up for you. The only thing you need to keep in mind, is that there is a trail of you sending your coins into BitcoinFog, which some people may or may not find suspicious. But what you do with your coins after BitcoinFog is going to be extremely difficult to track, if not impossible due to the vast number of transactions that are occurring in and out of BitcoinFog.

When you withdraw your coins from BitcoinFog, please make sure you send them to a new wallet, and not the same wallet that you used to deposit them into BitcoinFog. Another option you can have when withdrawing the coins from BitcoinFog, is to get BitcoinFog to withdraw the coins directly to the person you want to buy something from. This takes the step of creating a new wallet and then having to forward it on and will keep things again extremely hard to track. Just keep their transaction fees in mind to make sure your desired seller is going to receive the correct amount of Bitcoins needed for the purchase or exchange.

Two other options you can use are provided by Blockchain.info and can be accessed by creating a wallet and logging in to it. Send Shared and Shared Coin. Send Shared is another way of mixing up coins, the way that it works is, you send your money into the giant pot and it gets matched up with somebody else who is sending the same amount. An example of this is let us say we have 4 people. A, B

and X, Y. Person A is sending 1 Bitcoin to person B and person X is sending 1 Bitcoin to person Y. Send Shared will match these amounts together, and it will mix them so that person A sends their 1 Bitcoin to person Y and person X sends their Bitcoin to person B. This way you are breaking the chain that links person A to person B because there is no record of person A ever sending anything to person B. This is a very good option to use, and one that many people prefer. Of course, there are many people using Send Shared, so the likelihood of there just being 4 people mixing up transaction is going to be more like 10,000 or more, making it pretty much impossible to track.

Shared coin uses a different method called coinjoin. Shared coin hosts a coinjoin server which acts as a meeting point for multiple people to join together in a single transaction. Having multiple people in a transaction improves privacy by making transactions more difficult to analyse. The important distinction between traditional mixing services is the server cannot confiscate or steal your coins. A sharedcoin transaction will look something like the following.

<https://blockchain.info/tx/e4abb15310348edc606e597effc81697bfce4b6de7598347f17c2befd4febf3b>

As you can see multiple inputs and outputs make the determining the actual sender and receiver more difficult. Basically it sends the coins in and out of many different wallets that are participating in Shared coin at the time and it does this to throw hundreds or thousands of transactions in all the wallets participating making it extremely difficult to track. The downside though is that coinjoin can never completely sever the link between the input and destination address, there will always be a connection between them, it is just more difficult to analyse. The benefit to Shared Coin is that while this processing is happening, you can hit cancel and get your coins back. When you send your coins into a traditional mixing service, an untrustworthy mixing service could potentially steal your coins.

Now that you have the knowledge to make an educated decision on how to mix up your coins en route to your intended destination, I feel that you can now put your mind at ease when looking to buy something with Bitcoins. It should be noted that you can reverse the process if you want to cash out your Bitcoins as well.

CLEARNET VS HIDDEN SERVICES - WHY YOU SHOULD BE CAREFUL

Some of you may have seen links to different websites on these forums. In fact my thread is full of them.

As you probably know by now, a hidden service is a website that uses a .onion address and a clearnet site uses the regular internet. You must be on TOR to access the onion network, whereas clearnet sites can be accessed from any browser. So why should you be careful when visiting clearnet sites?

When you see an article, link or video posted on the Silk Road forums, please note, that you should only be viewing those videos over TOR or possibly but as a last resort use a VPN and here is why. Let us use YouTube for example. YouTube is owned by Google, Google tracks **everything**. YouTube keeps track of which IP addresses search for what videos, and tons of meta data about it's users.

When a link to a YouTube video is posted on the SR forums, we likely have to use our regular browsers to watch it because Tor browser is not good for watching flash videos. But the problem is, if a post on SR was written on January 10, 2014 recommending a video, and this video only has 500 views, perhaps this video has been up for a few months and did not end up being very popular. And then within the few days that this article was posted, 50 people viewing the Silk Road forum watch this video. The number of views just went up in a short period of time.

It is pretty easy to correlate that it is possible, that the people who watched that YouTube video, especially since it is not a popular video came from Silk Road, and if you made the mistake of using your real IP address, you have now been added to a list of people of interest. And if you do this multiple times with different YouTube videos, then they start to see a pattern and before you know it, they are confident that you are coming to watch these videos from Silk Road because every time a video is posted on Silk Road forums, your IP address comes up to watch this video.

But if you use a VPN, this makes things a little harder in that they are not as easily going to be able to link the video to you yet. But once they see a VPN address constantly popping up on those videos being linked from the forums, they might submit a court order to monitor the activities of the users of the VPN. HideMyAss was one of the most well known examples of VPNs being ordered to hand over information on their users.

The same thing goes with all clearnet sites. You never know who is monitoring their activity, and if it is an old article, more than a couple of years, then you can almost bet that the number of people viewing that article are down. So when somebody posts a clearnet link on the forums and people visit that link using an unprotected IP address, then the LE can start to correlate patterns against you. Of course, these articles and links are not as likely to be visited without TOR from the SR forums because you need TOR to view the forums, but especially things like YouTube videos since TOR does not work well with YouTube can be problematic.

So what can you do to protect yourself? Ask yourself first, do I really need to watch that YouTube video? Is it something important that I need to see? If it is, you might consider an option that I spoke about earlier called Tortilla, but it is only available to Windows users. I talk it about it at the following article.

<http://silkroad5v7dywlc.onion/index.php?topic=14555.msg304569#msg304569>

You will run a Virtual Machine such as Debian, but do not connect to TOR using the Virtual Machine. The VM uses a bridged adapter and routes all traffic through Tortilla which routes all traffic through TOR on your Windows host OS without having to use the TOR browser on your VM. MAC users and Linux users may just want to view the YouTube video in a one time use proxy that does not keep any logs or maybe a public wifi network that has lots of users on it daily.

There is an infamous case of a murderer who called the sister of his victim from his victim's cell phone. He would call from her Time Square in New York and taunt her and talk about how she was torturing her sister and the police put a trace on the phone. Unfortunately because Time Square is such a crowded place, even with all the cameras, they were unable to pinpoint exactly which person was

making the call on that phone and they never ended up catching the guy. He ended up ditching the phone after he finally killed his victim. They knew he was a guy walking around Time Square on a cell phone but if you have ever been to Time Square, you know that there are millions of people doing the exact same thing, he just blended right in.

So you may want to use a public wifi in a crowded area that has many users all day long to watch a video and keep your IP address safe. If you cannot watch videos safely without identifying yourself, then do not watch them. It is as simple as this. Yes I know it is annoying that Tor does not work well with flash videos, but it is better than being thrown in jail where you will never be able to watch any YouTube videos.

The main reason I wrote this post was to remind you that correlating two users together on the internet is easier than you think. Once you start developing patterns and leaving your footprints behind, the LE have an unlimited storage space available to them to keep track of everything you do. Remember how Sabu got caught? He just logged onto IRC with his real IP address, **one time**. One time is all it takes for them to take you down. Always think before opening a link, what will this website identify about me?

THEY ARE WATCHING YOU - VIRUSES, MALWARE, VULNERABILITIES

Your computer will always be vulnerable to some sort of attack from those who want to harm you in some way. Whether it is harm your privacy, steal your information or throw you in jail.

It should come to no surprise to us that the US government is actually the largest purchaser of malware.

Quote

According to a new report, the United States government is now in fact the single largest buyer of malware in the world thanks to the shift to “offensive” cybersecurity and is leaving us all vulnerable in the process.

In order for the government to exploit vulnerabilities discovered in major software, they cannot disclose those vulnerabilities to the manufacturers or the public, lest the exploit be fixed.

“My job was to have 25 zero-days on a USB stick, ready to go,” one former executive at a defense contractor told Reuters. The defense contractor would purchase vulnerabilities from independent hackers and then turn them into exploits for the government to use as an offensive cyberweapon.

<http://endthelie.com/2013/05/10/report-us-government-now-buys-more-malware-than-anyone-else-in-the-world/#axzz2qljeZ32e>

After reviewing the sources in the article and other articles, some of these defense contractors expressed concern that the government was essentially funding criminal activity. They are paying independent hackers, in some cases blackhats to find zero day exploits (ones that have not been

publicly announced yet) and buy these exploits off of them for huge sums up money, upwards of \$100,000.

If you are using a laptop with a built-in microphone and camera, you are extremely vulnerable to an attack as John McAfee, the man who started McAfee Anti Virus explains.

Quote

"We don't have much [security] anymore, and certainly not in the online world," he said at Saturday's talk. "If you can give me just any small amount of information about yourself, I promise you, within three days, I can turn on the camera on your computer at home and watch whatever you're doing."

<http://abcnews.go.com/Technology/john-mcafees-product-aims-make-internet-users-virtually/story?id=20424182>

So the first thing you should do right now is go grab some opaque tape and put it over your camera. If you are on a desktop and you have a webcam plugged in, unplug it unless you are using it. There is no reason to give an attacker an open window into your home. Next is your microphone, again desktops usually do not have built in microphones, but most laptops do. A microphone can be activated to listen to you talking and you need to find a way to physically disable it. The best way of course is to physically remove it, but I am not writing a tutorial on how to do that.

The FBI developed a keystroke logging software called Magic Lantern. Magic Lantern can reportedly be installed remotely, via an e-mail attachment or by exploiting common operating system vulnerabilities, unlike previous keystroke logger programs used by the FBI. It has been variously described as a virus and a Trojan horse. It is not known how the program might store or communicate the recorded keystrokes.

Quote

The FBI intends to deploy Magic Lantern in the form of an e-mail attachment. When the attachment is opened, it installs a trojan horse on the suspect's computer. **The trojan horse is activated when the suspect uses PGP encryption, often used to increase the security of sent e-mail messages. When activated, the trojan horse will log the PGP password, which allows the FBI to decrypt user communications.**

Spokesmen for the FBI soon confirmed the existence of a program called Magic Lantern. They denied that it had been deployed, and they declined to comment further

Source: https://en.wikipedia.org/wiki/Magic_Lantern_%28software%29

Then of course we have cell phones which can be activated remotely as well.

Quote

Mobile phone (cell phone) microphones can be activated remotely, without any need for physical access. This "roving bug" feature has been used by law enforcement agencies and intelligence services to listen in on nearby conversations

https://en.wikipedia.org/wiki/Covert_listening_device#Remotely_activated_mobile_phone_microphones

According to a few of the sources in the Wikipedia article, the cell phone can be activated to listen to you even when it is off. Pulling the battery will likely do the job, but there is no guarantee. So make sure the phone is not in the same room as you if you are talking about anything sensitive. As always, be super paranoid. Turn on the shower and put the phone in the bathroom if you have to, or better yet if you are going somewhere and you do not need your cell phone, leave it at home. Since most people never leave home without their cell phones, if somebody is snooping on you, they might think you are still at home. The first group of people that went to visit Snowden in Russia were told not to bring any laptops or cell phones with them for those reasons.

So we know the government is actively trying to gain remote access to your computer, they can listen to your phones, what should you do about it ?

You need to do the best you can to make sure the computers that you use are not exposed to the elements of risk. Always disable Javascript when visiting any websites unless the website is 100% trusted. Start phasing out the use of Microsoft Windows and MAC OSX because these closed source proprietary operating systems are not open to scrutiny and auditing the way open source Linux distributions are. There are more Windows users and thus more exploits available for Windows.

Running your operating system in a Virtual Machine, even if your host OS is Linux (remember Virtual Box can run on Linux) will help cut down on the retention of any malware you might pick up when on the internet. Do not go to any potentially harmful sites on your freedom fighting computers. Do not open any emails from anyone that you do not trust 100%. Regularly format your hard drives to keep them clean of any hidden viruses.

If you are unsure if something is safe, test it on a computer only meant for testing and one that is not connected to the internet. If you can reset your boot sector on your hard drive from time to time that would be a good idea as well, because you can get master boot sector viruses that would boot up a virus before your computer even boots into the OS.

Flash your BIOS, the BIOS is the first thing that runs when you turn on your computer, if you have a virus in your BIOS, there is no antivirus that can remove it, you would need to flash your BIOS and install a new firmware. Make sure the firmware is 100% trustworthy as infected firmware is the most common way to get a BIOS virus.

In the interest of saving space I will not go into detail on how to do all of these virus removals because there are numerous tutorials online and I am certainly not an expert in this field. I am sure there are many other things I have not covered in this post and if somebody else wants to chime in, please feel free to do so as long as you can provide sources for the claims you are making. I do not want to turn this thread into a bunch of unsubstantiated claims and paranoid conspiracy theories. But if you have something valuable to add to this, I am open to your input.

MONITORING YOU WITH AN ANTENNA

First thing I want you to do is find a secure way of watching this video. Remember they log everyone who watches these videos and since I am linking you to them from Silk Road, they will be watched even closer.

http://www.dailymotion.com/video/x74iq0_compromising-electromagnetic-emanat_tech

This video shows how using a strong antenna, sitting in a van outside your home, the FBI could be picking up on your keystrokes on a **wired** keyboard. In fact many people speculate that the new smart meters installed in many homes already have this technology to determine everything you are doing in your home electronically. Wired and wireless keyboards emit electromagnetic waves, because they contain electronic components. This electromagnetic radiation could reveal sensitive information such as keystrokes as shown in the video. Every electromagnetic wave is unique to the device using it, which gives a person spying on you the ability to tell the difference between you using your computer versus the dishwasher.

According to the people who did this experiment, they were able to extend the range up to 20 meters using relatively cheap technology. This was for wired keyboards by the way, and they go on to explain that wireless keyboards and mice are even easier. Which brings us to another area of interest, wireless transmissions. Things like wireless keyboards and wireless mice (or mouses?) are vulnerable to eavesdropping as well. If they are not using a strong enough encryption to send data to the receiver, anyone can be listening in on your keystrokes and mouse activity. Probably something most people never thought about either, this is on top of the electromagnetic waves that can also be picked up.

Quote

Microsoft has upgraded the weak encryption found on today's mass-market wireless keyboards with a new design that uses 128-bit AES to secure communication to and from the PC.

Hitherto, keyboard encryption has been weak, with keys chosen from a small palette of possibilities, **with one hacking group claiming in 2009 that it had developed a tool specifically to sniff keystrokes from Microsoft keyboards at a range up to a 10 metres.**

<http://news.techworld.com/security/3284218/new-microsoft-wireless-keyboard-gets-128-bit-encryption/>

Are you using wireless technology? How old is it? Might be time to upgrade your equipment. 10 meters is about 33 feet, but remember the technology available to the government could potentially reach beyond that. Then there are other things people forget such as wireless monitors which broadcast your screen to a receiver that can be picked up. Just think about the old antennas people used to have on top of their homes, and how far away those could pick up signals from TV stations, if you had one of those pointed at you in a van across the street, there is no doubt they could be eavesdropping on your activities inside.

One researcher was able to use a wireless signal sent by a smart meter from up to 300 meters away

(900 feet) to find out which house it was coming from and what the current power consumption was in plain text. She was then able to use this information to determine when people were and were not home based on average spikes in consumption since the meters pulse every 30 seconds.

Quote

The data sent was in plain text and carried the identification number of the meter and its reading. The name of the home owner or the address aren't included, but anyone motivated enough could quickly figure out the source.

"The meter ID was printed on the front of the meter we looked at, so theoretically you could read the ID [off a target meter] and try to sniff packets," Xu said.

In her tests, Xu found she was able to pull packets out of the air from target meters between once every 2 to 10 minutes. That's fast enough to be able to work out the average power consumption of a house and notice start to deduce when someone is at home.

<https://www.networkworld.com/news/2012/110512-smart-meters-not-so-clever-263977.html>

Things like automatic timers that flip switches might be worth investing in to always make it look like someone is home until security researchers start looking into ways to avoid the wide open door we are giving to anyone who wants to find data about us.

What can you do about these types of eavesdropping? Not a whole lot unless you want to start turning into a tin-foil hat type of person. There are some fun things you can do if you want to go crazy with it though as recommended by the following site.

<http://www.lessemf.com/smart.html>

Quote

Y-SHIELD

YShield High Frequency Shielding Paint

Easy to apply water-based paint for walls, ceilings, doors and other interior OR exterior surfaces. Very effective for blocking cell phone signals, CB, TV, AM, FM signals, radiofrequency radiation and microwaves. Tested highly effective up to 18 GHz!

<http://www.lessemf.com/paint.html#290>

There are lots of other things on there as well like drapes, curtains, garments, fabrics and so forth which disrupt the transmission of these signals. It is completely up to you what you want to do, I am just giving you the options and the education so you can make an educated decision of how far you want to go to protect your privacy.

COOKIES & JAVASCRIPT REVISITED, PLUS FLASH COOKIES AND OTHER BROWSER TRACKING

Your browser can reveal an alarming amount of information about you.

Surprisingly enough, or not too surprising, when you visit a website there is a surprisingly large amount of identifying data being sent to the website you are communicating with.

Cookies

Cookies are pieces of information that a web site can send to your browser. If your browser "accepts" them, they will be sent back to the site every time the browser accepts a page, image or script from the site. A cookie set by the page/site you're visiting is a "second party" cookie. A cookie set by another site that's just providing an image or script (an advertiser, for instance), is called a "third party" cookie.

Cookies are the most common mechanisms used to record the fact that a particular visitor has logged in to an account on a site, and to track the state of a multi-step transaction such as a reservation or shopping cart purchase. As a result, it is not possible to block all cookies without losing the ability to log into many sites and perform transactions with others.

Unfortunately, cookies are also used for other purposes that are less clearly in users' interests, such as recording their usage of a site over a long period of time, or even tracking and correlating their visits to many separate sites (via cookies associated with advertisements, for instance).

With recent browsers, the cookie setting that offers users the most pragmatic tradeoff between cookie-dependent functionality and privacy is to only allow cookies to persist until the user quits the browser (also known as only allowing "session cookies"). Tails does this automatically by the way with Iceweasel.

Recent Cookie-Like "Features" in Web Browsers

In addition to the regular cookies that web browsers send and receive, and which users have begun to be aware of and manage for privacy, companies have continued to implement new "features" which behave like cookies but which are not managed in the same way. Adobe has created "Local Stored Objects" (also known as "Flash Cookies") as a part of its Flash plug-ins; Mozilla has incorporated a feature called "DOM storage" in recent versions of Firefox. Web sites could use either or both of these in addition to cookies to track visitors. It is recommended that users take steps to prevent this.

Managing Mozilla/Firefox DOM Storage Privacy. If you use a Mozilla browser, you can disable DOM Storage pseudo-cookies by typing about:config into the URL bar. That will bring up an extensive list of internal browser configuration options. Type "storage" into the filter box, and press return. You should see an option called dom.storage.enabled. Change it to "false" by right-clicking and choosing Toggle.

Managing Adobe Flash Privacy.

Adobe lists advice on how to disable Flash cookies on their website.

<http://helpx.adobe.com/flash-player/kb/disable-local-shared-objects-flash.html>. There are some problems with the options Adobe offers (for instance, there is no "session only" option), so it is probably best to globally set Local Stored Object space to 0 and only change that for sites which you are willing to have tracking you. On the Linux version of Adobe's Flash plugin there does not seem to be a way set the limit to 0 for all sites and therefore its use should be limited or avoided. Luckily Tails does not have flash installed, but in case you are not using Tails be aware of this.

If you absolutely need to watch a video online, find a way to download the video to your computer and watch it that way. This takes the browser out of the loop of processing a video for you and eliminates those Flash cookies which help identify you.

Javascript

Javascript is probably the grand daddy of all vulnerabilities in internet browsing. The majority of exploits, malware, viruses and other computer take overs happen because of Javascript code executing in your browser. Javascript has many uses. Sometimes it is simply used to make webpages look flashier by having them respond as the mouse moves around or change themselves continually. In other cases, javascript adds significantly to a page's functionality, allowing it to respond to user interactions without the need to click on a "submit" button and wait for the web server to send back a new page in response.

Unfortunately, javascript also contributes to many security and privacy problems with the web. If a malicious party can find a way to have their javascript included in a page, they can use it for all kinds of evil: making links change as the user clicks them; sending usernames and passwords to the wrong places; reporting lots of information about the users browser back to a site. Javascript is frequently a part of schemes to track people across the web, or worse, to install malware on people's computers. It is best to disable Javascript (about:config in URL bar search for Javascript and Toggle it to disabled) unless you absolutely trust the site or use the browser add-on NoScripts that comes with Tails and is available in Firefox to at least selectively block malicious scripts. Disabling Javascript outright is the best option though, and gumby has added a suggestion that can make it even easier to do this.

Quote from: gumby on January 14, 2014, 08:59:57 pm

Supposedly NoScript doesn't block all Javascript even when it is enabled and no sites are on the whitelist. Not sure about that claim but I've seen people make it. There's a Firefox add-on (which also works in Tor Browser) called `toggle_js` which lets you toggle the `about:config javascript.enable` parameter through a toolbar icon so you don't have to go into `about:config`. I find it quite useful.

Javascript can also reveal an alarming amount of information about you even if you are using TOR or a VPN, including your browser plug-ins, your time zone, what fonts you have installed (flash does this as well) and of course most browsers will send your user agent, meaning they tell the website what browser you are using and in some cases your operating system! Some of these details may not seem very important, but collected as a whole, it can make it easier to identify who you are online by almost generating a finger print of you with your specific settings related to your browser. Then as you hop

around from site to site with your finger print, correlations and patterns can be drawn from this and eventually linked to you if you are not extremely careful.

Luckily, Tails and Whonix overrides the majority of this identifying information, so as long as you use Tails with Javascript disabled, or at the very least with NoScripts (Flash is disabled automatically) then you can cut down on the amount of information you share. Needless to say, it is not always possible to browse with Tails, so these are things you need to be aware of when you are browsing with regular browsers on your native OS with your browser of choice.

See what your browser is revealing about you at this page below. Do not visit it from your real IP address, since this page will be linked to the Silk Road forums from the moment I make this post part of my thread. As a result, you may wish to search online for other sites that check what information your browser is revealing about you. If you are confident in your OpSec abilities, use the one below.

<http://browserspy.dk/>

A FEW RECOMMENDATIONS

Here are a few recommendations that may slip by the average user on these forums.

1. Never leave your computer that you use for your freedom fighting unattended.

This may seem like a no-brainer, but if you have kids, or a spouse or a sibling that does not understand what you do on the computer and they decide to hop on your account and sign into their email, Facebook or doing things that could compromise your location while on that computer because they simply did not know, this could potentially cause you problems.

Maybe you are connecting through multiple layers like this TOR -> VPN(1) -> TOR -> VPN(2), so that is 4 layers and VPN(2) is the IP address that everyone sees. Then your child or spouse goes on to their email with that IP address, then signs off without your knowledge. That VPN is now linked to you. And we remember how when under pressure, companies will likely give out information about their customers to avoid fines, shut downs and prosecution.

2. Do not tell your family members what you are doing, just instruct them not to touch your computer. Keep it passworded.

- You should never tell anyone what you are doing on your computer because if law enforcement ever did show up, they would question your family and friends about you. If they honestly do not know, then they cannot be held in contempt of court, so it is better to keep them in the dark. Or maybe the police might scare them into giving up all your secrets because they tell your family that if they do not confess that yourself and them will be going to jail, possibly for a long time. Just password your computer and never leave it unattended with the screen unlocked.

3. If you use multiple layers to connect, make sure you regularly check to make sure all your layers are in tact.

VPNs can drop sometimes without warning and while you should never set yourself up so that if one layer drops you lose everything, just keep in mind when one drops that you may need to adjust the

way you handle yourself online until you get that next layer up. This is one of the reasons I like Tortilla so much, if my TOR layer does not work, it does not bypass it and go to my next layer, instead it just stops working altogether. When VPNs drop, your computer bypasses the dropped VPN and moves onto the next layer, which in some cases could be your real IP address. Just something to keep in mind.

4. **Do not use the same password for multiple forums, marketplaces, emails and so forth.** - Expect that one or more of the websites you are registered with is storing your password in plain text. This means that if somebody finds an exploit in the software and is able to dump the entire database, they can find your password. And if you used the same password for other sites, and god forbid with the same username as well, your entire list of accounts is compromised. Always use different passwords and keep them strong. Do not let anything about your password identify how you choose passwords, or identify anything personal about you.

COLD BOOT ATTACKS, UNENCRYPTED RAM EXTRACTION

Did you know that even if your system is whole disk encrypted, your data can still be extracted using something called a cold boot attack? Read on.

The first thing we need to talk about is RAM. RAM stands for random access memory. All you need to know about RAM is that RAM is the place in a computer where the operating system, application programs, and data in current use are kept so that they can be quickly reached by the computer's processor. RAM is much faster to read from and write to than the other kinds of storage in a computer, the hard disk, floppy disk, and CD-ROM. However, the data in RAM stays there only as long as your computer is running. When you turn the computer off, RAM loses its data.

When you turn your computer on again, your operating system and other files are once again loaded into RAM, usually from your hard disk. RAM can be compared to a person's short-term memory and the hard disk to the long-term memory. The short-term memory focuses on work at hand, but can only keep so many facts in view at one time. If short-term memory fills up, your brain sometimes is able to refresh it from facts stored in long-term memory. A computer also works this way. If RAM fills up, the processor needs to continually go to the hard disk to overlay old data in RAM with new, slowing down the computer's operation. Unlike the hard disk which can become completely full of data, RAM never runs out of memory.

Data can be extracted from the RAM using various tools. When you have a text document open and you are working on it, you are working from the RAM. Meaning that if you are working on a sensitive document, that document is temporarily stored in the RAM and is vulnerable to being extracted while the computer is on. When RAM is being stored, it is being stored **without** any form of encryption, making it very easy to steal and a huge security risk.

Shutting down a computer through its normal shutdown cycle usually goes through a process of clearing the RAM. However, if the computer loses power abruptly like in a power outage, the computer does not go through its normal shut down cycle and some information remains on the RAM chips for a few seconds up to a few minutes. This is one of the ways cold boot attacks can work.

I also want to quickly introduce a type of RAM to you which will help you understand the rest of this article better. Below is a research paper and they used a type of ram called DRAM. DRAM stands for **dynamic random access memory**. DRAM is the most common kind of random access memory (RAM) for personal computers and workstations. DRAM is dynamic in that, unlike static RAM (SRAM), it needs to have its storage cells refreshed or given a new electronic charge every few milliseconds. DRAM is designed to lose its memory quickly after losing power. Then there are subsections of DRAM called DDR. This is a way of making the memory more quickly available, but it is not really important to fully understand. Wikipedia can give you all you need to know about DDR. In this article we are focusing on just the concept of DDR, DDR2 and DDR3.

These are newer versions of DRAM that keep getting better, and I believe we are currently up to DDR4. But most computers circulating around today have DDR2 and DDR3 in them unless they are older computers, this includes laptops. DRAM is known as a type of volatile memory, it is computer memory that requires power to maintain the stored information. It retains its contents while powered, but when power is interrupted, stored data is quickly lost. But how quickly is it lost?

In 2008, a group of researchers wanted to see the practicality of extracting unencrypted data from the RAM in your computer. They argued that DRAMs used in most modern computers retain their contents for seconds to minutes after power is lost, even at operating temperatures and even if removed from a motherboard. And by using an analysis tool they were able to search for key files (such as PGP keys) held in the RAM that could be used to decrypt encrypted volumes (drives) on your computer. They successfully were able to decrypt volumes using BitLocker, FileVault, dm-crypt, and TrueCrypt. Below is the abstract of their research.

Quote

Lest We Remember: Cold Boot Attacks on Encryption Keys

Abstract Contrary to popular assumption, DRAMs used in most modern computers retain their contents for seconds to minutes after power is lost, even at operating temperatures and even if removed from a motherboard. Although DRAMs become less reliable when they are not refreshed, they are not immediately erased, and their contents persist sufficiently for malicious (or forensic) acquisition of usable full-system memory images. We show that this phenomenon limits the ability of an operating system to protect cryptographic key material from an attacker with physical access. We use cold reboots to mount attacks on popular disk encryption systems — BitLocker, FileVault, dm-crypt, and TrueCrypt — using no special devices or materials. We experimentally characterize the extent and predictability of memory remanence and report that remanence times can be increased dramatically with simple techniques. We offer new algorithms for finding cryptographic keys in memory images and for correcting errors caused by bit decay. Though we discuss several strategies for partially mitigating these risks, we know of no simple remedy that would eliminate them.

<https://citp.princeton.edu/research/memory/> [Abstract]

<http://citpsite.s3-website-us-east-1.amazonaws.com/oldsite-htdocs/pub/coldboot.pdf> [Full Text]

Here is an FLV video you can download to watch exactly how they did it.

<https://anonfiles.com/file/97b5737dba6b96871fd862b8a587b8f0>

This was very troubling to most people, and had many people freaking out when the research paper was released back in 2008 because even tough encryption tools like TrueCrypt could be rendered useless with an attack like this. Upon further analysis of the paper, I wanted to note that they used SDRAM, DDR and DDR2, and not DDR3 because it was not available at that time. This prompted TrueCrypt to release the following statement on their website.

Quote

Unencrypted Data in RAM

It is important to note that TrueCrypt is disk encryption software, which encrypts only disks, not RAM (memory).

Keep in mind that most programs do not clear the memory area (buffers) in which they store unencrypted (portions of) files they load from a TrueCrypt volume. This means that after you exit such a program, unencrypted data it worked with may remain in memory (RAM) until the computer is turned off (and, according to some researchers, even for some time after the power is turned off*). Also note that if you open a file stored on a TrueCrypt volume, for example, in a text editor and then force dismount on the TrueCrypt volume, then the file will remain unencrypted in the area of memory (RAM) used by (allocated to) the text editor. This applies to forced auto-dismount too.

Inherently, unencrypted master keys have to be stored in RAM too. When a non-system TrueCrypt volume is dismounted, TrueCrypt erases its master keys (stored in RAM). When the computer is cleanly restarted (or cleanly shut down), all non-system TrueCrypt volumes are automatically dismounted and, thus, all master keys stored in RAM are erased by the TrueCrypt driver (except master keys for system partitions/drives — see below). However, when power supply is abruptly interrupted, when the computer is reset (not cleanly restarted), or when the system crashes, TrueCrypt naturally stops running and therefore cannot erase any keys or any other sensitive data. Furthermore, as Microsoft does not provide any appropriate API for handling hibernation and shutdown, master keys used for system encryption cannot be reliably (and are not) erased from RAM when the computer hibernates, is shut down or restarted.**

To summarize, TrueCrypt cannot and does not ensure that RAM contains no sensitive data (e.g. passwords, master keys, or decrypted data). Therefore, after each session in which you work with a TrueCrypt volume or in which an encrypted operating system is running, you must shut down (or, if the hibernation file is encrypted, hibernate) the computer and then leave it powered off for at least several minutes (the longer, the better) before turning it on again. This is required to clear the RAM.

* Allegedly, for 1.5-35 seconds under normal operating temperatures (26-44 °C) and up to several

hours when the memory modules are cooled (when the computer is running) to very low temperatures (e.g. -50 °C). New types of memory modules allegedly exhibit a much shorter decay time (e.g. 1.5-2.5 seconds) than older types (as of 2008).

** Before a key can be erased from RAM, the corresponding TrueCrypt volume must be dismounted. For non-system volumes, this does not cause any problems. However, as Microsoft currently does not provide any appropriate API for handling the final phase of the system shutdown process, paging files located on encrypted system volumes that are dismounted during the system shutdown process may still contain valid swapped-out memory pages (including portions of Windows system files). This could cause 'blue screen' errors. Therefore, to prevent 'blue screen' errors, TrueCrypt does not dismount encrypted system volumes and consequently cannot clear the master keys of the system volumes when the system is shut down or restarted.

<http://www.truecrypt.org/docs/unencrypted-data-in-ram>

A few key points to extract from here are that properly shutting down your computer reduces, if not completely eliminates this risk except in the case of encrypted system disks. What is meant by this is, for example, if your main operating system is Windows and you have encrypted that drive, this is your system drive and the master key for that drive is not cleared upon shutdown or restart. The solution is simply to never store anything sensitive on your system volume. Whether you use a partitioned drive or a USB stick that is encrypted, just make sure that your main drive that is booted into does not contain sensitive data. And if you have no other choice, then you need to separately encrypt the data inside the system volume with a different passphrase and private key so that even if they get into your system volume, they cannot access the other encrypted data you want to protect.

They can use these same techniques to sniff around for your PGP private key files in the RAM, so this is a very real threat in the case that if your computer is still powered on if they come to get you, they can use these techniques to retrieve data from your computer. However, there is a debate about whether or not this type of attack can persist even now into 2014 with newer types of RAM. I point to a random blog online and I make no judgement as to whether or not this is a legitimate claim, but it is interesting nonetheless.

Quote

Now to test the actual cold-boot attack. Fill memory with around 1000 taint markers, just to be sure there are enough.

Now shut down. Ostensibly, the markers could be recognizable in RAM after whole minutes, but I'm impatient, so I just waited 10 seconds for the first test. Boot up, into the minimal linux installation. Load the kernel module: `insmod ./rmem.ko`. Run hunter.

Nothing.

That's ok, though. There should be at least some data corruption. The default marker size is 128 bytes, so let's set the hamming distance to 128, meaning that one bit out of every byte is allowed to be flipped. (Statistically, that's equivalent to a 25% corruption rate, since a corrupted bit has a 50% chance

of remaining the same).

Nothing.

Looks like in 10 seconds, memory was completely corrupted. Let's try a shorter interval: 2 seconds. Same results. Nothing is left of our "encryption key".

<http://bytbox.net/blog/2013/01/cold-boot-attacks-overrated.html>

The user claimed to be using a newer type of RAM called **DDR3**, which is known to hold memory for a much shorter time than DDR2. And a newer research paper released in September 2013 tried to reproduce the findings of the 2008 research but using computers with DDR1, DDR2 and DDR3 and their findings were interesting.

Quote

Even though a target machine uses full disk encryption, cold boot attacks can retrieve unencrypted data from RAM. Cold boot attacks are based on the remanence effect of RAM which says that memory contents do not disappear immediately after power is cut, but that they fade gradually over time. This effect can be exploited by rebooting a running machine, or by transplanting its RAM chips into an analysis machine that reads out what is left in memory. In theory, this kind of attack is known since the 1990s. However, only in 2008, Halderman et al. have shown that cold boot attacks can be well deployed in practical scenarios. In the work in hand, we investigate the practicability of cold boot attacks. **We verify the claims by Halderman et al. independently in a systematic fashion. For DDR1 and DDR2, we provide results from our experimental measurements that in large part agree with the original results. However, we also point out that we could not reproduce cold boot attacks against modern DDR3 chips. Our test set comprises 17 systems and system configurations, from which 5 are based on DDR3.**

https://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6657268&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxppls%2Fabs_all.jsp%3Farnumber%3D6657268

So what does should you do? Number one, always shut down your computer when you are not around it or put it into hibernation mode, otherwise your sensitive documents could be lingering around in your RAM. Simply locking the screen will do you no good. Make sure your computer is using a DDR3 type of RAM, if possible. Some of you this means you need to upgrade. If you are unsure what kind of RAM your computer has, search online to find a tool that will detect it for you. Never store anything sensitive on an encrypted **system** volume, because this attack can be used to break into the volume and anything unencrypted can be retrieved. If you are using a laptop, pull the battery out so that if you need to quickly pull the power, it will turn it off immediately. If you have time, shut down the computer, otherwise turn it off immediately so that it is not running. The more time you can waste are precious seconds where they cannot retrieve any data. So immediately shut things off if you do not have enough time to do a proper shutdown.

Consider putting a lock on your computer case, and if you want to go take it a step further, bolt it to the floor. That way the amount of time it would take them to get inside your computer would waste

valuable minutes and more than likely render any recoverable memory useless. Some people have even suggested that you solder the RAM into the motherboard so they cannot take it out. This may help slow things down, but remember that cooling the memory down can preserve things for quite a while if you are using DDR1 or DDR2. With DDR3, you should be good to go and I believe with this realization, manufacturers will likely start looking at ways to encrypt RAM, but until that time you do need to be aware of this as a possible means for stealing your sensitive data and something you should keep in the back of your mind and prepare yourself for just in case.

THE STRENGTH OF CRYPTOGRAPHY AND ANONYMITY WHEN USED PROPERLY

This post is meant to serve as an example of how, when cryptography and anonymity is used properly, you can evade just about anybody including the police.

By now, everyone has likely heard of someone getting locked out of their computer and being forced to pay by the attacker to have it unlocked, this is CryptoLocker. Dell SecureWorks estimates that CryptoLocker has infected 250,000 victims. The average payout is \$300 each, and millions in laundered Bitcoin have been tracked and traced to the ransomware's money runners.

CryptoLocker is a ransomware trojan which targets computers running Microsoft Windows[1] and first surfaced in September 2013. A CryptoLocker attack may come from various sources; one such is disguised as a legitimate email attachment. A ZIP file attached to an email message contains an executable file with the filename and the icon disguised as a PDF file, taking advantage of Windows' default behaviour of hiding the extension from file names to disguise the real .EXE extension. When activated, the malware encrypts certain types of files stored on local and mounted network drives using RSA public-key cryptography to generate a 2048-bit RSA key pair, with the private key stored only on the malware's control servers.

The malware then displays a message which offers to decrypt the data if a payment (through either Bitcoin or a pre-paid voucher) is made by a stated deadline, and threatens to delete the private key if the deadline passes. If the deadline is not met, the malware offers to decrypt data via an online service provided by the malware's operators, for a significantly higher price in Bitcoin.

Dell SecureWorks estimates that CryptoLocker has infected 250,000 victims. The average payout is \$300 each, and millions in laundered Bitcoin have been tracked and traced to the ransomware's money runners. In November 2013, the operators of CryptoLocker launched an online service which claims to allow users to decrypt their files without the CryptoLocker program, and to purchase the decryption key after the deadline expires; the process involves uploading an encrypted file to the site as a sample, and waiting for the service to find a match, which the site claims would occur within 24 hours. Once a match is found, the user can pay for the key online; if the 72-hour deadline has passed, the cost increases to 10 Bitcoin.

To date, no one has successfully defeated CryptoLocker. The Swansea, Massachusetts police department was hit in November. The officers paid CryptoLocker's ransom. Police Lt. Gregory Ryan told

press that his department shelled out around \$750 for two Bitcoin on November 10. One of the reasons I am posting this, is that CryptoLocker uses 2,048 RSA encryption, and if you remember in the PGP posts earlier in this thread I recommended to use 4096. Even with 2,048 bit encryption, no one has successfully defeated CryptoLocker, and this is the power of properly implemented cryptography.

And, using the proper methods of anonymity, this person or group has managed to acquire, according to research done by ZDNet, around 41,928 BTC.

<http://www.zdnet.com/cryptolockers-crimewave-a-trail-of-millions-in-laundered-bitcoin-7000024579/>

Quote

In research for this article ZDnet traced four bitcoin addresses posted (and re-posted) in forums by multiple CryptoLocker victims, showing movement of 41,928 BTC between October 15 and December 18.

Based on the current Bitcoin value of \$661, the malware ninjas have moved \$27,780,000 through those four addresses alone - if CryptoLocker cashes out today.

If CryptoLocker's supervillans cash out when Bitcoin soars back up to \$1000, like it did on November 27... Well, \$41.9 million isn't bad for three months of work.

As you can see, properly executed cryptography and anonymity allowed this group of people acquire the Bitcoin equivalent of almost \$42 million in just now 4 months at the time of this post. I am not recommending or advocating that you do this, but just giving you a perfect example of how powerful the combination of these two very important factors are in protecting anybody online when used properly.

ANOTHER SCAM EMAIL - BEWARE

If you have been following my thread for a while now, you will remember the previous email scam going around trying to get people to download an infection version of tor. With Silk Road at the time of this post now boasting over 25,000 members, it is easy to get that these occurrences are likely going to increase and unfortunately some people are going to fall for them. This new one is directed at vendors, but it nonetheless should serve as an example of the type of scams that people are going to be trying to pull on members of this forum and other forums.

Quote

Dear Valued Vendor,

Due to the recent instability of the site, and our programmers inability to remedy the problems in a timely manner, we are going to have to temporarily shut down vendor accounts. Since we can't just stop operation of the site completely, we are forced to develop a way for only some of the vendors to go into a temporary vacation mode. In need of recent server upgrades, as well as this new method we are implementing, it has occured to us that the only way to pick which vendors are going to remain in

business is by how much sales/profit they are doing, as well as how much being a vendor on our site means to them. Here's how this is going to work:

If you would like to keep vending on the site during our upgrades/repairs, we are going to require that you pay an additional .3BTC bond to us. If you pay this .3BTC bond, your account will remain active and you will keep vending while we work to fix the problems. If you do not pay this .3BTC bond, your account will be temporarily put into vacation mode status and you will be unable to vend until we locate and remedy the problem. We are very sorry for these changes!

In the event you do pay the bond, as soon as the vending opens back up to everyone, you will have your .3BTC bond returned, and you will receive a premium vendor account status. You will have a title on your page that displays you as "Hardcore Vendor". We are terribly sorry we have to ask this of all our hardworking vendors, but there is really no other way for us to decide WHO gets to keep vending and who has to wait until we fix things.

Our team is working hard at the problem, and we estimate it will be no longer than a week for the changes to be made and vending to open back up to everyone.

Vendors who will pay bond: Please send .3BTC to BTC Address:
1NbEs2rJgreRUvjp9o7hUWo3akeLA3Efy

Vendors who are unable to pay bond: Your accounts will go into vacation mode at 12:01AM UTC February 2nd.

Let us never forget this recent hurdle in our battle for freedom. But let us not allow it to stop our fight, either – it is now time to simply pick ourselves back up, dust ourselves off, and continue fighting this revolution like we've never fought it before.

I'm proud to have you all at my side.

Yours Loyally

Dread Pirate Roberts

The user who sent out this message actually used the name **Dread Pirates Robert**, which is similar but not correct. One thing you should be aware of, is that any type of announcement like this from a high ranking Administrator like DPR will always be signed with their PGP signature. And remember, we discussed how to verify these signatures in a previous post. I remember when a moderator named **Sarge** was in charge of vendor bonds, there was a user with the name Sarrge (two r's) that was trying to scam vendors into sending their bonds to his address instead and unfortunately, several people fell for this scam.

Please always check if there is a PGP signature, and if there is not, kindly ask the Administrator or Moderator to resend the message to you using a signature. Protect yourself by verifying the name and make sure this user has an Administrator or Moderator status on the forum. Be safe!

AN INTRODUCTION TO AN EXPERT ON OPSEC, PLUS MD5 & SHA-1 CHECKSUMS

This post, I would like to focus on introducing you to an expert in the field of OpSec.

*Note this message contains a download, therefore this message has been **PGP signed** to ensure that if this message is altered, you will be aware of it.*

This is a man who has done several public presentations, yet, many people still do not know about him. OpSec stands for Operations Security and in this context refers to people keeping themselves anonymous online. He goes by the online handle, "The Grugq", and Grugq has his own blog which can be found at the following webpage.

<http://grugq.github.io/>

It should be noted that Grugq was at one time on the payroll of the US government for finding and selling zero day exploits. If you remember the previous post about how the US federal government is the singlemost purchaser of malware in the world, well Grugq was one of those who sold malware to the government. Unfortunately for him, when he went public about it, they no longer wanted to buy malware from him because they like to maintain their own anonymity when purchasing these exploits. And here is a short biography from an online website.

Quote

Biography:

The Grugq is an Information Security Professional who has has worked with digital forensic analysis, binary reverse engineering, rootkits, Voice over IP, telecommunications and financial security. He has reported to be an exploit broker for 15% of the sale. Last but not least, he has also spoken at various security conferences.

Facts

He developed "userland exec"

He is the author of Hash (hacker shell), a tool to enable people to evade detection while penetrating a system.

He has released a voip attack software.

Claims to have made mad loot on being an exploit broker (middleman).

<https://www.soldierx.com/hdb/Grugq>

Why are we talking about the Grugq? Who cares? Well, he has some of the best information on keeping yourself anonymous and maintaining privacy online and he is somebody who you should all familiarize yourselves with. He writes blog posts, and he has done video presentations at security and hacker conferences, with his most famous presentation, at least in the world of Silk Road being the one he did on OpSec. Since I know it is hard for Tails users to watch videos on YouTube, I decided to download it from YouTube and upload it to AnonFiles.com so you all can watch it. The presentation is about 1 hour long, and an essential to everyone who wishes to maintain their anonymity online. Remember, you only have to screw up once.

<https://anonfiles.com/file/b6de41da8d1fca2fabf725f79d2a90df>

SHA1 Sum: 1a9e6c67a527b42a05111e1b18c7a037744bb51e

MD5 Sum: b6de41da8d1fca2fabf725f79d2a90df

Once you have downloaded the file, I want you to check something called the checksum of the file. The checksum is where the contents of the entire file get plugged into a mathematical algorithm and output a specific string. You can see the two strings above. This is something you should all get into the habit of doing when possible is verifying the checksum of your files. If you remember when we talked about signature files and PGP, this is another method of verifying your downloads but not as good as the signature files. It should however, whenever provided be performed to verify your downloads when the signature file + PGP combination is not available.

Once you have downloaded the file in Tails, the first thing you should do, is move the file you downloaded to your tmp folder. In order to do this, look up at the top and click **Places -> Computer -> File System -> tmp**. This is where you move the file your downloaded to, and to keep things easier, rename the file **grugq.zip** and you will see why you want to do that in a second.

Next we are going to open a terminal window (like a DOS prompt) by clicking the black rectangle icon in the upper left center area of Tails. Once you have opened your terminal window, we are going to perform some Linux commands.

cd /tmp - This will change the current directory you are operating within the terminal to your tmp folder and allow you to more easily access the files in that folder.

sha1sum grugq.zip - This will perform a SHA1 checksum on the file you just downloaded, and you can see why you wanted to rename the file. It should give you the same output as the SHA1 sum listed above.

md5sum grugq.zip - This will perform an MD5 checksum on the file you just downloaded, and is another way of checking the file. SHA1 is better because it is harder produce the same output twice with different file contents using SHA1 versus MD5, but nonetheless, use both whenever possible and always check your downloaded files.

Ok, assuming your downloaded video passed the checksum test, you can be assured that the video file that I uploaded has not been tampered with, or had any malicious code injected into it. When even a single character is changed in the source code of a given file, the checksum output will be completely different. Most people think it may be off by a few characters, but the difference is always quite large and is why performing checksums is an important way of verifying your downloads.

Since you now have a 1 hour video presentation that you all need to watch and rewatch (You can do this in Tails), I will end this post and continue with my next post from the assumption that you can completed watching this highly recommended and endorsed (by SR administrators and moderators) video on OpSec. We will start looking more into the recommendations from the Grugq. He will be an invaluable resource of information for us, and I will mainly be translating some of his posts into a more understandable format for those of you who are less technically capable and also keeping them on the Silk Road forum hidden services.

-----BEGIN PGP SIGNATURE-----

```
iQIcBAEBCgAGBQJS7wteAAoJEPuh6tSg81nyhyYP/OnFaWRq0GPe6/5XeMUj3yiZ
2fBaJ+7SXOMxnNXPZw9XAN5Hkpp9wPQmk8W27otuIk2N+iom8H0tJcGZi7hiMd45
DvONOrt/gS3bst/G37I+tPwDnWxb1pVNCS+3XnuLOo9IA7VdykU8tz6R+68kPB25
9lDguaUYVeGp2AJMezQ01LL60xQvv25TFLgiPrYD611bscVadckhSV5upXlbMW9+
WVzJG1mgY9gmUYQV6D5ErPGLvxm8cC+IVlzwgGHQPd3kq2QImQF3XJrXqWGPXd8d
ewkD6VnrU8yO6tVMCG57K1xO9a9zPYp6yN1IOe69IsRkK7g266D+cz6ldwt97/Vr
5jgu1Ook8dfFGA3Sxg+qpoARt5diWKchvmqbxRrnFdOtCAawH1+DgNcVdepi7agk
zhIES1drHdIM1uQ9Wg3vegCLrU3HDpRwwyWoSZxH4kxruU7aByOH5ZdAZw9JV6Lk
b5JzVjrvrhayXwiHPQnnjM50RT9jPH44PhNZCN4G7Ln2Rkb7qa/kS5sA4W2dRwXf
SjtYXf+18pCp/7NL09LD+LsabZHEAa/MilWxjsAnLLlrJsnw3YbSUola/ebmnlq8
oUW20yP0fDOHdeSGVq1uLNZladZHZtmZIGqBigPU3XAKLxYajssglAgcPxD8E4vc
rkKb3PIyz1k1/JXulymR
=zJvP
```

-----END PGP SIGNATURE-----

IT IS OBVIOUS WHEN YOU ARE USING TOR

This is going to be a short post about a mistake we can all learn from when a Harvard student emailed a bomb threat to his school while using tor to avoid a final exam.

<http://www.forbes.com/sites/runasandvik/2013/12/18/harvard-student-receives-f-for-tor-failure-while-sending-anonymous-bomb-threat/>

Quote

...the student “took steps to disguise his identity” by using Tor, a software which allows users to browse the web anonymously, and Guerrilla Mail, a service which allows users to create free, temporary email addresses.

Despite 20-year-old Eldo Kim’s goal of anonymity, his attempts to mask his identity led authorities right to his front door. Does that mean that Tor failed a user looking to delay his “Politics of American Education” exam? Not in the slightest.

While the Harvard student did indeed use Tor, it was his other sloppy security measures that led to his arrest. The complaint says the university “was able to determine that, in the several hours leading up to the receipt of the e-mail messages ... **Eldo Kim accessed Tor using Harvard’s wireless network.**”

What Kim didn’t realize is that **Tor, which masks online activity, doesn’t hide the fact that you are using the software.** In analyzing the headers of the emails sent through the Guerrilla Mail account, authorities were able to determine that the anonymous sender was connected to the anonymity network.

Using that conclusion, they then attempted to discern which students had been using Tor on the Harvard wireless network around the time of the threats. Before firing up Tor, Kim had to log on to the school’s wireless system, which requires users to authenticate with a username and password. By going through network logs and looking for users who connected to the publicly-known IP addresses that are part of the Tor network, the university was able to cross-reference users that were using both Tor and its wireless internet around the time the bomb threats were received.

There is not much for me to add other than the fact that, if you are planning on doing some freedom fighting, activism or just using Silk Road, make sure that you are able to do so where using tor is not going to raise some flags. In the case of this student, he was likely the only student at Harvard using tor at the moment this email was sent, and when the authorities came to his dorm he quickly admitted he was responsible.

He likely never would have been caught, but remember when you use tor, others can be aware that you are using it. A better idea for him would have been to connect to another computer remotely and have that computer connected to tor to send the email. This way, they never could have seen his computer connected to tor. I would not worry about using tor on a regular basis from your home, because there are hundreds of thousands of tor users, but it is again, something to be aware of. tor will not cover your bad OpSec mistakes like in the case of Eldo Kim.

ARE YOU USING SAFE-MAIL.NET ?

A recent article on Forbes.com talks about a false sense of security users may have when using Safe-Mail.net

<http://www.forbes.com/sites/runasandvik/2014/01/31/the-email-service-the-dark-web-is-actually-using/>

If you are a user of Silk Road, you have likely seen many users advocating the use of a service called Safe-Mail.net. This company describes itself as "the most secure, easy to use communication system", and many Silk Road users have adopted it. But there are some things you should be aware of.

Quote

Known users of the Safe-mail web service include operators, vendors and customers of the dark web's many drug market sites, journalists writing about the investigation into Silk Road, and BTCKing, the vendor who ran an underground anonymous Bitcoin exchange and allegedly worked with BitInstant CEO Charlie Shrem to sell more than \$1 million worth of Bitcoins to users of Silk Road.

When I reached out to Safe-mail for comment, Amiram Ofir, Safe-mail's President and CEO, responded in an email that the company and its employees "certainly are not aware of any criminal activity," adding that the company does "follow court orders that are issued in Israel by an Israeli court. Any other law enforcement agency should contact the Israeli authorities." It's worth noting, however, that Israel signed a Mutual Legal Assistance Treaty (MLAT) with the U.S. in 1998. An MLAT request was used to image the Silk Road web server, according to the criminal complaint of Sept. 27, 2013.

Ofir told me that communications between users and the web service are SSL protected, and that information stored on the server is encrypted with user-specific keys. **When asked if Safe-mail has received court orders issued by an Israeli court on behalf of a non-Israeli law enforcement agency, such as the FBI, Ofir replied with a short "Yes." My followup email, asking if Safe-mail has the ability to decrypt information without a user's key, went unanswered.**

So, the first time to note is that the FBI is already aware of Safe-Mail.net and is already receiving court orders from non-Israeli law enforcement agencies. And they are likely giving them everything they need in order to read the emails. Therefore, you should remember that no email service should be trusted. No email service is going to go to jail for you. And if you are sending anything sensitive over email using plain text, it will likely be read eventually by somebody other than the intended recipients. This is why things such as strong PGP encryption are essential to any type of sensitive communication.

With this, it should be noted that Safe-Mail is no safer than Gmail when it comes to protecting your privacy with its centralized email service. Never trust any company with your privacy, always encrypt.

LOCALBITCOINS PART 1 - POLICE ARE WATCHING IT!

I have a few stories to share from people who used LocalBitCoins to sell their Bitcoins.

Quote

In September and October, I sold 213 BTC (gradually) to some random guy on localbitcoins. Everything went fine, each time I got the money, I sent the bitcoins. 5 days after the last transaction, I get arrested by the police. "Where does this money come from?" I explain about bitcoins, and tell them all I know about the random guy, I volunteer my phone to analyse my emails and check my story. Once they were

sure that the guy contacted me and not the other way around, I was finally free to go. Later they told me that the money was stolen and they thought I was doing money laundering.

Now after almost 3 months and a lot of back and forth with the police, they are now suggesting that I send back the money. I would gladly do that if they arrested the criminal and found out he can not repay. Right now if I send back the money, the innocent person who got his money stolen gets it back, but then I become the innocent person who got his money stolen, so that makes no sense to me.

Edit: I just saw a lawyer. According to him I already won the case. But it's going to cost me some serious money in lawyer's fees... More than my cumulated profits. I take that as the cost of a great life lesson and a wake-up call.

He also told me I can disclose the info that the police already knows. So here we go. I'm in Brisbane, Australia. The reason the police froze my account and not the criminal's account is that they wanted to know where the money was going. The police are regularly checking my house to make sure the criminal is not seeking revenge (he has my full address and I have 2 kids).

http://www.reddit.com/r/Bitcoin/comments/1to08d/arrested_by_the_police_for_localbitcoins_business/

This guy, likely a BTC miner, was arrested and questioned by police for selling BTC to a buyer over several transactions. They must have assumed that the buyer was using fraudulent funds and this shifted suspicion onto the seller as well. I do not know if this story is true, but I am tending to believe it is. Police are monitoring these transactions, so you better make sure you have a reasonable explanation as to where you obtained the Bitcoins you are trying to sell.

This next story was removed by the original poster (OP), but luckily somebody in the replies quoted the entire post and therefore I was able to grab it.

Quote

So, as a few of you guys know, I'm moving to another country soon enough. When I get over there I won't have access to my bank account, so a few weeks ago I decided it might be a good idea to sell some of my BTC for cash. I had done this a couple of times before and had a positive experience, so had no whims about doing it again.

So I received a request from someone who wanted to buy 500euro worth of BTC in a f2f transaction. I drove down to meeting spot, met the guy, he gave me the 500euro and basically ran back to his car and drove off. I obviously found this strange, but it was an escrow tx, so I released escrow from my phone and went back to my car.

On my drive back, I noticed that there was a Ford Mondeo behind me (the kind of car that is usually used by undercover police in my city). It seemed to be following me, I didn't have all my paperwork on my car in order, so I decided to take a detour down some local back-roads and shake it.

So anyways, I lost the car, drove home and thought nothing of this strange encounter.

Over the next few days, I noticed strange needle marks and tiny tears in all of my mail, I also noticed a really strange parked car outside my house one day, when I walked over to it to ask them what they were doing there, they drove off at speed. I probably should've been suspicious then, but I had done nothing wrong and shrugged it off.

A couple of days later, I wake up to the sound of my door being smashed in. I run down to find 5 police officers in my house. They showed me a search warrant under the misuse of drugs act. The national drugs unit were parked outside with sniffer dogs ready, they left after a few minutes though and didn't come inside with the dogs. **The police told me the person I met on localbitcoins was an undercover police officer, and they had copied the registration number off of my car and got my address from it.**

They stripped the whole house down, turned everything upside down looking for drugs. They found 1 joint of weed and they also seized a clock which they thought was a digital scale (it wasn't) and informed me that they were going to prosecute me for intent to supply, even though I wasn't selling, and I showed them a prescription from a doctor in another country (that isn't valid here) and told them the superintendent of the local police station had informally told me that they wouldn't prosecute me for possession if it was medical use even though I was technically breaking the law. They also found padded envelopes and accused me of selling drugs through the post (a complete lie with no evidence).

They then told me that if I didn't give them all the messages & phone numbers of everyone I had met to sell BTC that they were going to seize all my bitcoin miners, computers etc and have them "analyzed". I was about to move country in the next few days and didn't want the hassle of having to deal with this, so I told them that I had deleted all the messages (which I did) but that I would be able to get them back if they left my computers there, and that I would co-operate fully (I'm obviously not going to co-operate). They then left and I changed my flight date and basically fled the country the next day, luckily I was planning on moving in a week anyways.

So, a warning to you guys, be careful doing f2f transactions or buying/selling BTC in general, even though we're not breaking the law it doesn't mean you won't get unwanted attention from the police.

<https://bitcointalk.org/index.php?topic=174918.msg1820363#msg1820363>

This story above, I do not know if it is true either, but it is something to think about. According to the OP, law enforcement wanted all his messages and phone numbers, obviously to try and find other people involved in money laundering and the drug trade. He was scared enough to have deleted the original post, but as I mentioned, some other people quoted it and I was able to grab it.

To summarize, the police are likely watching these Bitcoin transactions to some degree and you need to establish a buyer or seller that you can trust. Once you find a good one, stick with them, even if their rates go up. Try to search for people with established feedback, ask for ID if you want, and make sure you have nothing incriminating on you, or at your home around the time of these transactions. You never know when you could be trying to offload your BTC to a cop!

LOCALBITCOINS PART 2 - THIEVES, SCAMMERS AND COUNTERFEIT

BILLS!

This post is a continuation from the last one. The threat of being ripped off or scammed on LocalBitcoin is a very real threat. One that you need to be aware of.

I want to share a few stories with you.

Quote

Going to keep it short and simple. I live in a major metropolitan city, and do a lot of business of craigslist. Meet in person, public location, inspect the item, hand cash and be on my way. I'm sure I have 25+ transactions, never been scammed.

Today, I saw someone include just as a footnote "I also accept bitcoins". Not "I only accept bitcoins" or "plz send bitcoins i mail" just a little footnote that they are fine with it.

Contacted, mentioned purchasing in cash, that was fine, and at the end decided to do it in bitcoins. Brought my laptop, public wifi, took a seat at a McDonalds. Inspected the headphones - Perfect condition, as described, everything was looking good.

He hands me a paper cutout with a wallet address, I key it into blockchain, he is looking at the address on screen. I confirm the price (80 USD, was .8xxbtc), he says good, I hit send, the little blockchain beep plays over the speakers.

He casually stands up, has the headphones, and walks away. I stand up pretty quick, and shout after to him, accusing him of theft. He says a quick comment around the lines of "If you can't pay the price don't waste my time, I said \$80" and walks out.

I contemplate chasing after him, calling the police, or fuck maybe getting some public attention, then I realized I didn't have a leg to stand on.

Cameras would show a guy sitting down at a table, showing me headphones, me inspecting them, then playing on a computer for a bit, with him walking off. I attempt to accuse him of theft, he probably didn't even have \$80 in his wallet, nothing would show me handing him cash, and the worst part, as I sat there with a mixture of adrenaline, rage and frustration - is that It was impossible for me to get that money back.

Can you imagine trying to talk to the police about this? So yeah officer, I sent him bitcoins, a virtually currency for this craigslist transaction, and then he walks off - Sir, do you have any proof of this? Well, he gave me this address of random letters, but I swear it's his, but it isn't there anymore, it's gone to a mixing service where it gets pu-

You get the point. I have a decently hard time explaining bitcoins to my eager, willing to learn friends. I can't imagine trying to explain it to an officer who thinks I just tried to give someone WoW gold for headphones.

So, is there any safety precaution out there I didn't take, or should you just keep BTC and Craigslist as far apart as possible?

Thanks for reading the rant. Sorry for the wall of text. I guess I just kinda needed to get it out there.

http://www.reddit.com/r/Bitcoin/comments/1b89wm/i_just_got_robbed_blind_of_bitcoins_in_person_im/

Remember, the risk of something like the above happening increases with the amount of Bitcoin being traded for FIAT currency (Government paper or electronic currency). So if you are trying to unload a few Bitcoins to a seller, you may find yourself in a similar situation from time to time and it is best to prepare yourself in case this happen. Bring a friend with you, have them wait at the door in case the person tries to run away, or better yet, multiple friends. If you live in a country or state where it is legal to carry a concealed weapon, then you might want to consider doing this as well.

Quote

A dangerous new scamming trend? £15,000 too close

So it appears that unfortunately scammers have changed their tactics. I have been advised by police not to disclose the username or details of the person concerned until their investigation is complete but I am ok to disclose the story as a warning to others. **Today I went to meet a buyer who was looking for £15,000 worth of bitcoins and wanted to pay in cash but this particular user had a good buyer history so although cautious I agreed to meet him in London in a place I knew there would be CCTV and security for my own safety.** Arriving there today in a public place, all went fine initially from chatting with him but when I pulled out a quick form to comply with AML's he seemed very uncomfortable which although I didn't show it, it sent alarm bells ringing in my head as he kind of covered his ID whilst in terrible handwriting filled out the form and done a completely unreadable signature looking nothing like the name. At this point, I was very tempted to call the deal off simply because my gut instinct was really telling me to back out of this but he brought up he had to withdraw another £200 from his bank and so I asked him what bank he was with, which was Nationwide, which I am too, so I went with him to the branch with the cash and forms etc in my bag and said I would just sit in the branch since it had air conditioning and was only 5 stores away. In my head at this point, I was trying to get into the branch and see if I could overhear the name he was withdrawing from and also to see if he actually owned the card he had in his hand so I could match the details up with what was on the form.

Upon arrival at the branch, he handed his card over and the bank teller gave everything a quick glance and asked him for further ID and a security check so whilst he done that, I thought I would ask the teller next to him who was free if they could put it on their cash counting machine and showed all the relevant documentation. The cash went behind the counter when she agreed and put it straight on the machine without even looking at the documents surprisingly. Immediately as this happened, the male buying the bitcoins said to me "What are you doing?" looking terrified and visibly sweating and shaking and I was absolutely certain something was very wrong at this point and before I could turn to the cashier and ask her to keep hold of the documents & cash and call security and the police for me (I was planning to write it on the piece of paper in my hand to be subtle), I heard a loud beeping sound from behind the desk which was the cash machine, rejecting every note in the pile because they were

counterfeit notes, £15,000 worth of them. As you can imagine, we had 3 security guards onto us in seconds and police arrived only 2 minutes later and as myself and the other male sat there in handcuffs, the police began to ask questions to me and the other male was taken into another room inside the branch.

Fortunately this day I had my CSV dumps of recent transactions, a letter from my HMRC communications recently as per my other post and also a bank statement to verify the recent transactions, plus copies of the emails I had exchanged with the male concerned as I bring them to every meeting in my bag for reference purposes if anything arises. Soon enough, having went back to the original place we met and reviewing CCTV footage of the whole thing, I was released but they kept everything in my bag, all the money of course and frozen my accounts whilst they investigate which I complied with voluntarily. The male who passed the counterfeit notes has been taken to the police station and will be in court tomorrow and I was advised by the Inspector he will probably be referred to the crown court on the matter and is being held in custody until his trial.

The bank and police were both present for this and the bank strongly recommended I be careful in future and transactions that large can be run through the bank if need be and they can be the third party to sign it and check everything out for £35, which will completely cover me for the AML's over £10,000 and the buyer doesn't need to go on the bank records but the bank will verify the ID is real for me in some branches too. Whilst I was there I also was given a 10 pack of pens to check notes with for future deals and police have asked me to cease trading until this case is resolved and be prepared to be asked to come to court to present testimony if required.

Again I can't name and shame the individual due to a police request, but for what it is worth, that is the story and lesson I have learned from today and despite the many big deals I have done in the past and the many shady characters, this one has really rattled me up.

<https://localbitcoins.com/forums/#!/regional/uk#a-dangerous-new-scamming-tr>

So, another recommendation if you are dealing with cash often is to get yourself some currency detection pens and a black light to check the bills for hidden logos. A quick search online will give you an idea of what to look for in the currency your country uses. Here is one more story about counterfeit money.

Quote

I occasionally trade bitcoins via localbitcoins.com, to ensure that I have a good feel for the liquidity of the market and the ability to exit at will. I've never had any problems before.

Last week, I responded to a request to buy \$500 worth of bitcoin, via a local buyer here in San Francisco.

Nothing unusual about the meet, or the buyer, other than the fact that he wanted to find a contact for regular and higher amount buys. I think he was trying to get me to increase the amount.

Anyhow, I had funded \$500 in bitcoins, in escrow with localbitcoins.com and we sat down to do the

trade. He gave me 25 x \$20 bills, which I counted. The bills felt a bit stiff, like brand new bills from an ATM. I looked at them carefully (or so I thought) and they seemed real. I pocketed the money and moved on.

Fast forward three days later, I go out with a friend. Just before leaving the house, I grab a few \$20s and put them in my wallet. At the first bar I paid for a drink, the bartender came running out 5 min later into the bar area to find me. He showed me the bill I had given him, said "this is fake, it fell apart when it got wet". True enough, the bill had not held up to water like a normal bill. I showed him the other money I had on me and he confirmed it was all fake, except for one \$20 I had from before. So I paid for my drink with the real money and left.

For those wondering, the bills are indistinguishable from real \$20s unless you know exactly what to look for. The smell and texture are slightly off. The most important clue is that the iridescent "20" on the side that changes from red-green to black-green depending on the angle you look at it. On the fake bills it does not change color.

For my next bitcoin sale, I will be carrying a UV light and pen and will be more careful in scrutiny of the bills. As always, I will only meet in public and I am never unarmed, but now I also have counterfeit detection gear.

Seller Beware - Counterfeit money being passed to bitcoin sellers in San Francisco

Edit: I will be writing an article about this for letstalkbitcoin.com and will provide links to detection tips and products to help with detection. Will also provide a more detailed story and pictures of the notes. Standby a few days for that...

Edit 2: I will be reviewing the following products against these counterfeit notes, in an upcoming article for letstalkbitcoin.com:

- Dri-mark and sharpie brand pens
- UV light + magnifying combos
- Magnetic testers

http://www.reddit.com/r/Bitcoin/comments/1nj88k/i_was_given_counterfeit_20_bills_in_exchange_for/

If this is not enough to make you feel a bit uncomfortable, then you need to read them again. But what you can do is simply learn how to inspect bills for authenticity. Again, get yourself a handheld black light, a currency marker and anything else that applies to your country's currency and you can likely protect yourself against this. If the person buying the Bitcoins off of you seems nervous, or like they are in a hurry to get away, then take greater caution with this buyer. Always try to find buyers with good feedback (although this is not perfect), possibly ask for ID if you would feel more comfortable, and bring a friend with you, but do not make it obvious that you brought a friend with

you. Getting scammed, robbed or ripped off sucks, and you need to do whatever you can to avoid it happening to you.

LOCALBITCOINS PART 3 - MORE SCAM STORIES

This post is more stories about people being scammed or robbed on LocalBitcoins.

Quote

AmbysWorld:

just got robbed in Oklahoma City - Edmond, a kid about 20 years old, brownish-blond hair, 6 ft tall, 150-160lbs

\$950.00 -

be careful doing bitcoin trades. I know it's tough to get trust, so my advice is start small and after you have gained trust, make sure the money is in your pocket before you release the coins!!

I guess it is just the price to pay to learn a lesson.

realestone:

can you give more details what happened exactly?

AmbysWorld:

We met, inside a coffee shop, introduced myself, asked him if he had done trades before. He said he had done several. I wanted to make sure he was familiar with how the site worked and then to see if he had any questions about bitcoins in general. I released the coins, and we started to shake hands as he was handing me the envelope. He jerked the envelope out of my hand and took off running.

Every person I have met has been awesome and excited about bitcoins. So I let my guard down. Showed up wearing flip flops. I started to pursue after he had already taken 3 steps, but then realized I would not be able to run in flippers for very long and stopped after about 100 yards.

The worst part is that I had my 14 year old daughter with me. There is a special place in hell for people like this!

<https://bitcointalk.org/index.php?topic=288053.0>

Here is a story from a group of people trying to test out LocalBitcoins for the first time and ended up

losing their Bitcoins due to their own ignorance. But the buyer could have done the right thing, and did not.

Quote

Bitcoin in hand, we decided to take a look at Localbitcoin and see how easy the system is for someone who does not know the lingo and does not have much experience with computers to see, what the difficulties could be.

So we found a buyer and proceeded to do an exchange of a small amount of Bitcoins. Everything looked great at first as we signed up, got verified and then proceeded to transact with the trader. We sent our Bitcoins and were confronted with some windows which began to confuse our tester, who mistakenly confirmed the transaction, minutes after sending the Bitcoins. Our tester was not sure if they needed to click the confirmation to advise the trader that the coins were sent, so spent some time in the FAQ to find out what to do next. No information was found by our tester, who then guessed that since there was no mention of it, then it must be a trivial issue and confirmed the transaction anyway. What happened next worried our tester as the transaction was marked as closed and they had sent the Bitcoins to the trader without knowing if the fiat money would be deposited into the bank account. We waited 24 hours to confirm a cash transaction into a designated account and lo and behold, it's not there.

Next we proceeded to contact the trader and as of writing, we have not heard from them. We contacted Localbitcoin support and began a ticket. Shortly afterwards, we received an email from Localbitcoin support staff and explained the situation and were told that the confirmation did need to be done AFTER we confirmed the funds had been placed into our designated account by the trader.

After a few emails to the support staff, we did explain that we were testing the system usability for the everyday mom and pop situation, because if Bitcoin is to be used properly, it needs to have an easy (dumbed down) system so the inexperienced user can make a trade without making mistakes like our tester did.

<http://mentaso.com/bitcoin-news/item/224-localbitcoins-scammed-on-our-first-test-of-the-system.html>

This next one is an attempt at a phishing scam. A phishing scam is when somebody sends you to a URL that looks like the real URL, but it is actually set up so that when you login, it steals your login credentials and the attacker takes over your account. In this case, take over the LocalBitcoin account and steal the Bitcoin

Quote

User requested nearly \$2k CAD worth of bitcoins using my localbitcoins ad.

Immediately asked to move the conversation to text messaging, asked me "how many coins I have in there (localbitcoins wallet)" then (after some dawdling and chitchat) asked me to "check out and read his other localbitcoins ad first".

Included was a URL to localbitcoinis.com with an ad url long enough I know this was copy/pasted and not a typo.

A quick WHOIS reveals a domain by proxy, but some google-fu on the contact telephone number registered to the domain reveals that domains related to this phone number have been involved in other scams.

<http://bitcoinviews.com/scam-alert-localbitcoinis-com-scammer-contacting-localbitcoins-com-users/>

Luckily for the seller, he did not fall for the scam. But anybody not careful enough could fall victim to this scam. Always make sure you read the url closely.

LOCALBITCOINS PART 4 - SELLERS BUSTED FOR MONEY LAUNDERING

Here is a simple copy and paste story you should be aware of.

Quote

State authorities in Florida on Thursday announced criminal charges targeting three men who allegedly ran illegal businesses moving large amounts of cash in and out of the Bitcoin virtual currency. Experts say this is likely the first case in which Bitcoin vendors have been prosecuted under state anti-money laundering laws, and that prosecutions like these could shut down one of the last remaining avenues for purchasing Bitcoins anonymously.

Working in conjunction with the Miami Beach Police Department and the Miami-Dade State Attorney's office, undercover officers and agents from the U.S. Secret Service's Miami Electronic Crimes Task Force contacted several individuals who were facilitating high-dollar transactions via localbitcoins.com, a site that helps match buyers and sellers of the virtual currency so that transactions can be completed face-to-face.

One of those contacted was a localbitcoins.com user nicknamed "Michelhack." According to this user's profile, Michelhack has at least 100 confirmed trades in the past six months involving more than 150 Bitcoins (more than \$110,000 in today's value), and a 99 percent positive "feedback" score on the marketplace. The undercover agent and Michelhack allegedly arranged a face-to-face meeting and exchanged a single Bitcoin for \$1,000, a price that investigators say included an almost 17 percent conversion fee.

According to court documents, the agent told Michelhack that he wanted to use the Bitcoins to purchase stolen credit cards online. After that trust-building transaction, Michelhack allegedly agreed to handle a much larger deal: Converting \$30,000 in cash into Bitcoins.

Investigators had little trouble tying that Michelhack identity to 30-year-old Michell Abner Espinoza of Miami Beach. Espinoza was arrested yesterday when he met with undercover investigators to finalize the transaction. Espinoza is charged with felony violations of Florida's law against unlicensed money transmitters – which prohibits "currency or payment instruments exceeding \$300 but less than \$20,000 in any 12-month period" — and Florida's anti-money laundering statutes, which prohibit the

trade or business in currency of more than \$10,000.

Police also conducted a search warrant on his residence with an order to seize computer systems and digital media. Also arrested Thursday and charged with violating both Florida laws is Pascal Reid, 29, a Canadian citizen who was living in Miramar, Fla. Allegedly operating as proY33 on localbitcoins.com, Reid was arrested while meeting with an undercover agent to finalize a deal to sell \$30,000 worth of Bitcoins.

Documents obtained from the Florida state court system show that investigators believe Reid had 403 Bitcoins in his on-phone Bitcoin wallet alone — which at the time was the equivalent of approximately USD \$316,000. Those same documents show that the undercover agent told Reid he wanted to use the Bitcoins to buy credit cards stolen in the Target breach.

Nicholas Weaver, a researcher at the International Computer Science Institute (ICSI) and at the University of California, Berkeley and keen follower of Bitcoin-related news, said he is unaware of another case in which state law has been used against a Bitcoin vendor. According to Weaver, the Florida case is significant because localbitcoins.com is among the last remaining places that Americans can use to purchase Bitcoins anonymously.

“The biggest problem that Bitcoin faces is actually self-imposed, because it’s always hard to buy Bitcoins,” Weaver said. “The reason is that Bitcoin transactions are irreversible, and therefore any purchase of Bitcoins must be made with something irreversible — namely cash. And that means you either have to wait several days for the wire transfer or bank transfer to go through, or if you want to buy them quickly you pay with cash through a site like localbitcoins.com.”

One very popular method of quickly purchasing Bitcoins — BitInstant — was shuttered last year. Last month, BitInstant CEO Charlie Shrem was arrested for money laundering, following allegations that he helped a man in Florida convert more than a million dollars in Bitcoins for use on the online drug bazaar Silk Road.

It’s still unclear how the defendants Espinoza and Reid were able to obtain so many Bitcoins for sale, although a review of Michelhack’s profile suggests little more than arbitrage — that is, buying Bitcoins for \$700 apiece and selling them for a couple hundred dollars more.

There is nothing that links either defendant to the Silk Road trade. But it’s notable that a third individual charged with money laundering as part of this investigation — 28-year-old Canadian citizen Vincente Loyola — is currently serving a 12-month sentence at a U.S. federal detention center for narcotics trafficking.

In any case, Weaver said he anticipates that more states will soon seek to crack down on high-dollar Bitcoin sellers on localbitcoins.com. “I’d expect many more state cases like this one because it will act to strangle the lifeblood of the online dark markets,” such as Silk Road, Weaver said. “If you want a significant amount of anonymous Bitcoins, right now this community is about the only mechanism still available.”

News of the Florida actions comes on the heels of the arraignment of Ross Ulbricht — the alleged onetime owner of the Silk Road. Ulbricht was scheduled to be arraigned in New York today.

The court documents in this case also offer a great example of the traceability of Bitcoin transactions — a potential danger for both those seeking anonymous payments and for law enforcement officials posing as criminals as part of an undercover investigation. The ICSI's Weaver noted that, by examining the times and transactions in the criminal complaint, it appears that this is the Bitcoin wallet associated with the undercover officer.

<https://krebsonsecurity.com/2014/02/florida-targets-high-dollar-bitcoin-exchangers/>

As you can see, the cops are watching LocalBitcoins. Laundering Bitcoins is like laundering real money. You need to have a way to justify where the money came from. Back in the day, the Mafia had small legitimate businesses it would run that it could claim as an income, and they might fix the numbers a bit and say they made more money than they really did. This would provide an income they could use as a reason for having money. If you are somebody who does not work, and only sell drugs on Silk Road, and are trying to cash out your coins, then I hope you have a legitimate reason for holding that many Bitcoins, otherwise you could end up like these two guys.

HIDING TOR FROM YOUR ISP - PART 1 - BRIDGES AND PLUGGABLE TRANSPORTS

This post is going to talk about something that has been commonly discussed on the forums recently. How can I hide my tor usage from my ISP ?

People are more worried about hiding their tor usage from their ISP, than hiding it from a VPN. There seems to be a back and forth debate about whether using a VPN will or will not protect you. Whether or not the VPN can be convinced to log your connection, and so forth. A few of my previous posts regarding LulzSec and the YardBird pedophile rings have shown that those who rely on VPNs to protect them are historically known to end up in jail. Even our friend we were recently introduced to, The Grugq says, TOR -> VPN is ok, but VPN -> TOR, go to jail.

In my previous posts about VPN -> TOR and TOR -> VPN, I tried to remain neutral in that you should be able to make your own decisions about how you wish to protect yourself. But just remember, at the end of the day, nobody is going to go to jail for you. If you simply want to hide the fact that you are using tor from your ISP, then we have other options than a VPN. We have bridges, and several different pluggable transports. What are these, and how can we use them in Tails?

Quote

What bridges are and when to use them

When using Tor with Tails in its default configuration, anyone who can observe the traffic of your Internet connection (for example your Internet Service Provider and perhaps your government and law enforcement agencies) can know that you are using Tor.

This may be an issue if you are in a country where the following applies:

1. Using Tor is blocked by censorship: since all connections to the Internet are forced to go through Tor, this would render Tails useless for everything except for working offline on documents, etc.
2. Using Tor is dangerous or considered suspicious: in this case starting Tails in its default configuration might get you into serious trouble.

Tor bridges, also called Tor bridge relays, are alternative entry points to the Tor network that are not all listed publicly. Using a bridge makes it harder, but not impossible, for your Internet Service Provider to know that you are using Tor.

https://tails.boum.org/doc/first_steps/startup_options/bridge_mode/index.en.html

The first thing we are going to do is get some bridges. Let us do this before we configure Tails to use bridges, because once Tails is in bridge mode, we will not be able to connect to tor without working bridges. So the first thing we want to do is visit the following webpage.

<https://bridges.torproject.org/bridges>

Enter the impossibly difficult captcha, and click "I am human", and you should get a list of bridges that look like this. These are actual bridges pulled from the tor bridges page.

Quote

```
5.20.130.121:9001 63dd98cd106a95f707efe538e98e7a6f92d28f94
106.186.19.58:443 649027f9ea9a8e115787425430460386e14e0ffa
69.125.172.116:443 43c3a8e5594d8e62799e96dc137d695ae4bd24b2
```

These bridges are publicly available on the Tor Project website, so they may or not may be the best choice to use, but they are a good start. Another option is to send an email to bridges@bridges.torproject.org with a message in the body saying "get bridges" without the quotes. This will only work if sent from a Gmail account or Yahoo, unfortunately. If you want to use this, set up the email account using tor and you will receive a list of around 3 bridges shortly thereafter. Save them somewhere you can use them the next time you boot up Tails, or write them down.

Ok, so now we have our bridges. How do we use bridges in Tails? This is an option we need to activate when we boot up Tails. To activate the bridge mode, we will be adding the bridge boot option to the boot menu. The boot menu is the first screen to appear when Tails starts. It is the black screen that says Boot Tails and gives you two options. 1. Live, 2. Live (Fail Safe). When you are on this screen, press Tab and a list of boot options will appear in the form of text at the bottom of the screen. To add a new boot option, add a Space then type "bridge" without the quotes and press enter. You have now activated bridge mode.

Once Tails boots up completely, you will get a warning that you have entered bridge mode and not to delete the default IP address in there, which is 127.0.0.1:*. This is advice we will follow, so just click OK and the settings window for tor will pop up. At this point you need to add your bridges. So you are

going to take the three bridges you got, and enter the IP address and the port. If we were going to use the example above this is what we would enter.

Quote

```
5.20.130.121:9001  
106.186.19.58:443  
69.125.172.116:443
```

For each bridge you add, type it in the available text box where it says "Add A Bridge" and then click the green + button to add that bridge. You will need to add one bridge at a time. Once you are finished adding your bridges, you can click OK. At this point, your yellow tor onion icon in the top right should turn green shortly after and you will be connected to the tor network using a bridge. Again, since these bridges are less likely to be known by your ISP, they are less likely to know that you are using tor when you use bridges.

You may wish to look up your bridge before you use it however. Maybe you want to find out where your bridge is located, maybe you want to see who is hosting the bridge, and you can do this by looking for a IP look up service online, by doing a search and typing in the IP address. The three listed above are located in the following locations.

Quote

```
5.20.130.121 - Country: Lithuania  
106.186.19.58:443 - Country: Japan  
69.125.172.116:443 - Country: New Jersey, United States
```

And with that, you can decide which bridge would be a better choice for you to use. I suggest however, that you go and get new bridges and do not use the ones I listed above for obvious reasons that they are now linked to Silk Road users by me posting them on this forum. I should note that the way bridges hide the fact that you are using tor from your ISP, is that you are connected to an IP address that is likely not known to your ISP to be affiliated with tor entry nodes.

While bridges are a good idea, unfortunately they may not be enough. According to Jacob Applebaum, (a tor developer) bridge traffic is still vulnerable to something called DPI (deep packet inspection) to identify internet traffic flows by protocol, in other words they can tell you are using tor by analyzing the traffic. While tor uses bridge relays to get around a censor that blocks by IP address, the censor can use DPI to recognize and filter tor traffic flows even when they connect to unexpected IP addresses. This is less likely to be done by your ISP, and more likely to be done by the NSA, or other oppressive governments like in China and Iran, so you can choose if this is an issue for you.

Quote

Lately, censors have found ways to block Tor even when clients are using bridges. They usually do this by installing boxes in ISPs that peek at network traffic and detect Tor; when Tor is detected they block the traffic flow.

To circumvent such sophisticated censorship Tor introduced obfuscated bridges. These bridges use special plugins called pluggable transports which obfuscate the traffic flow of Tor, making its detection

harder.

<https://www.torproject.org/docs/bridges#PluggableTransports>

Pluggable transports are a more new, but less talked about technology being implemented by tor to disguise the fact that you are using tor to your ISP and other censors. As mentioned above, it attempts to transform your tor traffic into innocent looking traffic that would hopefully be indistinguishable from normal web browsing traffic. Currently the most popular pluggable transports are obfuscated bridges. Obfuscation by definition, is the hiding of the intended meaning in communication, making communication confusing, willfully ambiguous, and harder to interpret. Obfuscated bridges actually transform the traffic to look like random packets of data. Obfuscated bridges currently have 2 protocols.

1. obfs2
2. obfs3

Obfs2 (The Twobfuscator) is talked about at length at the following official page.

<https://gitweb.torproject.org/pluggable-transports/obfsproxy.git/blob/HEAD:/doc/obfs2/obfs2-protocol-spec.txt>

But for the laymans out there, basically obfs2 uses a protocol that disguises your traffic to look like random data, whereas tor has a more distinct structure to it. However, it should be noted in the case of obfs2, that if an attacker sniffs the initial handshake between your computer and the obfuscated bridge, they could get the encryption key used to disguise your traffic and use it to decrypt the disguised traffic which would reveal it as tor traffic. They would not be able to decrypt your tor traffic, but they would be able to see you are using tor. This is not likely something your ISP would do, but it may be something law enforcement or the NSA would do. So if you are only worried about your ISP, then obfs2 would likely suffice.

Obfs3 (The Threebfuscator) is talked about at length at the following official page.

<https://gitweb.torproject.org/pluggable-transports/obfsproxy.git/blob/HEAD:/doc/obfs3/obfs3-protocol-spec.txt>

Obfs3 uses a very similar protocol to disguise your traffic as obfs2, however it uses a more advanced method of an initial handshake called the Diffie Hellman key exchange. They however found some vulnerabilities in the protocol and had to go a step further and customize the Diffie Hellman key exchange to make it an even more robust method of establishing that initial handshake. Using obfs3 would be a better bet to disguise your traffic if your adversary is the NSA or other law enforcement.

So how do you get these obfuscated bridges? They are not as easy to get, but they can be obtained from tor through email. However, you need to request those bridges specifically to get them. You need to use a Gmail or Yahoo account and send an email to bridges@bridges.torproject.org and enter in the body of the email "transport obfs2" without the quotes, and for obfs3, simply enter "transport obfs3".

Please note that you can only send one request to tor per email, every 3 hours. Which one you should use, is entirely your choice, I am just giving you the information necessary to make an informed choice. Enter them in this format so that Tails knows which protocol to use.

```
obfs3 83.212.101.2:42782
obfs2 70.182.182.109:54542
```

tor also provides a few obfuscated bridges on their home page which you can use as well, and I will list them below. If you send a request to tor and get a response containing bridges without obfs2 or obfs3 at the beginning of the lines, then these are normal bridges, not obfuscated, and they are likely to be out of obfuscated bridges at the moment. You will have to try again another day. So if you get a response with bridges that are without obfs2 or 3 at the beginning of each line, please again, be aware these are normal bridges, unlike the ones below.

```
obfs3 83.212.101.2:42782
obfs3 83.212.101.2:443
obfs3 169.229.59.74:31493
obfs3 169.229.59.75:46328
obfs3 209.141.36.236:45496
obfs3 208.79.90.242:35658
obfs3 109.105.109.163:38980
obfs3 109.105.109.163:47779
obfs2 83.212.100.216:47870
obfs2 83.212.96.182:46602
obfs2 70.182.182.109:54542
obfs2 128.31.0.34:1051
obfs2 83.212.101.2:45235
```

I have a feeling that some of you reading this will be inclined to go out and get yourself some obfs3 bridges right away, because you think they are the best choice out there for staying anonymous. And right now they have the potential of being what you hope for in that regard, except for one huge flaw. The number of obfs3 bridges is small. Last report I read put it at around 40 bridges running obfs3, and obfs2 was around 200. So while obfs3 is the most secure option out there, its limited number of available bridges would pool you into a smaller group of people making connections to the 40 available bridges and may not provide any more anonymity for you. tor is in desperate need of more obfs2 and obfs3 bridges at this time and these factors should be taken into account when using obfuscated bridges.

One of the solutions to this shortage problem, is to run your own obfuscated bridge. I am not going to go into it, but if you are interested in doing this, you should visit the following page to set up an obfuscated proxy, or better yet, purchase a few VPS and set them up as obfs2 or obfs3 proxies. One of the best things about doing it this way, is that you can configure it (with the instructions provided) to be a private obfuscated bridge, and therefore tor will not give it out to the public. You can then connect to your own private obfs3 bridge. You can also use a friend's computer, or use a server that you know is secure. But again, make sure that you trust the computer you are using, otherwise it is no

more secure than a VPN.

Another possible solution to the lack of obfuscated bridges may be another pluggable transport option, something called a flash proxy. This is brand new and not perfectly implemented yet, and please be aware that this is basically still in beta. When thinking about a flash proxy, think about the characteristics of a flash, quick and short lived. This protocol was developed by a tor developer who attended Stanford University, and the idea is that the IP addresses used are changed faster than a censoring agency can detect, track, and block them. This method is similar to using normal bridges, in that, it hides the fact you are connecting to IP addresses known to be related to tor, including when the bridge's IP addresses listed by tor are discovered by your ISP or law enforcement. This does not however, hide the fact you are using tor if somebody is analyzing your traffic using DPI (deep packet inspection).

The main benefit to this option is that the proxies are run by many people all over the world. They are run when random internet users visit a webpage with a specific plugin that turns their browser into a proxy as long as they are on that page. You are basically using somebody else's connection through their browser to connect to a tor relay. You are only using 1 active connection at any time, but you have around 5 established connections to different proxies in case your active connection drops off, then you can start using another proxy in its place. Below is another explanation of how this process works.

Quote

In addition to the Tor client and relay, we provide three new pieces. The Tor client contacts the facilitator to advertise that it needs a connection (proxy). The facilitator is responsible for keeping track of clients and proxies, and assigning one to another. The flash proxy polls the facilitator for client registrations, then begins a connection to the client when it gets one. The transport plugins on the client and relay broker the connection between WebSockets and plain TCP. (Diagram below)

<https://crypto.stanford.edu/flashproxy/arch.png>

A sample session may go like this:

1. The client starts Tor and the client transport plugin program (flashproxy-client), and sends a registration to the facilitator using a secure rendezvous. The client transport plugin begins listening for a remote connection.
2. A flash proxy comes online and polls the facilitator.
3. The facilitator returns a client registration, informing the flash proxy where to connect.
4. The proxy makes an outgoing connection to the client, which is received by the client's transport plugin.
5. The proxy makes an outgoing connection to the transport plugin on the Tor relay. The proxy begins sending and receiving data between the client and relay.

In other words, you end up going from your computer, to the proxy, then the proxy to the tor relay. - JR

The whole reason this is necessary is because the client cannot communicate directly with the relay. (Perhaps the censor has enumerated all the relays and blocked them by IP address.) In the above diagram, there are two arrows that cross the censor boundary; here is why we think they are justified.

The initial connection from the client to the facilitator (the client registration) is a very low-bandwidth, write-only communication that ideally may happen only once during a session. A careful, slow, specialized rendezvous protocol can provide this initial communication. The connection from the flash proxy to the client is from an IP address the censor has never seen before. If it is blocked within a few minutes, that's fine; it wasn't expected to run forever anyway, and there are other proxies lined up and waiting to provide service.

I know this might be a bit complicated, but you really do not need to understand how it works to benefit from it. You also might be asking about somebody just blocking your ability to connect with the facilitator (the supplier of the proxies). But, the way you actually connect to the facilitator is in a very special way that tor has designed, and this is built into the flash proxy pluggable transport. This explanation is just for your comfort, not to help you make it work.

Quote

The way the client registers with the facilitator, is a special rendezvous step that does not communicate directly with the facilitator, designed to be covert and very hard to block. The way this works in practice is that the flash proxy client transport plugin makes a TLS (HTTPS) connection to Gmail, and sends an encrypted email from an anonymous address (nobody@localhost) to a special facilitator registration address. The facilitator checks this mailbox periodically, decrypts the messages, and inserts the registrations they contain. The result is that anyone who can send email to a Gmail address can do rendezvous, even if the facilitator is blocked.

<https://trac.torproject.org/projects/tor/wiki/FlashProxyFAQ>

Two questions you should be asking. 1) Can I trust the proxies, and/or facilitator? 2) How do I use this?

Well, the facilitator is chosen and currently only run by tor, so you can take that at face value. As far as the proxies go, the proxies themselves may or may not be trustworthy, and this is the risk you run every time you use tor. Your bridges that you use may be compromised, your entry nodes, your exit nodes, every single possible hop along your way to the internet can be compromised at any given time. Luckily, even if the proxy is compromised and logging your traffic, they are only going to be able to see encrypted tor traffic. And as I mentioned above, anybody who visits a webpage with a specific plugin on it, becomes a flash proxy as long as they are on that site. This means, some people will be a flash proxy without their knowledge, and others will be flash proxies because they want to be one. The idea behind this is to have multiple users, tens of thousands, if not hundreds of thousands of flash proxies available at all times to increase the number of possible IP addresses you rotate between to keep your ISP and possibly the NSA guessing.

So do you use this? **It actually currently is not supported in Tails.** But it can be used with Tor Pluggable Transports Tor Browser Bundle outside of Tails. You can get it at the following page and it will run on your normal operating system, whether it is Windows, MAC, or Linux. Get the package at the following page.

<https://www.torproject.org/docs/pluggable-transport.html.en#download>

Next follow the following tutorial, which is pretty straight forward and has pictures of exactly what you need to do, and will probably do a better job than I would at explaining how to set it up.

<https://trac.torproject.org/projects/tor/wiki/FlashProxyHowto>

Essentially it comes down to, enable port forwarding for port 9000, add "bridge flashproxy 0.0.1.0:1" without the quotes, to your torrc, and leave everything else alone unless you need to use a different port, which is unlikely. You may need to make an exception in your firewall for the flashproxy plugin if it asks you. As long as you are using the Tor Pluggable Transports Tor Browser Bundle, it should be pretty easy to get this feature working. But until Tails adds support for it, this is the only option you have if you want to use flash proxy bridges.

Ok, so you have a lot of information right now and maybe are left a bit confused, but read over this one a few times and try to extract as much out of it as possible at once. Try setting up normal bridges, then try doing the obfuscated bridges, and once you get those working, then maybe consider doing the flash proxies if you are okay without using Tails. Tails will likely implement support for this later. Ask yourself some questions, do I just want to hide the fact that I am using tor from my ISP? Or am I hiding from somebody much bigger than that?

Consider whether it is plausible for you to run a private obfuscated proxy, or even a private bridge. Hopefully now you have enough information to make an informed decision.

Currently there are other pluggable transports currently under developed, but not yet deployed. Here is a list of upcoming projects.

Quote

ScrambleSuit is a pluggable transport that protects against follow-up probing attacks and is also capable of changing its network fingerprint (packet length distribution, inter-arrival times, etc.). It's part of the Obfsproxy framework. See its official page. Maintained by Philipp Winter.

<http://www.cs.kau.se/philwint/scramblesuit/>

Status: Undeployed

StegoTorus is an Obfsproxy fork that extends it to a) split Tor streams across multiple connections to avoid packet size signatures, and b) embed the traffic flows in traces that look like html, javascript, or pdf. See its git repository. Maintained by Zack Weinberg.

<https://gitweb.torproject.org/stegotorus.git>

Status: Undeployed

SkypeMorph transforms Tor traffic flows so they look like Skype Video. See its source code and design paper. Maintained by Ian Goldberg.

<http://crysp.uwaterloo.ca/software/SkypeMorph-0.5.1.tar.gz>

<http://cacr.uwaterloo.ca/techreports/2012/cacr2012-08.pdf>

Status: Undeployed

Dust aims to provide a packet-based (rather than connection-based) DPI-resistant protocol. See its git

repository. Maintained by Brandon Wiley.
<https://github.com/blanu/Dust>
Status: Undeployed

Format-Transforming Encryption (FTE) transforms Tor traffic to arbitrary formats using their language descriptions. See the research paper and web page.

<https://eprint.iacr.org/2012/494>
<https://kpdyer.com/fte/>
Status: Undeployed

Also see the unofficial pluggable transports wiki page for more pluggable transport information.
<https://trac.torproject.org/projects/tor/wiki/doc/PluggableTransports>

Source: <https://www.torproject.org/docs/pluggable-transport.html.en>

CAPABILITIES OF THE NSA

I wanted to share a 1 hour video by one of the tor developers Jacob Applebaum.

He talks about legitimate, confirmed capabilities of the NSA from FOIA leaked documents showing just how technically capable the NSA is. Anywhere from simple backdoors, flying a drone over top of your house to sniff packets, mold injecting backdoor chips into your computer case, to beaming energy into your house. None of this is conspiracy theory, it is all confirmed with documents shown in his presentation.

The video can be watched on YouTube using HTML5 embedded instead of flash at the following page.
<https://www.youtube.com/embed/vILAlhwUgIU>

I also uploaded it on AnonFiles.com in case you would prefer to download it and watch it in Tails.

<https://anonfiles.com/file/eb07bbcc15ae5aeba1e1322d2995fdde>

The SHA1 checksum is 801fa9c2b3f2dfe120f93e6ffa6e6a666e5aa12a
The MD5 checksum is eb07bbcc15ae5aeba1e1322d2995fdde

For those of you using Tails, just use place this file in your tmp folder [Places -> File System -> tmp]
Open a terminal (black rectangle icon) and type the following commands.

```
cd /tmp  
md5sum 1391628603972.zip
```

sha1sum 1391628603972.zip

And check that the outputted string matches.

WHY YOU SHOULD ALWAYS BACK UP YOUR DRIVES, ESPECIALLY ENCRYPTED DRIVES

This is an embarrassing story of something that happened to me in the past few days, and it was a lesson well learned, for some of the things I have lost are not recoverable. - Jolly Roger

Do you have your Bitcoin wallets saved on a flash drive? What would happen if you lost your flash drive? Do you have a backup? What would happen if your files became corrupted and were not able to be recovered, could you live with that? Do you have certain things that would absolutely cause a huge problem if you lost them? Then you better start backing up your drives regularly, better yet, **do it daily!**

I am the type of person who usually backs up his files regularly, but unfortunately do to the large amount of strange events occuring online lately with Utopia being brought down, BMR forums being seized, Silk Road being robbed and so forth, I had not backed up my files in about 2 weeks. I had all of my most recent files, including a few new Bitcoin wallets with balances on them on my main portable drive, and on top of it, this drive was encrypted.

Then, without warning, I suddenly received an error that the file system was corrupted and my disk could not be read. No matter, if you have an unencrypted drive, you can simply run a data recovery program such as **testdisk**. Open up your terminal and type the following. Make sure you started Tails with a login at the boot up when it asks you.

sudo apt-get install testdisk

Using this program (follow documentation online) you can likely recover most of your files because it ignores file system headers and other types of file organization required to identify the way the files are stored. There are many other programs as well. The problem in my case, was that all my files were encrypted. This means, that in order to decrypt the files, I needed a key file that is stored on the drive to unlock my files. If this key file gets damaged, then even if you have the password for your files, you will not be recovering your files.

The key is unique to that particular instance when you encrypted the drive. Meaning that even if I tried to recreate the key file with the same password, the result would be a different key file. This means essentially that my data is unrecoverable, because my key file was somehow corrupted. Technology is delicate, data is stored in the form of magnetic frequencies and there is no guarantee that files will not become corrupted one day for seemingly no reason. Here are some things that could ruin your data.

Flood, hurricane, power surge, fire, moisture damage, accidentally stepping on your drive, a family member (usually a child) breaks it, you lose it, spill water on it, over heats, and so forth.

All of these could result in your data or drive getting damaged and losing all of your data. This is why you need a minimum of 2 backups. Not 1, but 2. And have one of your backups preferably stored outside of your home. If you work, store one at work, or in your car, or somewhere you can access regularly, and try to back up your data as often as possible. If your house burns down and you kept all your backups at home, then you lose everything. If you kept a copy at work, then you can recover it. The more backups the better, as long as they are encrypted. Any time you create a new wallet and transfer Bitcoin into it, back it up. Any time you set up a new account or a new email with a unique password (which should be every time), back it up. You need to be backing up everything.

Luckily for myself my main wallet was recoverable with the majority of my coins, but I did lose some coins, which can never be recovered, trust me, I tried. Getting extra USB drives or SD cards are very cheap and inexpensive, so you owe it to yourself to spend a few extra dollars to have multiple backups just in case you wind up in my situation where you had not backed up your drive in a couple of weeks and end up losing data that could cost you a lot more than what it would have costed to have a few extra drives laying around as back ups.

BITCOIN CLIENTS IN TAILS - BLOCKCHAIN AND ELECTRUM

Note: as of now, electrum is included in TAILS, no need to setup anything. This is obsolete and insecure as the download is not checked - I did copy it anyway for your information but you'd rather use the electrum client that comes with TAILS.

In this post I want to talk about 2 options for trading your Bitcoins.

#1 - Blockchain

#2 - Electrum

By now, hopefully you know how to use BlockChain. If not, you simply go to <http://blockchain.info> and press the button "Wallet" and you can open up your existing wallet or create a new account. Very straight forward and can be done all from your web browser.

But what about Electrum? Electrum is an easy to use Bitcoin client. It protects you from losing coins in a backup mistake or computer failure, because your wallet can be recovered from a secret phrase that you can write on paper or learn by heart. There is no waiting time when you start the client, because it does not download the Bitcoin blockchain. If you use the normal Bitcoin client from <https://bitcoin.org> then you would need to download the entire blockchain, which is several GB of data. In Tails, we are trying not to download too much to our computers. Downloading the entire BlockChain can take over 24 hours.

So how do we set up Electrum in Tails? First thing we need to do is download it.

<https://download.electrum.org/Electrum-1.9.7.tar.gz>

Now extract it (right click -> Extract here) and rename the folder to electrum to make things easier.

(Right click -> Rename). You might also want to move the folder to the **tmp** directory so it is easier to find. (Places -> Computer -> File System -> tmp)

Next open up a terminal and type the following command

```
cd /tmp/electrum
```

You can replace /tmp/electrum with whatever directory electrum is currently in, but this is why we put it in tmp, to make things easier for us. Next type the following command.

```
./electrum -s 56ckl5obj37gypcu.onion:50001:t -p socks5:localhost:9050
```

This will allow your electrum to connect through Tor, to make sure it does not connect over clearnet. You will get a warning when you do this that electrum is attempting to connect in an unsafe manner, but this is expected, and do not worry, it is safe to do this. This step was recommended on the Tails web page at the following URL.

https://tails.boum.org/forum/Report:_the_electrum_bitcoin_client_in_tails/

Since you are likely going to want to reuse your wallet that is generated in Electrum, you can specify where your wallet is kept by replacing the above command with the following command.

```
./electrum -s 56ckl5obj37gypcu.onion:50001:t -p socks5:localhost:9050 -w /tmp/electrum.dat
```

You would replace /tmp/electrum.dat with whatever the path to your wallet is, and you can rename **electrum.dat** to whatever you want to call your wallet, like **srwallet.dat** or whatever you want. Or leave it the way that it is. Then each time you want to start up electrum, reuse the same command, and make sure you copy electrum.dat into **/tmp** or whatever directory you wish to use. Then when you are finished, make sure to back up electrum.dat onto your USB drive or SD card, especially if you do not have Tails persistence. This way you can reuse the same wallet and you will not lose your balance.

Electrum is likely going to be the Bitcoin client of choice for Tails users. And you can read more about how to use Electrum by visiting the home page at the following link.

<https://electrum.org>

YET ANOTHER EXAMPLE OF HOW STRONG CRYPTOGRAPHY AND PROPER OPSEC CAN PROTECT EVEN PEDOPHILES

Yes, you read the title correctly. Using the same types of techniques taught in this thread, you can and should remain anonymous no matter what you are doing.

Pedophiles and child pornographers are some of the most wanted people on the planet. They are up there with terrorists and serial killers. They are hunted by federal law enforcement agencies, and punished very seriously, as they should. So the reason for this post is to demonstrate, that if somebody

who is as wanted as much as pedophiles and child pornographers can remain free by using proper OpSec, then you can too.

Quote

If your secure communications platform isn't being used by terrorists and pedophiles, you're probably doing it wrong.

<http://grugq.github.io/blog/2013/12/01/yarbirds-effective-usenet-tradecraft/>

I want to talk to you about a group of child pornographers that operated for several years online, called YardBird. During a period of 15 months, there were around 400,000 images and 11,000 videos uploaded to a central server run by the group and shared by the members. The reason we know that, is because during that 15 months, the FBI performed an undercover operation to infiltrate the group in hopes of apprehending the members. They successfully apprehended 1 in 3 members of the group. One of those who remain free to date, was the leader of the group, who also went by the online name YardBird.

How is it possible that after so much effort was put in by the American Federal Bureau of Investigation (FBI), the Australian Federal Police (AFP) and the Australian Queensland Police Service, that people high up on the wanted lists were able to evade capture. They used strong cryptography, and proper OpSec rules. Let us now talk about the history of the attempted apprehension of this group.

According to the FBI.

Quote

There were approximately 60 members that were loosely identified, and from the 60, approximately 20 were positively identified in this group.

There were numerous challenges presented during Operation Achilles. The group utilized an unprecedented level of organization and sophistication. They had a timed test for prospective new members. They had to use encryption technology and Internet-based anonymizers, re-mailing services. They also intentionally corrupted their own child pornography files and only the new members knew how to reconfigure those files to be able to read the pictures or the video. They also had the uncanny ability to monitor worldwide news pertaining to law enforcement efforts in child pornography matters in order to better educate themselves to avoid law enforcement detection.

<https://www.fbi.gov/news/podcasts/inside/operation-achilles.mp3/view>

As I said earlier, the alleged leader of this ring used the online name "Yardbird". Yardbird made a re-appearance on Usenet in both 2009 and 2010 on the date corresponding to the first and second anniversaries of the busts in 2008. His intent was to show that he was still free, and to answer people's questions.

One of the most important things Yardbird stated were that everyone in the group who used Tor and remailers remained free, while those who relied on services such as Privacy.LI were arrested and

convicted. Privacy.li is an offshore VPN service that promises anonymity. They claim from their website the following.

Quote

If you need corporate and/or military strength encrypted networks, then a Virtual Private Network is the way to go. All and any traffic from and to your desktop are within an encrypted tunnel, and your originating IP-address is well concealed.

<http://www.privacy.li/services.html>

And their privacy policy makes the following promise.

Quote

Yes, we 101% honor your privacy, no logs, no snooping, no profiling. No legal mumbo-jumbo to disguise any hidden efforts. We believe in individualism and privacy, even anonymity.

<http://www.privacy.li/privacy-policy.html>

Yardbird further commented that several members of the group, including his second-in-command Christopher Stubbings (Helen) and Gary Lakey (Eggplant) were Privacy.LI users -- in fact he stated that they used it for everything. (Helen is currently serving a 25-year sentence in the UK, while Eggplant is serving life in an Arizona prison.)

Eggplant literally became notorious because of his constant promotion of Privacy.LI -- he continually boasted that he could not be caught because Privacy.LI did not keep logs, and they were located outside of U.S. jurisdiction.

Quote

I pointed out to anyone who would listen that services such as Privacy.LI were for /privacy/ -- not for anonymity. In an ideal situation, one needs both to be private as well as anonymous. Essentially, what Privacy.LI supplied was a type of VPN service, providing an encrypted tunnel for data to travel between two endpoints--the customer's computer being one endpoint, while the Privacy.LI servers provided the other. While there was a degree of privacy, there was NO anonymity at all--so it really didn't come as a surprise that Privacy.LI's customers were among those arrested.

<http://dee.su/uploads/baal.html>

At the end of the day, no service provider is going to go to jail for you. A simple court order can get even the toughest VPN providers to roll over on their users, because they would rather betray a \$20 per month user than be fined, shut down and possibly thrown in jail for interfering with a federal investigation.

What other mistakes were made to lead to the arrest of some members of this group? The Australian police arrested a man on totally unrelated child pornography charges, and presumably as part of a plea deal, he revealed the existence of 'the group' and handed over a PGP public/private keypair and

password. Having acquired from the informer the current group PGP public/private keypair, and its passphrase meant that the police could assume this group member's identity, and furthermore, read all the encrypted traffic posted by members of the group.

Quote

Once the group was penetrated, the police were able to take advantage of a few factors:

- 1) They had the informant's computer, with all its email, PGP keys and the like. This provided a history, which made it easier to continue the impersonation.
- 2) By the time it was penetrated, the group had been operating for about 5 years. By this time, the group had jelled into a community -- people were familiar with each other, they often let their guards down, and would sometimes reveal tidbits of personal information. This is especially the case when they thought their messages were secure, and beyond the ability of the police to intercept--they would say things that they would **never** say in the open.

<http://dee.su/uploads/baal.html>

So it is important to note at this time, that you no matter how comfortable you become with somebody, there is always a chance that they can become compromised. In fact, the group has a set of rules, that all members were told to abide by, and if any member was found to be breaking the following rules, they would be expelled.

Quote

- Never reveal true identity to another member of the group
- Never communicate with another member of the group outside the usenet channel
- Group membership remains strictly within the confines of the Internet
- No member can positively identify another
- Members do not reveal personally identifying information
- Primary communications newsgroup is migrated regularly
- If a member violates a security rule, e.g. fails to encrypt a message
- Periodically to reduce chance of law enforcement discovery
- On each newsgroup migration
- Create new PGP key pair, unlinking from previous messages
- Each member creates a new nickname
- Nickname theme selected by Yardbird

<http://grugq.github.io/blog/2013/12/01/yarbirds-effective-usenet-tradecraft/>

The ones who got caught, were the ones who did not follow the rules by putting too much trust in their online "friends". We saw this in the arrest of Sabu when he helped the FBI bust his "friends" in LulzSec.

If someone is given a deal to cut the amount of time spent in prison in half, they likely will take the deal at your expense. Below is an example of a plea versus trying to fight the charges in this exact case.

Quote

...seven of the U.S. subjects pleaded guilty pre-trial to a 40-count indictment and received federal sentences ranging from 13-30 years in prison. The remaining seven defendants opted for a joint, simultaneous trial. All seven were convicted by a jury and subsequently sentenced to life in prison.

<https://www.fbi.gov/news/podcasts/inside/operation-achilles.mp3/view>

13-30 years versus life in prison, may entice even some of the hardest criminals, and if you think your online "friend" who you have never met in person is going to keep their mouth shut to keep you out of jail, you are in for a big surprise.

So, as you can see, the group was pretty much an open book to the police. They were completely and thoroughly penetrated. Despite that, however, the majority of the group were still able to remain at large, and were neither positively identified nor arrested. This is due to the privacy tools (pgp, tor, nymserver, remailers) that were employed. Even with everything else being an open book, those using these tools still managed to evade capture. But you may be saying, Ok, I understand PGP, I understand tor, but what the heck is a nymserver and a remailer?

In a nutshell, an anonymous remailer is a server that receives messages (in this case an email) with embedded instructions on where to send them next, and that forwards them without revealing where they originally came from. A nymserver also referred to as a pseudonymous remailer assigns its users a user name, and it keeps a database of instructions on how to return messages to the real user. These instructions usually involve the anonymous remailer network itself, thus protecting the true identity of the user.

Some of the advantages of using these services are to protect the intended recipient from an adversary, and also protect the sender of the message. Some of these services use what is called a common mailbox, in which all messages are stored in a central mail box with no "To and From" headers. It is up to the users who use the service to attempt to use their PGP keys to try and decrypt all of the messages stored in the central message box and see if they can decrypt any of them. If they can, this message is intended for them. This way it rules out again, the sender and receiver. This system of remailers, can also form a chain, in which the message is bounced off of multiple remailers before making it to its intended recipient to widen the gap between the sender and receiver.

Another effective option some services offer is the ability to delay when the message gets sent on to the next server in the chain, or the recipient itself. If you are found to be sending out PGP encrypted traffic through some type of analysis at 5:00PM, and another person being monitored receives it at 5:01PM, it is easier to correlate that this message may be from you to the other person being monitored. At this time I have no recommendations for service to use, but I am likely to post about them in the future. In the meantime, let us get back to the ring of pedophiles shall we?

Quote

Leaving aside my personal feelings about pedophiles, I brought up this case as an example for several reasons:

1) Child pornography is a serious crime in virtually every jurisdiction. As this example demonstrates, police will work together, even across national boundaries, to investigate these crimes. They are willing to invest considerable time, manpower and money in pursuit of these suspects. The only other crimes which usually merit this type of approach are drug/gun-running or terrorism. The level of effort expended in pursuing this group can be seen in that even FBI executive assistant director J. Stephen Tidwell was involved.

Normally one would not expect FBI personnel that highly placed to be involved -- this shows the level of importance placed on this particular investigation. (A year or so after the busts, Yardenbird himself expressed astonishment that the FBI would consider his group such a priority.)

2) This case is the only one that I'm aware of, where suspects were using sophisticated tools like PGP, Tor, anonymous remailers and nym servers.

3) This case underscores the effectiveness of these tools even against well-funded, powerful opponents like the FBI, Europol, and Interpol. (N.B.: FWIW, those who were caught used either inappropriate and/or ineffective tools and techniques to protect themselves.

4) I fully understand most people's disgust at the types of crimes/criminals being discussed here. That said, it is important to remember that one simply cannot design a system that provides protection for one class of people, but denies it for another. You can't, for example, deploy a system that provides privacy/anonymity for political dissidents, or whistle blowers, and yet denies it to pedophiles -- either **everyone** is safe, or *NO ONE* is safe. This may not be palatable, but these are the facts.

<http://dee.su/uploads/baal.html>

To summarize. We have seen that even the most hunted criminals, can evade capture when using strong cryptography and proper OpSec. The ring leader of one of the most investigated child pornography rings still remains at large today because those who followed the rules.

DENIABILITY, IDENTIFYING TAILS USERS, AND CAN YOU BE FORCED TO GIVE UP YOUR PASSWORDS?

Quote from: OCDPolak
Hi JR,

For some reason I have seen a lot of information and discussion about privacy and anonymity but nothing at all about deniability, which to be honest concerns me that some people may think that because the NSA can't crack their password, everything is safe but people easily overestimate their ability to stand up to sanctions imposed by a court should the shit hit the fan...

I was wondering about the deniability problems with using Tails (or any of the security measures really). You have to assume that if you get arrested and it goes to court, you will be compelled to give any of your passwords that they want. It's all well and good thinking that you won't give it to them, but when they sentence you to a \$1000 a day or simply jail until you tell them you will probably tell them your passwords...

With that in mind, is there any deniable way to use Tails (or at least deniable in some respects)? I used to run everything off a hidden volume in a Truecrypt memorstick (which is supposed to be impossible to prove exists), is there an equivalent with LUKS?

Also, can your ISP or FBI differentiate between Tor and Tails through your internet usage?

Thanks for your time

Here are some things to consider.

Quote

Tails makes it clear that you are using Tor and probably Tails

Your Internet Service Provider (ISP) or your local network administrator can see that you're connecting to a Tor relay, and not a normal web server for example. Using Tor bridges in certain conditions can help you hide the fact that you are using Tor.

The destination server that you are contacting through Tor can know whether your communication comes out from a Tor exit node by consulting the publicly available list of exit nodes that might contact it. For example using the Tor Bulk Exit List tool of the Tor Project.

So using Tails doesn't make you look like any random Internet user. The anonymity provided by Tor and Tails works by trying to make all of their users look the same so it's not possible to identify who is who amongst them.

<https://tails.boum.org/doc/about/warning/index.en.html#index2h1>

Quote

In this context, the term fingerprint refers to what is specific to Tails in the way it behaves on Internet. This can be used to determine whether a particular user is using Tails or not.

As explained on our warning page, when using Tails it is possible to know that you are using Tor. But Tails tries to make it as difficult as possible to distinguish Tails users from other Tor users, especially Tor Browser Bundle (TBB) users. If it is possible to determine whether you are a Tails user or a TBB user, this provides more information about you and in consequence reduces your anonymity.

This section explains some issues regarding the fingerprint of Tails and how this could be used to identify you as a Tails user.

For the websites that you are visiting

The websites that you are visiting can retrieve a lot of information about your browser. That information can include its name and version, window size, list of available extensions, timezone, available fonts, etc.

To make it difficult to distinguish Tails users from TBB users, the Tor browser tries to provide the same information as the TBB in order to have similar fingerprints.

See the fingerprint section of know issues page for a list of known differences between the fingerprints of the Tor browser and the TBB.

Apart from that, some of the extensions included in Tor browser are different than the ones included in the TBB. More sophisticated attacks can use those differences to distinguish Tails user from TBB users.

For example, Tails includes Adblock Plus which removes advertisements. If an attacker can determine that you are not downloading the advertisements that are included in a webpage, that could help identify you as a Tails user.

For the moment, you should consider that no special care is taken regarding the fingerprint of the Unsafe Browser.

For your ISP or local network administrator

Tor bridges are most of the time a good way of hiding the fact that you are connecting to Tor to a local observer. If this is important for you, read our documentation about bridge mode.

A Tails system is almost exclusively generating Tor activity on the network. Usually TBB users also have network activity outside of Tor, either from another web browser or other applications. So the proportion of Tor activity could be used to determine whether a user is using Tails or the TBB. If you are sharing your Internet connection with other users that are not using Tails it is probably harder for your ISP to determine whether a single user is generating only Tor traffic and so maybe using Tails.

Tails do not use the entry guards mechanism of Tor. With the entry guard mechanism, a Tor user always uses the same few relays as first hops. As Tails does not store any Tor information between separate working sessions, it does not store the entry guards information either. This behaviour could be used to distinguish Tails users from TBB users across several working sessions.

When starting, Tails synchronizes the system clock to make sure it is accurate. While doing this, if the

time is set too much in the past or in the future, Tor is shut down and started again. This behavior could be used to distinguish Tails from TBB users, especially this happens every time Tails starts.

<https://tails.boum.org/doc/about/fingerprint/index.en.html>

Read those pages directly as they have links to other articles on them as well.

Here is another little trick I know of. Never keep a password you can remember. You cannot give up a password you do not know. Perhaps you have a little piece of paper with your password on it that you swallow the second the cops come in. A long password that you could never remember.

Another thing you can say is, I wrote down my password on a piece of paper but the police must have destroyed the piece of paper when they raided my home. Check out the below quote from an article.

Quote

Dubois said that, in addition, his client may not be able to decrypt the laptop for any number of reasons. "If that's the case, then we'll report that fact to the court, and **the law is fairly clear that people cannot be punished for failure to do things they are unable to do,**" he said.

http://news.cnet.com/8301-31921_3-57364330-281/judge-americans-can-be-forced-to-decrypt-their-laptops/

And in the case of whether or not you can be forced to give up a password is a matter of debate that has gone back and forth in court cases to date.

Quote

Many in the legal arena say the issue is a tricky -- and largely unsettled one.

A small number of courts have permitted it, but only when prosecutors can point to specifically what files they need and where they are located.

In the motion filed earlier this week, Assistant County Prosecutor Matthew Meyer stated the law is not clear.

http://www.cleveland.com/court-justice/index.ssf/2014/03/bedford_judge_case_highlights.html

And what about the charges for failing to do so?

Quote

disobeying a judge's order to hand over a password could result in contempt of court charges or being jailed.

http://www.cleveland.com/court-justice/index.ssf/2014/03/bedford_judge_case_highlights.html

And in the US, since most people busted will be extradited there anyways, treats contempt in the following way.

Quote

If a person is to be punished criminally, then the contempt **must be proven beyond a reasonable doubt**, but once the charge is proven, then punishment (such as a fine or, in more serious cases, imprisonment) is imposed unconditionally.

A court cannot maintain an order of contempt where the imposed party does not have the ability to comply with the underlying order. This claim when made by the imposed party is known as the "impossibility defense".

https://en.wikipedia.org/wiki/Contempt_of_court#United_States

Furthermore.

Quote

"the government must prove the existence and location of the subpoenaed documents and possess independent evidence, other than compliance with the court order, for authenticating them" [1, p. 581]. In other words, law enforcement cannot simply go on a fishing expedition, hoping to turn up data that will be evidentiary [8]. They must be able to demonstrate the existence and likely location of specific documents.

http://www.asis.org/Bulletin/Dec-13/DecJan14_Oltmann.html

In regards to two cases in which defendants were not forced to give up their passwords

Quote

United States v. Kirschner (2010): Kirschner was indicted for child pornography charges, and the government subpoenaed his encryption key to gain further evidence from his encrypted drive. In this case, the judge determined that requiring a defendant to supply his password would violate his right against self-incrimination.

United States v. Doe (2012): Doe was charged with child pornography. He refused to supply his decryption key and was found in contempt of court, then jailed. A judge then ruled that supplying his decryption key would be tantamount to self-incrimination, so Doe did not have to supply it.

http://www.asis.org/Bulletin/Dec-13/DecJan14_Oltmann.html

The analysis of why they were not forced to give them up is below.

Quote

In contrast, law enforcement in the **Kirschner and Doe cases did not have prior evidence that illegal content was on their computers. In these cases, officers had suspicion of wrongdoing and were relying on the revelation of decryption keys to investigate and uncover evidence.** The court in Kirschner determined that sharing the key “would be testimonial because it would demonstrate knowledge of the password and access to the underlying computer files ...providing the password would reveal the contents of an arrestee’s mind by recalling the password” [5, pp. 1171-1172], [6]. Simply put, because the password was not written down (or already known to law enforcement) in Kirschner and Doe, and it existed only in their minds, compelling a defendant to reveal it would be self-incriminating testimony.

If law enforcement can describe the existence and location of evidence, they have a stronger case for requiring access; however, if they cannot demonstrate prior knowledge of the likely data, separate from a compelled revelation from a defendant, then law enforcement has a weaker position.

http://www.asis.org/Bulletin/Dec-13/DecJan14_Oltmann.html

But when law enforcement was able to provide proof of existing evidence on an encrypted drive, courts were much more likely to demand decryption, such as in the following cases.

Quote

In re Boucher (2009): Boucher entered the United States from Canada. A border agent examined Boucher’s computer and found child pornography after Boucher supplied the password. The agent then shut down the computer and arrested Boucher. Shutting down the computer triggered the encryption again, and prosecutors could no longer see or find the illegal images. Boucher was ordered by the courts to supply the password, but he invoked his Fifth Amendment privilege. The courts subsequently ruled he had to supply a decrypted copy of the drive’s contents.

Commonwealth v. Hurst (2011): Hurst was charged with offenses related to inappropriate sexual relations with a minor. Police suspected incriminating evidence was on Hurst’s cellphone, but he refused to supply the password. Before this case reached the court system, Hurst’s wife supplied the password, and Hurst himself pled guilty.

United States v. Fricosu (2012): Fricosu was indicted for mortgage and real estate fraud. She refused to surrender the password (at one point saying she forgot the password) to encrypted files that, the government believed, would incriminate her. The court ordered her to supply a decrypted version of the hard drive, rather than her password. Subsequently, a co-defendant supplied the needed passwords.

And the analysis of the cases below.

Quote

Law enforcement saw evidence of criminal wrongdoing in the Pearson, Boucher, Hurst and Fricosu cases.

Both Pearson and Boucher voluntarily agreed to let law enforcement search their computers; during those searches, the officers saw evidence. It was only after the initial search that the question of encryption became relevant. In these cases, because the defendants had “permitted investigators to see at least some” of the evidence, this “sufficed to render the existence of all the illegal files a ‘foregone conclusion’” rather than testimonial evidence [8, p. 544]. Hurst had sent inappropriate messages to a minor, which were visible on the minor’s phone. While the police sought confirmation of the transmission by searching Hurst’s phone, they had sufficient evidence without that step. In the Fricosu case, police had recorded conversations between the defendant and her husband (a co-defendant) that revealed the existence and content of the sought-after documents.

http://www.asis.org/Bulletin/Dec-13/DecJan14_Oltmann.html

1) It may be possible, to identify you as a Tails user, but it would take a lot of analysis to do so, and Tails is getting better at blending in with every update.

2) Think about what you could possibly be charged with, and think about whether or not it is more serious than a contempt charge. The longest sentence to date for contempt was 14 years, and this is almost unheard of. You are not likely to get this kind of charge against you, but if you do, would it be better than life in prison for whatever else you might be charged with?

Remember Sabu to LulzSec hacker? being charged with 112 years in prison for hacking? I think he would trade 14 years in prison for contempt over 112 years any day. I know I would.

3) Without the knowledge of incriminating evidence existing on your drives, you are less likely to be forced to decrypt your drives, and this even applies in child pornography cases as demonstrated above.

4) Maintain your right to remain silent, never keep anything on your computers that you do not have to.

5) Do not have a password you can remember. Or if you do, tell them you had it written down but it was misplaced or possibly damaged during the raid and you are unable to recall the password. Perhaps you are too traumatized from the even of having your face shoved into the floor to remember what happened during those 2 minutes?

Anyways, this is a lot of data to go through, so I will leave it at that and we can go from there. You just need to always follow best practices. Turn off your computer when you are not using it, encrypt everything, never tell anybody your passwords, never leave any evidence of the contents of your drives lying around (like notes or diary entries), and never admit having anything on your drives to anyone online, even under your pseudonym as that can be used against you in court.

Deny deny deny deny deny.

Hope this helps.

Security Culture: A Handbook for Activists

This handbook is the first edition of what we hope will be an evolving and growing document dealing with security issues and canadian activism. A lot of this information is general and can be applied to any locality - other information is easily adapted to fit other situations. For more information or to make contributions to this document – please email securitysite@tao.ca

Second edition - prepared August 2000

Introduction:

Resistance has been on the rise for the past few years, with activists adopting more and more effective tactics for fighting back. Now, more than ever, we pose some threat to the status quo. Our increased activity and effectiveness has meant that the RCMP, FBI, and local police have continued to escalate their activities against us. If we want our direct action movement to continue, it is imperative we start tightening our security and taking ourselves more seriously. Now is the time to adopt a security culture. Good security is certainly the strongest defense we have. This is a handbook for the Canadian (and even US) activist who is interested in creating and maintaining security awareness and culture in the radical movements. It is not nearly complete - but is what we have got finished at the moment. We are always looking for contributions - so please feel free to email securitysite@tao.ca with any images or text you think belong in a handbook such as this. If this material appears familiar to you - its because this is the second edition of this zine that we have put out... mostly to correct spelling errors and other small things. The three articles in this pamphlet have been mostly cobbled together from other writings

that already exist on this subject out there so we don't claim any of this to be 100% original material - though we have included quite a bit of fresh info on the Canadian state and its operation (mostly because we have found the majority of info out there to be very focused on US law enforcement). We hope that you will put the material contained within to good use!

Security What it is, why we need it and how we implement it...

A gitators; liberationists; abolitionists; union organizers; revolutionaries... From large uprisings challenging the entire political structure, to isolated environmental and social struggles, people have always worked to create a better world. For government the response has usually been to jail activists and revolutionaries through use of the courts and police forces. As direct action movements become more effective, government surveillance and harassment will increase. To minimize the destructiveness of this political repression, it is imperative that we create a security culture within our movements. This pamphlet is essential reading for anyone who is associated with groups that advocate and/or utilize sabotage, theft, arson and more militant tactics. The advice herein also applies to anyone who is associated with groups that practice civil disobedience, especially since membership often overlaps and gossip travels freely between groups.

Even if you have never picked up a monkeywrench or been arrested for civil disobedience, even if you think you have nothing to hide, these guidelines will enhance your personal safety as well as the movement's overall effectiveness. Surveillance has been set up on all sections of political movements in the past. Governments in the western industrialized world have targeted groups that have advocated sabotage and groups that have not, movements that have been militant and movements that have been markedly pacifist. The government's security machinery serves political and economic objectives, and there are over 250 political prisoners in Canada and the US that can testify to this from firsthand experience. By adopting a security culture, we can defeat various counterintelligence operations that would otherwise disrupt both mainstream organizing and underground resistance.

SO WHAT IS A SECURITY CULTURE?

It's a culture where the people know their rights and, more importantly, assert them. Those who belong to a security culture also know what behavior compromises security and they are quick to educate those people who, out of ignorance, forgetfulness, or personal weakness, partake in insecure behavior. This security consciousness becomes a culture when the group as a whole makes security violations socially unacceptable in the group.

WHAT NOT TO SAY

To begin with, there are certain things that are inappropriate to discuss. These things include:

- your involvement or someone else's involvement with an underground group
- someone else's desire to get involved with such a group
- asking others if they are a member of an underground group
- your participation or someone else's participating in any action that was illegal
- someone else's advocacy for such actions
- your plans or someone else's plans for a future action

Essentially, it is wrong to speak about a specific individual's involvement (past, present or future) with illegal activities. These are unacceptable topics of discussion regardless of whether it is rumor, speculation or personal knowledge.

Please note: this is not to say that it is wrong to speak about direct action in general terms. It is perfectly legal, secure and desirable that people speak out in support of mokeywrenching and all forms of resistance. The danger lies in linking individual activists to specific actions or groups.

THREE EXCEPTIONS

There are only three times that it is acceptable to speak specifically about actions and involvements. The first situation would be if you were planning an action with other members of your small group (your "cell" or "affinity group"). However, you should never discuss these things over the Internet (email), phone line, through the mail, or in an activist's home or car, as these places and forms of communication are frequently monitored. The only people who should hear this discussion would include those who are actively participating in the action. Anyone who is not involved does not need to know and, therefore, should not know. The second exception occurs after an activist has been arrested and brought to trial. If she is found guilty, this activist can freely speak of the actions for which she was convicted. However, she must never give information that would help the authorities determine who else participated in illegal activities. The third exception is for anonymous letters and interviews with the media. This must be done very carefully and without compromising security. Advice on secure communication techniques can be found in other publications. These are the only situations when it is appropriate to speak about your own or someone else's involvement or intent to commit illegal direct action.

SECURITY MEASURES

Veteran activists only allow a select few to know about their involvement with direct action groups. Those few consist of the cell members who they do the actions with AND NO ONE ELSE! The reason for these security precautions is quite obvious: if people don't know anything, they can't talk about it. It also means that only the people who know the secret can also face jail time if the secret gets out. When activists who do not share the same serious consequences know who did an illegal direct action, they are far more likely to talk after being harassed and intimidated by the authorities, because they are not the ones who will go to jail. Even those people who are trustworthy can often be tricked by the authorities into revealing damaging and incriminating information. It is safest for all cell members to keep their involvement in the group amongst themselves. The fewer people who know, the less evidence there is in the long run.

SECURITY VIOLATING BEHAVIOURS

In an attempt to impress others, activists may behave in ways that compromise security. Some people do this frequently – they are habitually gossiping and bragging. Some activists say inappropriate things only when they consume alcohol. Many activists make occasional breeches of security because there was a momentary temptation to say something or hint at something that shouldn't have been said or implied. In most every situation, the desire to be accepted is the root cause. Those people who tend to be the greatest security risks are those activists who have low self-esteem and strongly desire the approval of their peers. Certainly it is natural to seek friendship and recognition for our efforts, but it is imperative that we keep these selfish desires in-check so we do not jeopardize the safety of other activists or ourselves. People who place their desire for friendship over the importance of the cause can do serious damage to our security. The following are examples of security-violating behaviors:

Lying: To impress others, liars claim to have done illegal actions. Such lies not only compromise the person's security--as cops will not take what is said as a lie--but also hinders movement solidarity and trust.

Gossiping: Some weak characters think they can win friends because they are privy to special information. These gossips will tell others about who did what action or, if they don't know who did it, guess at who they think did what actions or just spread rumors about who did it. This sort of talk is very damaging. People need to remember that rumors are all that are needed to instigate a grand jury or other investigation.

Bragging: Some people who partake in illegal direct action might be tempted to brag about it to their friends. This not only jeopardizes the bragger's security, but also that of the other people involved with the action (as they may be suspected by association), as well as the people who he told (they can become accessories after the fact). An activist who brags also sets a horrible example to other activists.

Indirect-Bragging: Indirect- braggers are people who make a big production on how they want to remain anonymous, avoid protests, and stay "underground." They might not come out and say that they do illegal direct action, but they make sure everyone within ear- shot knows they are up to something. They are no better than braggers, but they try to be more sophisticated about it by pretending to maintain security. However, if they were serious about security, they would just make up a good excuse as to why they are not as active, or why they can't make it to the protest (that kind of lying is acceptable).

EDUCATE TO LIBERATE

It is fairly easy to spot those activists who compromise our movement's security. So what do we do with people who exhibit these behaviors? Do we excommunicate them from our movement? Actually, no—at least, not for a first offense.

The unfortunate truth is there are numerous security-ignorant people in the movement and others who have possibly been raised in a "scene" that thrives on bragging and gossiping. It doesn't mean these people are bad, but it does mean they need to be educated. Even seasoned activists can make mistakes when there is a general lack of security consciousness in our groups. And that's where those of you who are reading this can help. We must NEVER let a breach in security occur without acting to correct it. If an acquaintance of yours is bragging about doing an action or spreading security-compromising gossip, it is your responsibility to explain to her or him why that sort of talk violates security and is inappropriate.

You should strive to educate this person in a manner that encourages him to listen and to change his behavior. It should be done without damaging his pride. You should be humble and sincerely interested in helping him to become a better person and a more effective activists. Do not maintain a "holier than-thou" attitude. This will inevitably raise his defenses and prevent him from absorbing or using any of the advice you offer. Remember, the goal of educating people is to change their behavior, not boost your ego by showing them how much more security-conscious you are.

If possible the educational session should be done in private, so the person does not have to contend with the potential 'pride' issues. The educational reprimand should also be done as soon as possible after the mistake to increase its effectiveness. If each of us takes on the responsibility of educating those who slip up, we can dramatically improve movement security. Once people recognize lying, gossiping, bragging, and indirect-bragging as the damaging behaviors that they are, they will soon end. When we develop a culture where all breaches of security result in an immediate reprimand, all sincere activists will quickly get with the program.

DEALING WITH CHRONIC SECURITY PROBLEMS

So what do we do with activists who repeatedly violate security precautions even after multiple educational sessions? It's unfortunate, but the best thing to do with these people is cut them loose and kick them out of our meetings, base camps and organizations. With law enforcement budgets on the increase and with courts handing down long sentences for political "crimes", the stakes are too high to allow chronic security-offenders to work among us.

By creating a security culture, we have an effective defense against informers and agents who try to infiltrate groups. Imagine an informer who, every time she asked another activist about that person's activity, received a reprimand and an education on security. That informer would get frustrated really easily. Once the activists discovered she continued to violate security precautions after being repeatedly educated, they would have grounds for her dismissal. And that would be one less informer for us to deal with!

A Brief Primer on the Canadian State Security Apparatus

Recent repression against activists in British Columbia illuminates the need for grassroots people to understand and practice movement security. Police monitoring, infiltration and agent provocateurs are all techniques used by the state routinely against activists to turn up information about the activities of our movements and ourselves.

Although many activists have trouble believing that state security agencies have that much interest in their affairs, a few key court cases and hearings have helped activists to gain access to information that proves that police spying on activists is routine in Canada.

During the APEC hearings, it was revealed that over seventy groups and individuals were monitored before and during the APEC meetings in 1997. A paid industry informant/disruptor was identified at a wilderness action camp in 1999, and local activists have been targeted by provocateurs who have tried to convince them not only to disclose information but to break the law.

The Canadian security apparatus has identified a number of our movements as threatening to national security. They have targeted people and organizations widely. Even avowed pacifists have been included in surveillance and repressive measures. According to Canadian Security Intelligence

Service (CSIS) annual reports of the past five years, the Native Resistance and the Environmental/Animal Rights movements have been primary targets.

With the rise in militant First Nations struggles, covert direct action against corporations, and the growing focus by the media on general "anarchist" politics due to events in Seattle around the WTO among other major increases in movement strength and militancy, we can be pretty sure that this has been marked by a growing level of surveillance and monitoring as well.

The need for security in our movements is obvious – however, it is incredibly important that we don't fall into the trap of using our awareness of security issues to shut other people out of our growing movements. One of the key aims of COINTEL-PRO operations against the Black Panthers and American Indian Movement was to spread paranoia and distrust among those freedom fighters so that they would find it hard to accept new people into their work.

It is possible to build a movement large and at the same time create security culture. Arming ourselves with knowledge about how the system works against activists is the first step to creating that culture. The central aim of this article is to give a brief run-down of how domestic intelligence works in Canada so that we can better understand how to avoid its traps.

AN OVERVIEW OF DOMESTIC INTELLIGENCE ORGANIZATIONS

The Canadian Security and Intelligence Service (CSIS) is probably the best known of the "security" agencies that deal with activist "threats". They were originally a special surveillance wing of the RCMP until 1983 when they were split off into a separate agency due to protests that they were acting as a secret organization that was contravening Canadians' democratic rights to organize. Essentially, the split from the RCMP allowed the new spy agency to do legally what the Mounties had been doing illegally. At the operations level, the new agency was granted more freedom and more leeway than the Mounties ever had.

Today they continue to carry out a wide range of surveillance. As they are not a law-enforcement agency and therefore their evidence is not used in court, there is nothing stopping them from contravening the few regulations that exist regarding privacy rights. For example, CSIS is not required to inform people, as the RCMP does, ninety days after they have been wiretapped or bugged.

Agents working for CSIS are allowed, with "authorization", to enter people's homes to plant bugs, wiretap phones, open mail and look into health, employment and government records without ever having to tell a targeted individual what they are doing. The information that they gather is used to build profiles and dossiers (files) on individuals, organizations, networks, etc. The information that they gather is often passed on to other wings of the federal security system who are responsible for "law enforcement", and will then obtain whatever warrants are necessary for legal surveillance (to be brought into court as evidence).

The National Security Investigation Service (NSIS) is the primary law-enforcement wing of domestic security. The NSIS is a section of the Royal Canadian Mounted Police (RCMP). Most cities across Canada have an NSIS office including Vancouver, Edmonton, Montreal, Ottawa, and Toronto. The NSIS maintains a computer database on activists, immigrants and so called "terrorists" which is housed in Ottawa.

It is believed that the Vancouver NSIS employs between 12 and 18 members. Within NSIS there are several sub-groups called Team 1, Team 2, Team 3 – etc. that have different investigative targets.

They employ informants, infiltrators, personal physical surveillance, electronic surveillance including phone and room "bugs" and other means of investigation and research.

The RCMP/NSIS also have other resources at their disposal during counter-insurgency operations. "Special O" is a team of surveillance specialists that may be called upon. "Special I" is a penetration team whose specialty is to break into homes, vehicles and other properties for investigative purposes. They are the team, which among other things, installs listening devices, photo- graphs building interiors, etc.

In a long-running case based in Vancouver, all of these methods of surveillance were used against several Vancouver activists. During the Vancouver investigation, house and vehicle bugs were located by some targeted individuals. The bugs had large battery packs attached to facilitate less frequent battery changes. The NSIS also visited several activists across Canada in an attempt to question them regarding the individuals under investigation.

It cannot be stressed enough that no one is under any legal obligation to provide the police with any information other than one's own name and address. That is it. Saying anything more jeopardizes individuals' and movement security. Even answering seemingly insignificant questions can assist the police in developing personality profiles on a range of activists which may not contain "evidence" but may instead be used to give police "leads" on other suspects and to construct intent during legal proceedings. The only principled response to police questioning is to say nothing more than name and address.

The Communications Security Establishment is an agency of the defense department which has been long clouded in secrecy. They collect and process telephone, fax and computer communications of foreign states, corporations and individuals. The federal government uses the intelligence gleaned from the data to support troops abroad, catch "terrorists" and "further Canada's economic goals" (and what that means is up to them).

Although the CSE is not technically allowed to collect the communications of Canadian citizens, it is known to be a partner in the Echelon project - a multinational monitoring operation which sees CSE and counterpart agencies in the United States, Britain, Australia and New Zealand share intercepted communications of interest with one another, effectively creating a global surveillance web.

The Terrorist Extremist Section (TES Unit) is British Columbia's anti-terrorist unit. A joint Vancouver/Victoria Police Department/ RCMP unit called the Organized Crime Agency (formerly the Coordinated Law Enforcement Unit - CLEU), it is believed that the this unit employs two or three members only. Most activists will be intimately familiar with their local police forces. Be aware that cops do not only show up in blue uniforms - but routinely practice crowd infiltration and carry out surveillance and investigative activities either alone or jointly with the RCMP depending on the type of case. Watch for them on demonstrations - as they like to come along and take photo- graphs and video for the record - and they often appear in crowds as "fellow demonstrators".

THE COUNTER- INSURGENCY MODEL

Most Western nation-states follow a model of counter- insurgency developed by a British intelligence expert named Kitson who wrote, *Low Intensity Operations*, after much field work in the colonies. He broke down movement development into three stages:

The Preparatory Phase: is when the movement is small, tends to focus on education, publishing and groundwork.

The Non-Violent Phase: is when the movement takes on more of a mass character. Large demonstrations are the norm.

In the Insurgency Phase: the movement has taken on a popular character. Perhaps a more assertive, guerrilla component has emerged.

Kitson advises that the primary work of the intelligence agency should occur during the preparatory phase. At this time the movements are most vulnerable. They have not experienced a high degree of repression. They consider talk of security as mere paranoia. As they are not breaking laws they believe that it is safe to organize completely openly. The intelligence agency is therefore able to exploit these conditions and develop detailed dossiers on a wide range of people. The information will be extremely valuable to them later on. It is important that as a movement in we need to learn to practice security at all points in the movement's development. Remember that the State is interested in knowing about activists' beliefs, not just in "hard evidence". Learn and practice security to protect ourselves and our peoples. Don't be afraid. Remember - If an agent comes knockin', do no talkin'.

Everything You Ever Wanted to Know about Informers and Infiltrators

Informants and infiltrators operate in every radical movement. The rise of militant radicalism as seen at the WTO protests in Seattle, and the declarations by activists to continue the struggle in the streets and underground – mean that more and more attention will be paid to activists by law enforcement. Part of this will mean sending more infiltrators amongst our ranks to bribe and entice those weak individuals already involved.

Non-violent movements need to learn to identify such people and and let them know that their actions will never be tolerated by activists in any way.

This section is intended to arm you with information on how to spot and deal with informers and infiltrators in our ranks.

Who is an informer? There are actually two kinds of informers. The deliberate informer is someone who infiltrates an organization with the specific intent of getting incriminating evidence against activists or even setting them up to be arrested. These infiltrators are either on the payroll of a government agency or may be hired by industry. The second type of informer is the activist- turned-informant--either unwittingly or because of pressure put on them by the authorities. Make no mistake, both kinds exist throughout our ranks and are equally dangerous.

Let's discuss the deliberate informer (infiltrator) first. They are often difficult to identify, they come in all ages and types, but they usually have a similar modus operandi--they come out of nowhere and all of a sudden, they are everywhere. Whether it's a meeting, a protest, or an action, this person will be right in the thick of it.

Keep in mind however that this is also the hallmark of a new activist, whose enthusiasm and commitment is so strong that s/ he wants to fight the power every minute of the day.

How to tell them apart? Well, a planted infiltrator will ask a lot of questions about the direct action groups, individuals and illegal activities. S/he will suggest targets and volunteer to do reconnaissance as well as take part in the action.

An example of infiltration tactics can be found in an incident that occurred a few years ago when U.S. Surgical hired a security firm to infiltrate Friends of Animals in Connecticut. Their operative convinced an activist to put a pipe bomb in the car of the president of U.S. Surgical. Needless to say, the police were waiting for her and she ended up being charged with attempted murder.

State and industry infiltrators have been identified in operation in British Columbia over the past few years - attempting to incite illegal activity, sowing disruption in action camps, and gathering information on the who, what and when of our movement's activities .

Everyone who asks a lot of questions about the direct action isn't necessarily an infiltrator, but they ARE the ones to watch (at the very least, we should be educating them about security culture). Explain to new activists that direct action tactics can be risky (though some risks are worth taking!) and that it is dangerous to ask a lot of questions about it. If the person persists in asking questions, **STAY AWAY FROM THEM!** Any activist who can't understand the need for security is someone that should be held at arm's length from the movement.

Placing infiltrators into social justice movements isn't anything new. It was done to the Black Panthers and the peace movement in a big way. Unless you are only working with people you've known for years and who have earned your trust, you should assume there is an informant in your midst and act accordingly.

This doesn't mean that no one else should ever be allowed into the "inner circle." On the contrary, if our movement is to continue to grow, we must always be recruiting new members; we just need to keep security uppermost in our minds and exercise caution at all times.

Possibly an even greater threat is the activist-turned-informer, either unwittingly or through coercion.

The unwitting informer is the activist who can't keep his/her mouth shut. If someone brags to you about what s/he's done, make sure this person never has any knowledge that can incriminate you, because sooner or later, the wrong person will hear of it. These activists don't mean to do harm, but the results of their bragging can be serious. It is your responsibility to instruct these people on security culture and the importance of it.

The other type of activist- informer is person who cracks under pressure and starts talking to save his/her own skin. Many activists get drawn into situations they are not able to handle, and some are so caught up in the "excitement" that they either don't realize what the consequences can be or they just don't think they'll ever have to face them.

We have to know the possible consequences of every action we take and be prepared to deal with them. Someone who is easily influenced by his/her parents or dependent on them for support is not a good candidate for actions as they can be persuaded too easily to cooperate with the authorities. There is no shame in not being able to do an action because of responsibilities that make it impossible to do jail time. If others are depending on you for support or you aren't willing to lose your job or drop out of school, **DON'T DO THE ACTION.**

Make certain that others in your affinity group are not in situations which may cause them to cooperate with the police or abandon their friends.

Two activists were recently put in jail in Canada because a third party panicked - mainly about not being able to get his drugs in jail - and talked to free himself. (This is not to condemn those who have drug habits or criminal records – but are certainly things to keep in mind). Don't be afraid to talk about this. Ask hard questions, and if you aren't convinced that someone will be able to stay strong if the worst happens, then designate that person to do support. Make sure that those who go into battle with you are willing and able to take whatever comes, even if it means giving up their freedom for your goals. Remember - there is no excuse for turning in action comrades to the police - and those activists that do effectively excommunicate themselves from our movements. We must offer no legal or jail support to those activists who turn-in others for their impact on our movement is far-reaching and can have devastating effects. Some things to look out for in people you choose to do illegal direct action with are lengthy

Crypto Anarchy and Virtual Communities

Timothy C. May

*535 Monterey Drive
Aptos, CA 95003 U.S.A.
tcmay@netcom.com
December, 1994*

Extended Abstract

The combination of strong, unbreakable public key cryptography and virtual network communities in cyberspace will produce interesting and profound changes in the nature of economic and social systems. Crypto anarchy is the cyberspatial realization of anarcho-capitalism, transcending national boundaries and freeing individuals to make the economic arrangements they wish to make consensually.

Strong cryptography, exemplified by RSA (a public key algorithm) and PGP (Pretty Good Privacy), provides encryption that essentially cannot be broken with all the computing power in the universe. This ensures security and privacy. Public key cryptography is rightly considered to be a revolution.

Digital mixes, or anonymous remailers, use crypto to create untraceable e-mail, which has many uses. (Numerous anonymous remailers, in several countries, are now operating. Message traffic is growing exponentially.)

Digital pseudonyms, the creation of persistent network personas that cannot be forged by others and yet which are unlinkable to the "true names" of their owners, are finding major uses in ensuring free speech, in allowing controversial opinions to be aired, and in providing for economic transactions that cannot be blocked by local governments. The technology being deployed by the Cypherpunks and others, means their identities, nationalities, and even which continents they are on are untraceable -- unless they choose to reveal this information. This alters the conventional "relationship topology" of the world, allowing diverse interactions without external governmental regulation, taxation, or interference

Digital cash, untraceable and anonymous (like real cash), is also coming, though various technical and practical hurdles remain. "Swiss banks in cyberspace" will make economic transactions much more liquid and much less subject to local rules and regulations. Tax avoidance is likely to be a major attraction for many. An example of local interest to Monte Carlo might be the work underway to develop anonymous, untraceable systems for "cyberspace casinos." While not as attractive to many as elegant casinos, the popularity of "numbers games" and bookies in general suggests a opportunity to pursue.

Data havens and information markets are already springing up, using the methods described to make information retrievable anonymously and untraceably.

Governments see their powers eroded by these technologies, and are taking various well-known steps to try to limit the use of strong crypto by their subjects. The U.S. has several well-publicized efforts, including the Clipper chip, the Digital Telephony wiretap law, and proposals for "voluntary" escrow of cryptographic keys. Cypherpunks and others expect these efforts to be bypassed. Technology has let the genie out of the bottle. Crypto anarchy is liberating individuals from coercion by their physical neighbors--who cannot know who they are on the Net--and from governments. For libertarians, strong crypto provides the means by which government will be avoided.

The presentation will describe how several of these systems work, briefly, and will outline the likely implications of this combination of crypto anarchy and virtual cyberspace communities.

1 Introduction

This paper describes the combination of two major technologies:

- Strong Crypto: including encryption, digital signatures, digital cash, digital mixes (remailers), and related technologies.
- Cyberspatial Virtual Communities: including networks, anonymous communications, MUDs and MOOs, and "Multiverse"-type virtual realities.

These areas have generally remained separate, at least in published papers. Certainly the developers of cyberspace systems, such as MUDs, MOOs, and Habitat-like systems, appreciate the importance of cryptography for user authentication, overall security, and certainly for (eventual) digital purchase of services. But for the most part the combination of these two areas has been the province of the science fiction writer, notably writers such as Vernor Vinge, William Gibson, Bruce Sterling, and Orson Scott Card.

The "Cypherpunks" group, a loose, anarchic mailing list and group of hackers, was formed by several of us in 1992 as a group to make concrete some of the abstract ideas often presented at conferences. We've had some successes, and some failures. [1] The Cypherpunks group also appeared at a fortuitous time, as PGP was becoming popular, as Wired magazine appeared (they featured us on the cover of their second issue), and as the publicity (hype?) about the Information Superhighway and the World Wide Web reached a crescendo.

The site ftp.csua.berkeley.edu has a number of essays and files, including crypto files, in the directory pub/cypherpunks. I have also written/ compiled a very large 1.3 MB FAQ on these issues, the Cyphernomicon, available at various sites, including my ftp directory, ftp.netcom.com, in the directory pub/tc/tcmay.

The Cypherpunks group is also a pretty good example of a "virtual community." Scattered around the world, communicating electronically in matters of minutes, and seeming oblivious to local laws, the Cypherpunks are indeed a community, and a virtual one. Many members use pseudonyms, and use anonymous remailers to communicate with the list. The list itself thus behaves as a "message pool," a place where information of all sort may be anonymous deposited--and anonymous received (since everyone sees the entire list, like a newspaper, the intended recipient is anonymized).

Legal Caveat: Consult your local laws before applying any of the methods described here. In some jurisdictions, it may be illegal to even read papers like this (seriously). In particular, I generally won't be giving ftp site addresses for copies of PGP, remailer access, digital cash systems, etc. These are well-covered in more current forums, e.g., sci.crypt or talk.politics.crypto, and there are some unresolved issues about whether giving the address of such sites constitutes (or "aids and abets") violation of various export and munitions laws (crypto is considered a munition in the U.S. and probably elsewhere....some nations consider a laser printer to be a munitions item!).

2 Modern Cryptography

The past two decades have produced a revolution in cryptography (crypto, for short) the science of the making of ciphers and codes. Beyond just simple ciphers, useful mainly for keeping communications secret, modern crypto includes diverse tools for authentication of messages, for digital timestamping of documents, for hiding messages in other documents (steganography), and even for schemes for digital cash.

Public key cryptography, the creation of Diffie and Hellman, has dramatically altered the role of crypto. Coming at the same time as the wholesale conversion to computer networks and worldwide communications, it has been a key element of security, confidence, and success. The role of crypto will only become more important over the coming decades.

Pretty Good Privacy, PGP, is a popular version of the algorithm developed by Rivest, Shamir, and Adleman, known of course as RSA. The RSA algorithm was given a patent in the U.S., though not in any European countries, and is licensed commercially. [2]

These tools are described in detail in various texts and Conference proceedings, and are not the subject of this paper. [3] The focus here is on the implications of strong crypto for cyberspace, especially on virtual communities.

Mention should be made of the role of David Chaum in defining the key concepts here. In several seminal papers (for example, [4] [5]), Chaum introduced the ideas of using public key cryptography methods for anonymous, untraceable electronic mail, for digital money systems in which spender identity is not revealed, and in schemes related to these. (I make no claims of course that Chaum agrees with my conclusions about the political and socioeconomic implications of these results.)

3 Virtual Communities

Notes: cyberspace, Habitat, VR, Vinge, etc. Crypto holds up the "walls" of these cyberspatial realities. Access control, access rights, modification privileges.

Virtual communities are the networks of individuals or groups which are not necessarily closely-connected geographically. The "virtual" is meant to imply a non-physical linking, but should not be taken to mean that these are any less community-like than are conventional physical communities.

Examples include churches, service organizations, clubs, criminal gangs, cartels, fan groups, etc. The Catholic Church and the Boy Scouts are both examples of virtual communities which span the globe, transcend national borders, and create a sense of allegiance, of belonging, and a sense of "community." Likewise, the Mafia is a virtual community (with its enforcement mechanisms, its own extra-legal rules, etc.) Lots of other examples: Masons, Triads, Red Cross, Interpol, Islam, Judaism, Mormons, Sinderio Luminoso, the IRA, drug cartels, terrorist groups, Aryan Nation, Greenpeace, the Animal Liberation Front, and so on. There are undoubtedly many more such virtual communities than there are nation-states, and the ties that bind them are for the most part much stronger than are the chauvinist nationalism emotions. Any group in which the common interests of the group, be it a shared ideology or a particular interest, are enough to create a cohesive community.

Corporations are another prime example of a virtual community, having scattered sites, private communication channels (generally inaccessible to the outside world, including the authorities), and their own goals and methods. In fact, many "cyberpunk" (not cypherpunk) fiction authors make a mistake, I think, in assuming the future world will be dominated by transnational megacorporate "states." In fact, corporations are just one example--of many--of such virtual communities which will be effectively on a par with nation-states. (Note especially that any laws designed to limit use of crypto cause immediate and profound problems for corporations-countries like France and the Philippines, which have attempted to limit the use of crypto, have mostly been ignored by corporations. Any attempts to outlaw crypto will produce a surge of sudden "incorporations," thus gaining for the new corporate members the aegis of corporate privacy.)

In an academic setting, "invisible colleges" are the communities of researchers.

These virtual communities typically are "opaque" to outsiders. Attempts to gain access to the internals of these communities are rarely successful. Law enforcement and intelligence agencies (such as the NSA in the U.S., Chobetsu in Japan, SDECE in France, and so on, in every country) may infiltrate such groups and use electronic surveillance (ELINT) to monitor these virtual communities. Not surprisingly, these communities are early adopters of encryption technology, ranging from scrambled cellphones to full-blown PGP encryption. [8]

The use of encryption by "evil" groups, such as child pornographers, terrorists, abortionists, abortion protestors, etc., is cited by those who wish to limit civilian access to crypto tools. We call these the "Four Horseman of the Infocalypse," as they are so often cited as the reason why ordinary citizen-units of the nation-state are not to have access to crypto.

This is clearly a dangerous argument to make, for various good reasons. The basic right of free speech is the right to speak in a language one's neighbors or governing leaders may not find comprehensible: encrypted speech. There's not enough space here to go into the many good arguments against a limit on access to privacy, communications tools, and crypto.

The advent of full-featured communications systems for computer-mediated virtual communities will have even more profound implications. MUDs and MOOs (multi-user domains, etc.) and 3D virtual realities are one avenue, and text-centric Net communications are another. (Someday, soon, they'll merge, as described in Vernor Vinge's prophetic 1980 novella, True Names.)

4 Observability and Surveillance

An interesting way to view issues of network visibility is in terms of the "transparency" of nodes and links between nodes. Transparent means visible to outsiders, perhaps those in law enforcement or the intelligence community. Opaque mean not transparent, not visible. A postcard is transparent, a sealed letter is opaque. PGP inventor Phil Zimmermann has likened the requirement for transparency to being ordered to use postcards for all correspondence, with encryption the equivalent of an opaque envelope (envelopes can be opened, of course, and long have been).

Transparent links and nodes are the norm in a police state, such as the U.S.S.R., Iraq, China, and so forth. Communications channels are tapped, and private use of computers is restricted. (This is becoming increasingly hard to do, even for police states; many cite the spread of communications options as a proximate cause of the collapse of communism in recent years.)

There are interesting "chemistries" or "algebras" of transparent vs. opaque links and nodes. What happens if links must be transparent, but nodes are allowed to be opaque? (The answer: the result is as if opaque links and nodes were allowed, i.e., full implications of strong crypto. Hence, any attempt to ban communications crypto while still allowing private CPUs to exist....)

If Alice and Bob are free to communicate, and to choose routing paths, then Alice can use "crypto arbitrage" (a variation on the term, "regulatory arbitrage," the term Eric Hughes uses to capture this idea of moving transactions to other jurisdictions) to communicate with sites--perhaps in other countries--that will perform as she wishes. This can mean remailing, mixing, etc. As an example, Canadian citizens who are told they cannot access information on the Homolka-Teale murder case (a controversial case in which the judge has ordered the media in Canada, and entering Canada, not to discuss the gory details) nevertheless have a vast array of options, including using telnet, gopher, ftp, the Web, etc., to access sites in many other countries--or even in no country in particular.

Most of the consequences described here arise from this chemistry of links and nodes: unless nearly all node and links are forced to be transparent, including links to other nations and the nodes in those nations, then the result is that private communication can still occur. Crypto anarchy results.

5 Crypto Anarchy

"The Net is an anarchy." This truism is the core of crypto anarchy. No central control, no ruler, no leader (except by example, reputation), no "laws." No single nation controls the Net, no administrative body sets policy. The Ayatollah in Iran is as powerless to stop a newsgroup--alt.wanted.moslem.women or alt.wanted.moslem.gay come to mind--he doesn't like as the President of France is as powerless to stop, say, the abuse of French in soc.culture.french. Likewise, the CIA can't stop newsgroups, or sites, or Web pages, which give away their secrets. At least not in terms of the Net itself...what non-Net steps might be taken is left as an exercise for the paranoid and the cautious.

This essential anarchy is much more common than many think. Anarchy--the absence of a ruler telling one what to do--is common in many walks of life: choice of books to read, movies to see, friends to socialize with, etc. Anarchy does not mean complete freedom--one can, after all, only read the books which someone has written and had published--but it does mean freedom from external coercion. Anarchy as a concept, though, has been tainted by other associations.

First, the "anarchy" here is not the anarchy of popular conception: lawlessness, disorder, chaos, and "anarchy." Nor is it the bomb-throwing anarchy of the 19th century "black" anarchists, usually associated with Russia and labor movements. Nor is it the "black flag" anarchy of anarcho-syndicalism and writers such as Proudhon. Rather, the anarchy being spoken of here is the anarchy of "absence of government" (literally, "an arch," without a chief or head).

This is the same sense of anarchy used in "anarchocapitalism," the libertarian free market ideology which promotes voluntary, uncoerced economic transactions. [6] I devised the term crypto anarchy as a pun on crypto, meaning "hidden," on the use of "crypto" in combination with political views (as in Gore Vidal's famous charge to William F. Buckley: "You crypto fascist!"), and of course because the technology of crypto makes this form of anarchy possible. The first presentation of this was in a 1988 "Manifesto," whimsically patterned after another famous manifesto. [7] Perhaps a more popularly understandable term, such as "cyber liberty," might have some advantages, but crypto anarchy has its own charm, I think.

And anarchy in this sense does not mean local hierarchies don't exist, nor does it mean that no rulers exist. Groups outside the direct control of local governmental authorities may still have leaders, rulers, club presidents, elected bodies, etc. Many will not, though.

Politically, virtual communities outside the scope of local governmental control may present problems of law enforcement and tax collection. (Some of us like this aspect.) Avoidance of coerced transactions can mean avoidance of taxes, avoidance of laws saying who one can sell to and who one can't, and so forth. It is likely that many will be unhappy that some are using cryptography to avoid laws designed to control behavior.

National borders are becoming more transparent than ever to data. A flood of bits crosses the borders of most developed countries--phone lines, cables, fibers, satellite up/downlinks, and millions of diskettes, tapes, CDs, etc. Stopping data at the borders is less than hopeless.

Finally, the ability to move data around the world at will, the ability to communicate to remote sites at will, means that a kind of "regulatory arbitrage" can be used to avoid legal roadblocks. For example, remailing into the U.S. from a site in the Netherlands...whose laws apply? (If one thinks that U.S. laws should apply to sites in the Netherlands, does Iraqi law apply in the U.S.? And so on.)

This regulatory arbitrage is also useful for avoiding the welter of laws and regulations which operations in one country may face, including the "deep pockets" lawsuits so many in the U.S. face. Moving operations on the Net outside a litigious jurisdiction is one step to reduce this business liability. Like Swiss banks, but different.

6 True Names and Anonymous Systems

Something needs to be said about the role of anonymity and digital pseudonyms. This is a topic for an essay unto itself, of course.

Are true names really needed? Why are they asked for? Does the nation-state have any valid reason to demand they be used?

People want to know who they are dealing with, for psychological/evolutionary reasons and to better ensure traceability should they need to locate a person to enforce the terms of a transaction. The purely anonymous person is perhaps justifiably viewed with suspicion.

And yet pseudonyms are successful in many cases. And we rarely know whether someone who presents himself by some name is "actually" that person. Authors, artists, performers, etc., often use pseudonyms. What matters is persistence, and nonforgeability. Crypto provides this.

On the Cypherpunks list, well-respected digital pseudonyms have appeared and are thought of no less highly than their "real" colleagues are.

The whole area of digitally-authenticated reputations, and the "reputation capital" that accumulates or is affected by the opinions of others, is an area that combines economics, game theory, psychology, and expectations. A lot more study is needed.

It is unclear if governments will move to a system of demanding "Information Highway Driver's Licenses," figuratively speaking, or how systems like this could ever be enforced. (The chemistry of opaque nodes and links, again.)

7 Examples and Uses

It surprises many people that some of these uses are already being intensively explored. Anonymous remailers are used by tens of thousands of persons--and perhaps abused. [13] And of course encryption, via RSA, PGP, etc., is very common in some communities. (Hackers, Net users, freedom fighters, white separatists, etc....I make no moral judgments here about those using these methods).

Remailers are a good example to look at in more detail. There are two current main flavors of remailers:

1. "Cypherpunk"-style remailers, which process text messages to redirect mail to another sites, using a command syntax that allows arbitrary nesting of remailing (as many sites as one wishes), with PGP encryption at each level of nesting.
2. "Julf"-style remailer(s), based on the original work of Karl Kleinpaste and operated/maintained by Julf Helsingius, in Finland. No encryption, and only one such site at present. (This system has been used extensively for messages posted to the Usenet, and is basically successful. The model is based on operator trustworthiness, and his location in Finland, beyond the reach of court orders and subpoenas from most countries.)

The Cypherpunks remailers currently number about 20, with more being added every month. There is no reason not to expect hundreds of such remailers in a few years.

One experimental "information market" is BlackNet, a system which appeared in 1993 and which allows fully-anonymous, two-way exchanges of information of all sorts. There are reports that U.S. authorities have investigated this because of its presence on networks at Defense Department research labs. Not much they can do about it, of course, and more such entities are expected.

(The implications for espionage are profound, and largely unstoppable. Anyone with a home computer and access to the Net or Web, in various forms, can use these methods to communicate securely, anonymously or pseudonymously, and with little fear of detection. "Digital dead drops" can be used to post information obtained, far more securely than the old physical dead drops...no more messages left in Coke cans at the bases of trees on remote roads.)

Whistleblowing is another growing use of anonymous remailers, with folks fearing retaliation using remailers to publicly post information. (Of course, there's a fine line between whistleblowing, revenge, and espionage.)

Data havens, for the storage and marketing of controversial information is another area of likely future growth. Nearly any kind of information, medical, religious, chemical, etc., is illegal or proscribed in one or more countries, so those seeking this illegal information will turn to anonymous messaging systems to access--and perhaps purchase, with anonymous digital cash--this information. This might include credit data bases, deadbeat renter files, organ bank markets, etc. (These are all things which have various restrictions on them in the U.S., for example....one cannot compile credit data bases, or lists of deadbeat renters, without meeting various restrictions. A good reason to move them into cyberspace, or at least outside the U.S., and then sell access through remailers.)

Matching buyers and sellers of organs is another such market. A huge demand (life and death), but various laws tightly controlling such markets.

Digital cash efforts. A lot has been written about digital cash. [14] [15] David Chaum's company, DigiCash, has the most interesting technology, and has recently begun market testing. Stefan Brands may or may not have a competing system which gets around some of Chaum's patents. (The attitude crypto anarchists might take about patents is another topic for discussion. Suffice it to say that patents and other intellectual property issues continue to have relevance in the practical world, despite erosion by technological trends.)

Credit card-based systems, such as the First Virtual system, are not exactly digital cash, in the Chaumian sense of blinded notes, but offer some advantages the market may find useful until more advanced systems are available.

I expect to see many more such experiments over the next several years, and some of them will likely be market successes.

8 Commerce and Colonization of Cyberspace

How will these ideas affect the development of cyberspace?

"You can't eat cyberspace" is a criticism often levelled at argument about the role of cyberspace in everyday life. The argument made is that money and resources "accumulated" in some future (or near-future) cyberspatial system will not be able to be "laundered" into the real world. Even such a prescient thinker as Neal Stephenson, in *Snow Crash*, had his protagonist a vastly wealthy man in "The Multiverse," but a near-pauper in the physical world.

This is implausible for several reasons. First, we routinely see transfers of wealth from the abstract world of stock tips, arcane consulting knowledge, etc., to the real world. "Consulting" is the operative word. Second, a variety of means of laundering money, via phony invoices, uncollected loans, art objects, etc., are well-known to those who launder money...these methods, and more advanced ones to come, are likely to be used by those who wish their cyberspace profits moved into the real world.

(Doing this anonymously, untraceably, is another complication. There may be methods of doing this--proposals have looked pretty solid, but more work is needed.)

The World Wide Web is growing at an explosive pace. Combined with cryptographically-protected communication and digital cash of some form (and there are several being tried), this should produce the long-awaited colonization of cyberspace.

Most Net and Web users already pay little attention to the putative laws of their local regions or nations, apparently seeing themselves more as members of various virtual communities than as members of locally-governed entities. This trend is accelerating.

Most importantly, information can be bought and sold (anonymously, too) and then used in the real world. There is no reason to expect that this won't be a major reason to move into cyberspace.

9 Implications

I've touched on the implications in several places. Many thoughtful people are worried about some of the possibilities made apparent by strong crypto and anonymous communication systems. Some are proposing restrictions on access to crypto tools. The recent debate in the U.S. over "Clipper" and other key escrow systems shows the strength of emotions on this issue.

Abhorrent markets may arise. For example, anonymous systems and untraceable digital cash have some obvious implications for the arranging of contract killings and such. (The greatest risk in arranging such hits is that physical meetings expose the buyers and sellers of such services to stings. Crypto anarchy lessens, or even eliminates, this risk, thus lowering transaction costs. The risks to the

actual triggermen are not lessened, but this is a risk the buyers need not worry about. Think of anonymous escrow services which hold the digital money until the deed is done. Lots of issues here. It is unfortunate that this area is so little-discussed....people seem to have an aversion for exploring the logical consequences in such areas.)

The implications for corporate and national espionage have already been touched upon. Combined with liquid markets in information, this may make secrets much harder to keep. ((Imagine a "Digital Jane's," after the military weapons handbooks, anonymously compiled and sold for digital money, beyond the reach of various governments which don't want their secrets told.)

New money-laundering approaches are of course another area to explore.

Something that is inevitable is the increased role of individuals, leading to a new kind of elitism. Those who are comfortable with the tools described here can avoid the restrictions and taxes that others cannot. If local laws can be bypassed technologically, the implications are pretty clear.

The implications for personal liberty are of course profound. No longer can nation-states tell their citizen-units what they can have access to, not if these citizens can access the cyberspace world through anonymous systems.

10 How Likely?

I am making no bold predictions that these changes will sweep the world anytime soon. Most people are ignorant of these methods, and the methods themselves are still under development. A wholesale conversion to "living in cyberspace" is just not in the cards, at least not in the next few decades.

But to an increasingly large group, the Net is reality. It is where friends are made, where business is negotiated, where intellectual stimulation is found. And many of these people are using crypto anarchy tools. Anonymous remailers, message pools, information markets. Consulting via pseudonyms has begun to appear, and should grow. (As usual, the lack of a robust digital cash system is slowing things down.

Can crypto anarchy be stopped? Although the future evolution is unclear, as the future almost always is, it seems unlikely that present trends can be reversed:

- Dramatic increases in bandwidth and local, privately-owned computer power.
- Exponential increase in number of Net users.
- Explosion in "degrees of freedom" in personal choices, tastes, wishes, goals.
- Inability of central governments to control economies, cultural trends, etc. [9]

The Net is integrally tied to economic transactions, and no country can afford to "disconnect" itself from it. (The U.S.S.R. couldn't do it, and they were light-years behind the U.S., European, and Asian countries. And in a few more years, no hope of limiting these tools at all, something the U.S. F.B.I. has acknowledged. [11]

Technological Inevitability: These tools are already in widespread use, and only draconian steps to limit access to computers and communications channels could significantly impact further use. (Scenarios for restrictions on private use of crypto.)

As John Gilmore has noted, "the Net tends to interpret censorship as damage, and routes around it." This applies as well to attempts to legislate behavior on the Net. (The utter impossibility of regulating the worldwide Net, with entry points in more than a hundred nations, with millions of machines, is not yet fully recognized by most national governments. They still speak in terms of "controlling" the Net, when in fact the laws of one nation generally have little use in other countries.)

Digital money in its various forms is probably the weakest link at this point. Most of the other pieces are operational, at least in basic forms, but digital cash is (understandably) harder to deploy. Hobbyist or "toy" experiments have been cumbersome, and the "toy" nature is painfully obvious. It is not easy to use digital cash systems at this time ("To use Magic Money, first create a client..."), especially as compared to the easily understood alternatives. [12] People are understandably reluctant to entrust actual money to such systems. And it's not yet clear what can be bought with digital cash (a chicken or egg dilemma, likely to be resolved in the next several years).

And digital cash, digital banks, etc., are a likely target for legislative moves to limit the deployment of crypto anarchy and digital economies. Whether through banking regulation or tax laws, it is not likely that digital money will be deployed easily. "Kids, don't try this at home!" Some of the current schemes may also incorporate methods for reporting transactions to the tax authorities, and may include "software key escrow" features which make transactions fully or partly visible to authorities.

11 Conclusions

Strong crypto provides new levels of personal privacy, all the more important in an era of increased surveillance, monitoring, and the temptation to demand proofs of identity and permission slips. Some of the "credentials without identity" work of Chaum and others may lessen this move toward a surveillance society.

The implications are, as I see it, that the power of nation-states will be lessened, tax collection policies will have to be changed, and economic interactions will be based more on personal calculations of value than on societal mandates.

Is this a Good Thing? Mostly yes. Crypto anarchy has some messy aspects, of this there can be little doubt. From relatively unimportant things like price-fixing and insider trading to more serious things like economic espionage, the undermining of corporate knowledge ownership, to extremely dark things like anonymous markets for killings.

But let's not forget that nation-states have, under the guise of protecting us from others, killed more than 100 million people in this century alone. Mao, Stalin, Hitler, and Pol Pot, just to name the most extreme examples. It is hard to imagine any level of digital contract killings ever coming close to nationstate barbarism. (But I agree that this is something we cannot accurately speak about; I don't think we have much of a choice in embracing crypto anarchy or not, so I choose to focus on the bright side.)

It is hard to argue that the risks of anonymous markets and tax evasion are justification for worldwide suppression of communications and encryption tools. People have always killed each other, and governments have not stopped this (arguably, they make the problem much worse, as the wars of this century have shown).

Also, there are various steps that can be taken to lessen the risks of crypto anarchy impinging on personal safety. [10]

Strong crypto provides a technological means of ensuring the practical freedom to read and write what one wishes to. (Albeit perhaps not in one's true name, as the nation-state-democracy will likely still try to control behavior through majority votes on what can be said, not said, read, not read, etc.) And of course if speech is free, so are many classes of economic interaction that are essentially tied to free speech.

A phase change is coming. Virtual communities are in their ascendancy, displacing conventional notions of nationhood. Geographic proximity is no longer as important as it once was.

A lot of work remains. Technical cryptography still hasn't solved all problems, the role of reputations (both positive and negative) needs further study, and the practical issues surrounding many of these areas have barely been explored.

We will be the colonizers of cyberspace.

12 Acknowledgments

My thanks to my colleagues in the Cypherpunks group, all 700 of them, past or present. Well over 100 megabytes of list traffic has passed through the Cypherpunks mailing list, so there have been a lot of stimulating ideas. But especially my appreciation goes to Eric Hughes, Sandy Sandfort, Duncan Frissell, Hal Finney, Perry Metzger, Nick Szabo, John Gilmore, Whit Diffie, Carl Ellison, Bill Stewart, and Harry Bartholomew. Thanks as well to Robin Hanson, Ted Kaehler, Keith Henson, Chip Morningstar, Eric Dean Tribble, Mark Miller, Bob Fleming, Cherie Kushner, Michael Korn, George Gottlieb, Jim Bennett, Dave Ross, Gayle Pergamit, and--especially--the late Phil Salin. Finally, thanks for valuable discussions, sometimes brief, sometimes long, with Vernor Vinge, David Friedman, Rudy Rucker, David Chaum, Kevin Kelly, and Steven Levy.

13 References and Notes

1 The Cypherpunks group was mainly formed by Eric Hughes, Tim May, and John Gilmore. It began both physical meetings, in the Bay Area and elsewhere, and virtual meetings on an unmoderated mailing list. The name was provided by Judith Milhon, as a play on the "cyberpunk" genre and the British spelling of cipher. The mailing list can be subscribed to by sending the single message subscribe cypherpunks in the body of a message to majordomo@toad.com. Expect at least 50 messages a day. About 600 subscribers in many countries are presently on the list. Some are pseudonyms.

2 RSA Data Security Inc., Redwood Shores, California, is the license administrator. Contact them for details.

3 Many crypto texts exist. A good introduction is Bruce Schneier's Applied Cryptography, John Wiley and Sons, 1994. This text includes pointers to many other sources. The "Crypto" Proceedings (Advances in Cryptology, Springer-Verlag, annually) are essential references. The annual Crypto conference in Santa Barbara, and the Eurocrypt and Auscrypt conferences, are where most crypto results are presented.

4 David Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," Comm. ACM 24, 2, February 1981, pp. 84-88. Cypherpunks-style remailers are a form of Chaum's "digital mixes," albeit far from ideal.

5 David Chaum, "Security without Identification: Transaction Systems to make Big Brother Obsolete," Comm. ACM 28, 10, October 1985. This is an early paper on digital cash...be sure to consult more recent papers.

6 David Friedman, The Machinery of Freedom, 2nd edition. A leading theoretician of anarcho-capitalism. (Hayek was another.)

7 Tim May, The Crypto Anarchist Manifesto, July 1988, distributed on the Usenet and on various mailing lists.

8 The political opposition in Myanmar--formerly Burma--is using Pretty Good Privacy running on DOS laptops in the jungles for communications amongst the rebels, according to Phil Zimmermann, author of PGP. This life-and-death usage underscores the role of crypto.

9 See Kevin Kelly's *Out of Control*, 1994, for a discussion of how central control is failing, and how the modern paradigm is one of market mechanisms, personal choice, and technological empowerment.

10 Robin Hanson and David Friedman have written extensively about scenarios for dealing with the threats of extortionists, would-be assassins, etc. I am hoping some of their work gets published someday. (Much of the discussion was in 1992-3, on the "Extropians" mailing list.)

11 During the "Digital Telephony Bill" debate, an FBI official said that failure to mandate wiretap capabilities within the next 18 months would make it all moot, as the cost would rise beyond any reasonable budget (currently \$500 million for retrofit costs).

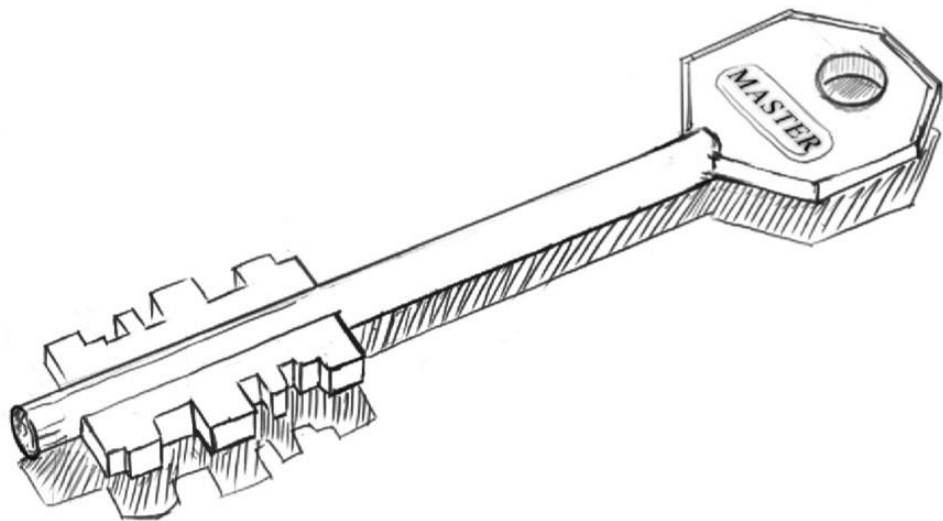
12 "Magic Money" was an experimental implementation of Chaum's digital cash system. It was coded by "Pr0duct Cypher," a pseudonymous member of the Cypherpunks list--none of us knows his real identity, as he used remailers to communicate with the list, and digitally signed his posts. Many of us found it too difficult to use, which is more a measure of the deep issues involved in using digital analogs (no pun intended) to real, physical money.

13 Abuse, according to some views, of remailers is already occurring. A Cypherpunks-type remailer was used to post a proprietary hash function of RSA Data Security, Inc. to the Usenet. (Let me hasten to add that it was not a remailer I operate, or have control over, etc.)

14 article on digital cash, *The Economist*, 26 November 1994. pp. 21-23.

15 article on digital cash, Steven Levy, *Wired*. December 1994.

the
CryptoParty
handbook



The CryptoParty Handbook

1	INTRODUCING CRYPTOPARTY	13
1.1	ABOUT THIS BOOK	13
1.2	A CRYPTOPARTY MANIFESTO	15
1.3	HOW TO CRYPTOPARTY	17
1.4	PARTY LIKE IT'S DECEMBER 31ST 1983	20
1.4.1	What is CryptoParty?	20
1.5	PREFACE	21
1.6	WHY PRIVACY MATTERS	21
2	UNDERSTANDING EMAIL	23
2.1	BASIC TIPS	23
2.1.1	In brief:	23
2.1.2	Passwords	23
2.1.3	Reading Email in Public Places	24
2.1.4	Cache Cunning	24
2.1.5	Securing your communication	25
2.1.6	DNSSEC & DANE	25
2.1.7	Account Separation	26
2.1.8	A note about hosted email	26
2.2	TYPES OF EMAIL	26
2.2.1	Remotely hosted email ('webmail'), resourced using a web browser	26
2.2.2	Remotely hosted email, resourced using an email program or using a web browser	27
2.2.3	Context considerations	27
2.2.4	Email & Metadata	28
2.2.5	Self-administered email server	28
2.2.6	'Free' email services	28
2.2.7	Non-profit	28
2.2.8	Notes on email forwarding	29
2.3	FEARS	29
2.3.1	Random abuse and theft by malicious hackers	30
2.3.2	Targeted abuse, harassment, and spying	31
2.3.3	When Encryption Goes Wrong	31
2.4	SECURE CONNECTIONS	32
2.4.1	Can other people read along when I check my email?	32
2.4.2	Notes	33

2.5	Secure Emails	33
2.5.1	What software can I use to encrypt my email?	34
3	Understanding Browsing	35
3.1	Basic Tips	35
3.1.1	In Brief:	35
3.1.2	Your browser talks about you behind your back	35
3.1.3	Web sites can track you as you browse	35
3.1.4	Searching online can give away information about you	36
3.1.5	More eyes than you can see	36
3.1.6	Your right to be unknown	37
3.2	Fears	37
3.2.1	Social Networking - what are the dangers?	37
3.2.2	Who can steal my identity?	38
3.2.3	Can I get in trouble for Googling weird stuff?	39
3.2.4	Who is keeping a record of my browsing and am I allowed to hide from them?	39
3.2.5	How to not reveal my Identity?	40
3.2.6	How to avoid being tracked?	40
3.3	What happens when you browse	40
3.3.1	A topography of you: footprints	41
3.4	Accounts and Security	43
3.4.1	Can malicious web sites take over my accounts?	43
3.5	Tracking	44
3.5.1	How do they track us?	44
3.5.2	How can I prevent tracking?	45
3.5.3	A word of warning	52
3.6	Anonymity	52
3.6.1	Intro	52
3.6.2	Proxy	52
3.6.3	Tor	53
3.7	VPN	54
4	Publishing And Distribution	57
4.1	Publishing Anonymously	57
4.1.1	Several Don'ts	58
4.2	Anonymous Email	59
4.2.1	Sending From Throw-away Email Accounts	59
4.2.2	Be Careful about what you say!	60
4.3	File Sharing	60
4.3.1	BitTorrent	61
4.3.2	SoulSeek	63
4.3.3	I2P	64
5	SECURE CALLS AND SMS	65
5.1	SECURE CALLS	65
5.1.1	iOS - Installing Signal	65
5.1.2	Android - Installing RedPhone	65

5.2	SECURE MESSAGING	66
5.2.1	Android	66
6	BASIC EMAIL SECURITY	67
6.1	START USING THUNDERBIRD	67
6.1.1	Installing Thunderbird on Windows	67
6.1.2	Installing Thunderbird on Ubuntu	72
6.1.3	Installing Thunderbird on Ubuntu 12.04 or newer	72
6.1.4	Installing Thunderbird on Mac OS X	73
6.1.5	Starting Thunderbird for the first time	76
6.2	SETTING UP SECURE CONNECTIONS	76
6.2.1	Configuration requirements	77
6.2.2	Preparing a Gmail account for use with Thunderbird	77
6.2.3	Configuring Thunderbird to use SSL/TLS	78
6.2.4	Manual setup	80
6.2.5	Finishing the setup, different encryption methods	82
6.2.6	Returning to the configuration screens	83
6.3	SOME ADDITIONAL SECURITY SETTINGS	83
6.3.1	Junk mail settings	83
6.3.2	Scam detection and warning system	84
6.3.3	Anti-virus integration	85
6.3.4	Set a master password	86
6.3.5	Adaptive junk mail controls	90
7	EMAIL ENCRYPTION	93
7.1	INTRODUCING MAIL ENCRYPTION (PGP)	93
7.1.1	Using a key-pair to encrypt your mail	94
7.1.2	Sending encrypted mails to other people: you need their public key 94	
7.1.3	Receiving encrypted mails from other people: they need my public key	94
7.1.4	Conclusion: encryption requires public key distribution!	94
7.2	INSTALLING PGP ON WINDOWS	94
7.2.1	Installing PGP (GPG) on Microsoft Windows	95
7.2.2	Installing with the Enigmail extension	95
7.2.3	Installation steps	96
7.3	INSTALLING PGP ON OSX	98
7.3.1	Getting started	98
7.3.2	Downloading and installing the Software	98
7.3.3	Installing up Enigmail	107
7.4	Installing PGP on Ubuntu	110
7.5	Installing GPG on Android	111
7.5.1	APG	111
7.5.2	GPG enabled e-mail on Android: K-9 Mail	112
7.6	Creating your PGP keys	112
7.7	Daily PGP usage	122
7.7.1	Encrypting attachments	122
7.7.2	Entering your pass-phrase	123

7.7.3	Receiving encrypted e-mails	123
7.7.4	Sending and receiving public keys	124
7.7.5	Receiving public keys and adding them to your keyring	125
7.7.6	Using public key servers	128
7.7.7	Signing emails to an individual	134
7.7.8	Sending encrypted mails to an individual	135
7.7.9	Automating encryption to certain recipients	136
7.7.10	Verifying incoming e-mails	141
7.7.11	Revoking your GPG key-pair	142
7.7.12	What to do when you have lost your secret key, or forgot your passphrase	142
7.7.13	What to do when your secret key has been stolen, or compromised	142
7.7.14	Receiving a revocation certificate	143
7.7.15	Preparing for the worst: backup your keys	143
7.7.16	Further reading	146
7.8	Webmail and PGP	146
8	Safer Browsing	147
8.1	Why Firefox?	147
8.2	Accessing Firefox on Ubuntu	147
8.3	Installing on Mac OS X	148
8.4	Installing Firefox on Windows	153
8.4.1	Troubleshooting	157
8.5	Extending Firefox	157
8.5.1	HTTPS Everywhere	158
8.5.2	Installation	158
8.5.3	Configuration	160
8.5.4	Usage	160
8.5.5	If networks block HTTPS	163
8.5.6	Adding support for additional sites in HTTPS Everywhere	163
8.5.7	Enforcing secure HTTPS server connections	163
8.5.8	Adblock Plus	164
8.5.9	Getting started with Adblock Plus	164
8.5.10	Choosing a filter subscription	164
8.5.11	Creating personalized filters	166
8.5.12	Enabling and disabling AdBlock Plus for specific elements or Web sites	166
8.5.13	Other extensions that can improve your security	166
8.6	PROXY SETTINGS	167
8.6.1	Default Firefox proxy configuration	168
8.7	USING TOR?	170
8.7.1	Using Tor Browser Bundle	171
8.7.2	Downloading Tor Browser Bundle	171
8.7.3	Running a Relay or Bridge	171
8.8	EXTENDING GOOGLE CHROME	172
8.8.1	Disabling Instant Search	172
8.8.2	AdBlock for Chrome	172
8.8.3	HTTPS Everywhere	172

8.8.4	PrivacyFix	172
9	PASSWORDS	173
9.1	KEEPING PASSWORDS SAFE	173
9.1.1	Password length and complexity	173
9.1.2	Easy to remember and secure passwords	173
9.1.3	Minimizing damage	173
9.1.4	Using a password manager	174
9.1.5	Physical protection	174
9.1.6	Other caveats	174
9.2	INSTALLING KEEPPASS	174
9.2.1	Installing KeePassX on Ubuntu	174
9.2.2	Installing KeePass on Windows	175
9.2.3	Installing KeePass on Mac OS X	181
9.3	ENCRYPTING PASSWORDS WITH A PASSWORD MANAGER	189
9.3.1	Encrypting Passwords with KeePassX on Ubuntu	189
9.3.2	Encrypting Passwords with KeePass on Windows	195
9.3.3	Encrypting Passwords with Keychain on Mac OSX	201
10	USING VPN	205
10.1	GETTING, SETTING-UP AND TESTING A VPN ACCOUNT	205
10.1.1	An account from a commercial VPN provider	205
10.1.2	Setting up OpenVPN client	207
10.1.3	Caveats & Gotchas	208
10.2	VPN ON UBUNTU	208
10.2.1	Preparing Network Manager for VPN networks	208
10.2.2	Configuring an OpenVPN network	214
10.2.3	Using your new VPN connection	220
10.3	VPN ON MACOSX	222
10.3.1	Setup	222
10.4	VPN ON WINDOWS	235
	10.4.1 Setup	235
10.5	Making Sure Your VPN Works	248
11	Disk Encryption	249
11.1	Installing VeraCrypt	249
11.1.1	Installing on Ubuntu/Debian	249
11.1.2	Installing on OSX	252
11.1.3	Installing on Windows	256
11.2	Using VeraCrypt	256
11.2.1	Creating a VeraCrypt Container	256
11.2.2	Mounting the Encrypted Volume	262
11.2.3	What does this mean?	265
11.2.4	Remember to dismount!	265
11.3	Setting up a hidden volume	265
11.4	Securely destroying data	270
11.4.1	A note on Solid State Hard Drives	271

11.4.2	Securely delete data under Windows	271
11.4.3	Securely delete data under MacOSX	273
11.4.4	Securely delete data under Ubuntu/Linux	278
11.5	About LUKS	285
11.5.1	Starting Disks	285
11.5.2	Encrypting a device	287
11.5.3	Using an encrypted device	291
12	Call Encryption	293
12.1	Installing CSipSimple	293
12.1.1	Introducing The OSTN Network	293
12.1.2	CSipSimple	294
13	Instant Messaging Encryption	299
13.1	Setting up Encrypted Instant Messaging	299
13.1.1	Android - Installing Gibberbot	299
13.1.2	iOS - Installing ChatSecure	299
13.1.3	Ubuntu - Installing Pidgin	299
13.1.4	OS X - Installing Adium	300
13.1.5	Windows - Installing Pidgin	300
13.1.6	All OS - crypto.cat	300
13.1.7	Chat Log Files	301
14	Secure File Sharing	303
14.1	Installing I2P on Ubuntu Lucid Lynx (and newer) and derivatives like Linux Mint & Trisquel	303
14.2	Instructions for Debian Lenny and newer	306
14.3	Starting I2P	306
14.4	ANONYMOUS BITTORRENT WITH I2PSNARK	306
15	APPENDICES	309
15.1	CRYPTOGRAPHY AND ENCRYPTION	309
15.1.1	Encryption examples	310
15.1.2	A Warning!	310
15.1.3	Historical ciphers	311
15.1.4	Modern ciphers	313
15.1.5	Quantum Cryptography	314
15.1.6	Challenges & Implications	314
15.2	GLOSSARY	315
15.2.1	aggregator	315
15.2.2	anonymity	315
15.2.3	anonymous remailer	315
15.2.4	ASP (application service provider)	316
15.2.5	backbone	316
15.2.6	badware	316
15.2.7	bandwidth	316
15.2.8	bash (Bourne-again shell)	316
15.2.9	BitTorrent	316

15.2.10	blacklist	316
15.2.11	bluebar	317
15.2.12	block	317
15.2.13	bookmark	317
15.2.14	bridge	317
15.2.15	brute-force attack	317
15.2.16	cache	317
15.2.17	censor	317
15.2.18	censorware	317
15.2.19	CGI (Common Gateway Interface)	318
15.2.20	chat	318
15.2.21	cipher	318
15.2.22	circumvention	318
15.2.23	Common Gateway Interface	318
15.2.24	command-line interface	318
15.2.25	cookie	318
15.2.26	country code top-level domain (ccTLD)	318
15.2.27	cryptography	319
15.2.28	DARPA (Defense Advanced Projects Research Agency)	319
15.2.29	decryption	319
15.2.30	disk encryption	319
15.2.31	domain	319
15.2.32	DNS (Domain Name System)	319
15.2.33	DNS leak	320
15.2.34	DNS server	320
15.2.35	DNS tunnel	320
15.2.36	Eavesdropping	320
15.2.37	e-mail	320
15.2.38	embedded script	321
15.2.39	encryption	321
15.2.40	exit node	321
15.2.41	file sharing	321
15.2.42	file spreading engine	321
15.2.43	filter	321
15.2.44	Firefox	321
15.2.45	forum	321
15.2.46	frame	322
15.2.47	FTP (File Transfer Protocol)	322
15.2.48	full disk encryption	322
15.2.49	gateway	322
15.2.50	GNU Privacy Guard	322
15.2.51	GPG	322
15.2.52	honeypot	322
15.2.53	hop	322
15.2.54	HTTP (Hypertext Transfer Protocol)	323
15.2.55	HTTPS (Secure HTTP)	323
15.2.56	IANA (Internet Assigned Numbers Authority)	323

15.2.57	ICANN (Internet Corporation for Assigned Names and Numbers) (IM)	323	15.2.58 Instant Messaging
15.2.59	Intermediary	323	
15.2.60	Internet	323	
15.2.61	IP (Internet Protocol) Address	323	
15.2.62	IRC (Internet relay chat)	324	
15.2.63	ISP (Internet Service Provider)	324	
15.2.64	JavaScript	324	
15.2.65	KeePass, KeePassX	324	
15.2.66	keychain software	324	
15.2.67	keyword filter	324	
15.2.68	latency	324	
15.2.69	log file	324	
15.2.70	low-bandwidth filter	325	
15.2.71	malware	325	
15.2.72	man in the middle	325	
15.2.73	middleman node	325	
15.2.74	monitor	325	
15.2.75	network address translation (NAT)	325	
15.2.76	network operator	325	
15.2.77	node	326	
15.2.78	non-exit node	326	
15.2.79	obfuscation	326	
15.2.80	open node	326	
15.2.81	OTR/Off-the-Record messaging	326	
15.2.82	packet	326	
15.2.83	password manager	326	
15.2.84	pastebin	327	
15.2.85	peer-to-peer	327	
15.2.86	perfect forward secrecy	327	
15.2.87	Pretty Good Privacy (PGP)	327	
15.2.88	PHP	327	
15.2.89	plain text	327	
15.2.90	plaintext	327	
15.2.91	privacy	328	
15.2.92	private key	328	
15.2.93	POP3	328	
15.2.94	port	328	
15.2.95	protocol	328	
15.2.96	proxy server	328	
15.2.97	Psiphon node	328	
15.2.98	private node	329	
15.2.99	public key	329	
15.2.100	public key encryption/public-key cryptography	329	
15.2.101	publicly routable IP address	329	
15.2.102	regular expression	329	
15.2.103	remailer	329	
15.2.104	router	330	

15.2.105	root name server	330
15.2.106	RSS (Real Simple Syndication)	330
15.2.107	scheme	330
15.2.108	shell	330
15.2.109	SOCKS	330
15.2.110	screenlogger	330
15.2.111	script	331
15.2.112	smartphone	331
15.2.113	spam	331
15.2.114	SSH (Secure Shell)	331
15.2.115	SSL (Secure Sockets Layer)	331
15.2.116	steganography	331
15.2.117	subdomain	332
15.2.118	threat analysis	332
15.2.119	Top-Level Domain (TLD)	332
15.2.120	TLS (Transport Layer Security)	332
15.2.121	TCP/IP (Transmission Control Protocol over Internet Protocol)	332
15.2.122	Tor bridge	332
15.2.123	traffic analysis	332
15.2.124	tunnel	333
15.2.125	UDP (User Datagram Packet)	333
15.2.126	URL (Uniform Resource Locator)	333
15.2.127	Usenet	333
15.2.128	VoIP (Voice over Internet Protocol)	333
15.2.129	VPN (virtual private network)	333
15.2.130	whitelist	334
15.2.131	World Wide Web (WWW)	334
15.2.132	Webmail	334
15.2.133	Web proxy	334
15.2.134	WHOIS	334
15.3	The necessity of Open Source	335

1 Introducing Cryptoparty

1.1 About This Book

The CryptoParty Handbook was born from a suggestion by Marta Peirano (<http://petitemedia.es>) and Adam Hyde (<http://booksprints.net>) after the first Berlin CryptoParty, held on the 29th of August, 2012. Julian Oliver (<http://julianoliver.com>) and Danja Vasiliev (<http://k0a1a.net>), co-organisers of the Berlin CryptoParty along with Marta were very enthusiastic about the idea, seeing a need for a practical working book with a low entry-barrier to use in subsequent parties. Asher Wolf, originator of the CryptoParty movement, was then invited to run along and the project was born.

This book was written in the first 3 days of October 2012 at Studio Weise7, Berlin, surrounded by fine food and a small ocean of coffee. Approximately 20 people were involved in its creation, some more than others, some local and some far.

The writing methodology used, BookSprint (<http://booksprints.net>), is all about minimising any obstruction between expertise and the published page. Face-to-face discussion and dynamic task-assignment were a huge part of getting the job done, like any good CryptoParty!

The open source, web-based (HTML5 and CSS) writing platform BookType (<http://booktype.pro>) was chosen for the editing task, helping such a tentacular feat of parallel development to happen with relative ease. Asher also opened a couple of TitanPad pages to crowd-source the Manifesto and HowTo CryptoParty chapters.

Combined, this became the official CryptoParty Handbook by midnight October the 3rd, GMT+1.

The Book Sprint was 3 days in length and the full list of onsite participants included:

- Adam Hyde (facilitator)
 - Marta Peirano
 - Julian Oliver
 - Danja Vasiliev
 - Asher Wolf (<http://cryptoparty.org>)
 - Jan Gerber
 - Malte Dik
 - Brian Newbold
 - Brendan Howell (<http://wintermute.org>)
 - AT
 - Carola Hesse
- Chris Pinchen (<http://chokepointproject.net>).
 - Cover art by Emile Denichaud (<http://about.me/denichaud>)

This version of the handbook has since moved to github to collaboratively edit it. Find it at <https://github.com/cryptoparty/handbook>. If you see areas that need improvement or simply come across a typo, create a github account and start editing, commenting or creating issues. For help using git and github, see <https://help.github.com/>.

CryptoParty HandBook Credits

Facilitated by:

- Adam Hyde Core

Team:

- Marta Peirano
- Asher Wolf
- Julian Oliver
- Danja Vasiliev
- Malte Dik
- Jan Gerber
- Brian Newbold
- Brendan Howell

Assisted by:

- Teresa Dillon
- AT
- Carola Hesse
- Chris Pinchen
- ‘LiamO’
- ‘13lackEyedAngels’
- ‘Story89’
- Travis Tueffel

Github migration, packaging and maintenance by:

- Yuval Adam
- Samuel Carlisle
- Daniel Kinsman
- pettter
- Jens Kubieziel
- Uwe Lippmann
- Kai Engert

1.2 A CryptoParty Manifesto

Cover Image by Emile Denichaud. Other material included:

- <https://www.flossmanuals.net/bypassing-censorship>

The manuals used in the second half of this book borrow from 2 books sprinted by FLOSS Manuals:

- “How to Bypass Internet Censorship” 2008 & 2010 Adam Hyde (Facilitator), Alice Miller, Edward Cherlin, Freerk Ohling, Janet Swisher, Niels Elgaard Larsen, Sam Tennyson, Seth Schoen, Tomas Krag, Tom Boyle, Nart Villeneuve, Ronald Deibert, Zorrino Zorrinno, Austin Martin, Ben Weissmann, Ariel Viera, Niels Elgaard Larsen, Steven Murdoch, Ross Anderson, helen varley jamieson, Roberto Rastapopoulos, Karen Reilly, Erinn Clark, Samuel L. Tennyson, A Ravi
- “Basic Internet Security” 2011 Adam Hyde (Facilitator), Jan Gerber, Dan Hassan, Erik Stein, Sacha van Geffen, Mart van Santen, Lonneke van der Velden, Emile den Tex and Douwe Schmidt

All content in the CryptoParty Handbook is licensed under the [Creative Commons Attribution-ShareAlike 3.0 Unported \(CC BY-SA 3.0\)](https://creativecommons.org/licenses/by-sa/3.0/).

All chapters list the contributors unless otherwise noted below.

1.2 A CryptoParty Manifesto

“Man is least himself when he talks in his own person. Give him a mask, and he will tell you the truth.” - Oscar Wilde

In 1996, John Perry Barlow, co-founder of the [Electronic Frontier Foundation \(EFF\)](#), wrote ‘A Declaration of the Independence of Cyberspace’. It includes the following passage:

Cyberspace consists of transactions, relationships, and thought itself, arrayed like a standing wave in the web of our communications. Ours is a world that is both everywhere and nowhere, but it is not where bodies live.

We are creating a world that all may enter without privilege or prejudice accorded by race, economic power, military force, or station of birth.

We are creating a world where anyone, anywhere may express his or her beliefs, no matter how singular, without fear of being coerced into silence or conformity.

Sixteen years later, and the Internet has changed the way we live our lives. It has given us the combined knowledge of humankind at our fingertips. We can form new relationships and share our thoughts and lives with friends worldwide. We can organise, communicate and collaborate in ways never thought possible. This is the world we want to hand down to our children, a world with a free Internet.

Unfortunately, not all of John Perry Barlow’s vision has come to pass. Without access to online anonymity, we can not be free from privilege or prejudice. Without privacy, free expression is not possible.

The problems we face in the 21st Century require all of humanity to work together. The issues we face are serious: climate change, energy crises, state censorship, mass surveillance and on-going wars. We must be free to communicate and associate without fear. We need to support free and open source projects which aim to increase the commons’ knowledge of technologies that we depend on <http://opensourceecology.org/wiki> Contribute!

To realise our right to privacy and anonymity online, we need peer-reviewed, crowd-sourced solutions. CryptoParties provide the opportunity to meet up and learn how to use these solutions to give us all the means with which to assert our right to privacy and anonymity online.

1. We are all users, we fight for the user and we strive to empower the user. We assert user requests are why computers exist. We trust in the collective wisdom of human beings, not software vendors, corporations or governments. We refuse the shackles of digital gulags, lorded over by vassal interests of governments and corporations. We are the CypherPunk Revolutionaries.
2. The right to personal anonymity, pseudonymity and privacy is a basic human right. These rights include life, liberty, dignity, security, right to a family, and the right to live without fear or intimidation. No government, organisation or individual should prevent people from accessing the technology which underscores these basic human rights.
3. Privacy is the right of the individual. Transparency is a requirement of governments and corporations who act in the name of the people.
4. The individual alone owns the right to their identity. Only the individual may choose what they share. Coercive attempts to gain access to personal information without explicit consent is a breach of human rights.

5. All people are entitled to cryptography and the human rights crypto tools afford, regardless of race, colour, sex, language, religion, political or other opinion, national or social origin, property, birth, political, jurisdictional or international status of the country or territory in which a person resides.
6. Just as governments should exist only to serve their citizens - so too, cryptography should belong to the people. Technology should not be locked away from the people.
7. Surveillance cannot be separated from censorship, and the slavery it entails. No machine shall be held in servitude to surveillance and censorship. Crypto is a key to our collective freedom.
8. Code is speech: code is human created language. To ban, censor or lock cryptography away from the people is to deprive human beings from a human right, the freedom of speech.
9. Those who would seek to stop the spread of cryptography are akin to the 15th century clergy seeking to ban the printing press, afraid their monopoly on knowledge will be undermined.

1.3 How To CryptoParty

- Throw a party. All you need is a time, a date and a location. Add it to the wiki: <https://cryptoparty.org>.
- Make sure you have Internet connectivity and enough power sources for all devices. If you do not have a place to hold a CryptoParty, find a pub or park where you can meet and squeeze the public bandwidth. That will really hone your skills!
- Bring USB sticks and printed handouts for those who need them, and set up old computers for people to fiddle with and try out new skills.
- Talk about Linux to everyone you meet at your CryptoParty. If you are new to CryptoParties - ask someone "what is Linux?" ASAP.
- Make entry free for all if possible - CryptoParties are not-for-profit, not commercially aligned and especially important for those without other resources.
- Teach basic cryptographic tools to the masses. Crowd-source the best crypto. We suggest PGP, OTR, and Tor as the first tools to install.
- Invite experts and non-experts from all fields. Everyone is an expert on something.
- If you want CryptoParty to do something, start doing it. Organise organically and chaotically. Have no clear leadership. Urge people to take on a sudo leadership role - take a tutorial, fix the wifi, update the wiki, or organise the next CryptoParty. If someone claims others are doing it wrong - invite them to nominate themselves to do it better.
- Ask for feedback. Assimilate critics - ask them for their help in creating a better CryptoParty. Do not be scared to troll the trolls back or boot them from your space. Share feedback on the wiki. Iterate.
- A successful CryptoParty can have as many or as few as two people. Size doesn't count, it's what you do with it that matters. The criterion for success should be that everyone had

fun, learned something and wants to come to the next party.

- Think of the CryptoParty movement as a huge Twitter hive ready to swarm at any moment. Tweet a lot, and make your tweets meaningful. ReTweet other CryptoPartiers frequently.
 - Make sure the way crypto is taught at your party could be understood by a 10 year old. Then have the 10 year old teach it to an 80 year old. Breach the digital divide with random acts of awesomeness such as unfettered use of images of kittens in all CryptoParty literature. Red underpants on heads is only mandatory if you wish to bid in our spectrum auction.
 - Consider hosting private, off-the-radar CryptoParties for activists, journalists and in individuals working in dangerous locations.
 - Don't scare non-technical people. Don't teach command lines before people know where the on-off buttons are located on their laptops. Everyone learns at their own pace - make sure there is support for those in need of help.
 - Doing excellent stuff at CryptoParty does not require permission or an official consensus decision. If you're uncertain about the excellence of something you want to do, you should ask someone else what they think.
 - Consider the need for a bouncer, particularly if your CryptoParty expects over 50 people. Dress the bouncer up as a Sumo wrestler. Do not be afraid to bounce people who breach CryptoParty's anti-harassment policy.
 - CryptoParty is dedicated to providing a harassment-free sharing experience for everyone, regardless of gender, sexual orientation, disability, physical appearance, body size, heritage, or religion. Behaving like an arsehole may mean you are permanently uninvited to CryptoParties events. Harassment includes:
 - hurtful or offensive comments
 - deliberate intimidation
 - direct or indirect threats
 - stalking
 - following
 - inappropriate physical contact
 - unwelcome sexual attention.
 - Encourage a culture of sharing. Encourage advanced users to help not-so advanced ones. Delegate.
 - Use online meeting platforms like mumble, or even chatrooms (e.g. #cryptoparty room on <http://occupytalk.org/>) when physical meetups are not possible or impractical.
 - Copy from other cryptoparties. Remix, Reuse and Share. Create a basket of old devices people are willing to donate to more needy CryptoPartiers.
- Get the word out! Print posters and/or flyers and distribute them in your neighbourhood, post online versions to social networks and mail them to friends, for them to distribute the info even further.
 - Don't sell out to sponsors for pizza and beer money. Ask people to try and bring food and drink to share. Host CryptoPicnics as often as possible. Make friends with librarians. They wield power

over keys to local, public meeting rooms that may be free of charge to utilize.

- Invite all the people. Bring people together who have a wide range of skills and interests - musicians, political pundits, activists, hackers, programmers, journalists, artists and philosophers. Spread the love.
- Invite the graphic designers and illustrators you know to contribute new ways to help people understand crypto.
- Invite everyone to share their knowledge and their skills. Individuals with little or no coding, programming, hacking or crypto skills can change cultures by promoting the idea that privacy is a fundamental right.
- Share music, beers, & chips. Bond together over eclectic music, cheeseballs, installing GPG, TrueCrypt, OTR and Tor, as well as watching movies together. We recommend Hackers, The Matrix, Bladerunner, Tron, Wargames, Sneakers, and The Net.
- Do not work too hard. Take breaks. Eat popcorn together. Create slang, phrases, memes.
- When people at CryptoParties ask for advice on “hacking the Gibson” refer them to episodes of ‘My Little Pony’.
- Create fliers and advertise using slogans like: “CryptoParties: If there is hope, it lies in the proles” and “CryptoParty like it’s 1984.” CryptoParty all the things to avoid oppression and depression.
- Seed CryptoParties in your local communities - at nursing homes, scout groups, music festivals, universities, schools. Take CryptoParty to isolated and remote communities. Make friends in far away places and travel whenever possible. Ask people in rural farming communities if they’d like to CryptoParty.
- Share shimmering opportunities of crowd-sourced privacy: swap cheap, pre-paid SIMs, handsets and travel cards.
- Create logos in bright pink and purple, with hearts all over them. Promote CryptoParties to rebellious 13 year old girls. Declare success if rebellious 13 year old girls demand to attend your parties.
- Become friends with journalists. Invite them to your parties. Teach them crypto. Do not scare them by discussing Assassination Markets.
 - Strew CryptoParty sigils across your city in 3am post-party raids. Make lots of stickers, paste them everywhere.
 - Experiment, constantly. Do not be afraid to make mistakes. Encourage people to tinker. Assume all mistakes are meant to be made. Most people under intel agency scrutiny have electronic devices already compromised before they walk in the door. Teach people to install tools from scratch, so they can do it on a new machine, away from prying eyes.
 - Assume intel agencies send representatives to CryptoParties. Acknowledge their presence at the start of your meeting, ask them to share their crypto skills. Joke about paranoia as often as possible without instilling panic. Wear tinfoil hats.
 - Be excellent to each other and CryptoParty on.

1.4 Party like it's december 31st 1983

1.4.1 What is CryptoParty?

CryptoParty is a decentralized, global initiative to introduce basic tools for protecting privacy, anonymity and overall security on the Internet to the general public.

The idea was conceived in the wake of the [Australian Cybercrime Legislation Amendment Bill 2011](#) and the reasoning is that laws like this are without substance when everybody encrypts their communication.

CryptoParties are neither commercially nor politically aligned, and free and open to attend for everyone as long as they live the following *guiding principles*:

Be excellent to each other

CryptoParties are environments where people feel welcome and safe to learn and teach no matter their background or level of expertise. All questions are relevant, all explanation shall be targeted at the person with least pre-knowledge.

This also means that any form of harassment or other behaviour that makes people uncomfortable has no place at CryptoParties. In our experience situations like these (as seldom as they occur) stem rather from social ineptitude than malice and can thus be resolved by making people aware of their behaviour and its effect on others, but in last consequence it is on the organizers of the CryptoParty to ask people to leave if they don't adhere to this very simple rule. Be excellent to each other. Awareness is key in this regard.

Do things

CryptoParties happen because people make them happen. The most amazing and unforeseen learning experiences happen because people make them happen. If you are uncertain if what you have in mind is on topic or if other people are interested as well:

1.5 Preface

Propose it anyway and see what other people have to say. If you are too shy to propose to the whole room: Ask the person standing next to you first.

On a more global scale, there is a mailing list global@cryptoparty.is which is open for questions and discussion of all kind, as well as country and city-specific mailing lists and other resources which can be found on <https://cryptoparty.in>.

For a guide on how to organize CryptoParties please be referred to the chapter of the same name.

1.5 Preface

This book is a collective and ongoing effort in that it is based on the two [FLOSS Manuals How to Bypass Internet Censorship](#) and [Basic Internet Security](#) and collaboratively edited on [Github](#) although other venues of collaborative editing are investigated.

Its goal is to give a comprehensive resource to people who would like to attend or organize a CryptoParty but lack the local expertise or just confidence in doing so. All chapters are designed to be self-contained.

All content in the *CryptoParty Handbook* is licensed under the [Creative Commons Attribution-ShareAlike 3.0 Unported \(CC BY-SA 3.0\)](#). The authors are listed in *Appendix A: Contributions*.

1.6 Why Privacy Matters

Privacy is a fundamental human right. It is recognized in many countries to be as central to individual human dignity and social values as Freedom of Association and Freedom of Speech. Simply put, privacy is the border where we draw a line between how far a society can intrude into our personal lives.

Countries differ in how they define privacy. In the UK for example, privacy laws can be traced back to the 1300s when the English monarchy created laws protecting people from eavesdroppers and peeping toms. These regulations referred to the intrusion of a person's comfort and not even the King of England could enter into a poor person's house without their permission. From this perspective, privacy is defined in terms of personal space and private property. In 1880 American lawyers, Samuel Warren and Louis Brandeis described privacy as the 'right to be left alone'. In this case, privacy is synonymous with notions of solitude and the right for a private life. In 1948, the Universal Declaration of Human Rights specifically protected territorial and communications privacy which by that became part of constitutions worldwide. The European Commission on Human Rights and the European Court of Human Rights also noted in 1978 that privacy encompasses the right to establish relationships with others and develop emotional well-being.

Today, a further facet of privacy increasingly perceived is the personal data we provide to organizations, online as well as offline. How our personal data is used and accessed drives the debate about the laws that govern our behavior and society. This in turn has knock-on effects on the public services we access and how businesses interact with us. It

even has effects on how we define ourselves. If privacy is about the borders which govern who we give permission to watch us and track aspects of our lives, then the amount and type of personal information gathered, disseminated and processed is paramount to our basic civil liberties.

An often heard argument, when questions of privacy and anonymity come up, goes along the lines of, "I only do boring stuff. Nobody will be interested in it anyway" or, "I have nothing to hide". Both of these statements are easily defeated.

Firstly, a lot of companies are very interested in what boring things you do precisely so they have opportunity to offer "excellent" products fitting interests. In this way their advertising becomes much more efficient - they are able to tailor specifically to assumed needs and desires. Secondly you do have lots to hide. Maybe you do not express it in explicitly stated messages to friends and colleagues, but your browsing - if not protected by the techniques laid out in this book - will tell a lot about things you might rather keep secret: the ex-partner you search for using Google, illnesses you research or movies you watch are just few examples.

Another consideration is that just because you might not have something to hide at this moment, you may very well in future. Putting together all the tools and skills to protect yourself from surveillance takes practice, trust and a bit of effort. These are things you might not be able to achieve and configure right when you need them most and need not take the form of a spy movie. An obsessed, persistent stalker, for example, is enough to heavily disrupt your life. The more you follow the suggestions given in this book, the less impact attacks like this will have on you. Companies may also stalk you too, finding more and more ways to reach into your daily life as the reach of computer networking itself deepens.

Finally, a lack of anonymity and privacy does not just affect you, but all the people around you. If a third party, like your Internet Service Provider, reads your email, it is also violating the privacy of all the people in your address book. This problem starts to look even more dramatic when you look at the issues of social networking websites like Facebook. It is increasingly common to see photos uploaded and tagged without the knowledge or permission of the people affected.

While we encourage you to be active politically to maintain your right to privacy, we wrote this book in order to empower people who feel that maintaining privacy on the Internet is also a personal responsibility. We hope these chapters will help you reach a point where you can feel that you have some control over how much other people know about you. Each of us has the right to a private life, a right to explore, browse and communicate with others as one wishes, without living in fear of prying eyes.

2 Understanding Email

2.1 *Basic Tips*

Just as with other forms of communication on the web, some basic precautions always ought to be taken to ensure you have the best chance at protecting your privacy.

2.1.1 *In brief:*

- Passwords shouldn't relate to personal details and should contain a mix of a reasonable amount of letters and other characters.
- To change passwords regularly is important as they might have been stolen, cracked or exposed to others in the meantime.
- Always be sure your connection is secure when reading email on a wireless network, especially in Internet cafes.
- Temporary files (the 'cache') on the computer that you use to check your email can present some risks. Clear them often.
- Create and maintain separate email accounts for different tasks and interests.
- Encrypt any message you wouldn't feel comfortable sending on a post card.
- Be aware of the risks of having your email hosted by a company or organization.

2.1.2 *Passwords*

Passwords are a primary point of vulnerability in email communication. Even a secure password can be read in transit unless the connection is secure (see TLS/SSL in the glossary). In addition, just because a password is long doesn't mean it cannot be guessed by using knowledge of you and your life to determine likely words and numbers.

The general rule for creating passwords is that it should be long (8 characters might be cracked within few hours) and have a mix of letters and other characters (numbers and symbols, which means you could just choose a sentence). Combining your birthday with that of a family name is however a great example of how not to do it. This kind of information is easy to find using public resources. A popular trick is to base it on a favourite phrase and then, just to throw people off, sprinkle it with a few numbers. Best of all is to use a password generator, either on your local system or online.

Often such passwords are difficult to remember and a second point of vulnerability is opened up – physical discovery. Since there is no better means of storing a password than in your own brain, services like OnlinePasswordGenerator (<http://www.onlinepasswordgenerator.com/>) offer a

great compromise by randomly generating passwords that vaguely resemble words and present you with a list to choose from.

If you do choose to store your password outside your head, you have the choice to either write it down or use keychain software. This can be a risky decision, especially if the email account and password are on the same device like your phone or computer.

Keychain software, like KeePass, consolidates various passwords and passphrases in one place and makes them accessible through a master password or passphrase. This puts a lot of pressure on the master password. If you do decide to use a keychain software, remember to choose a secure password.

Finally, you should use a different password for different accounts. In that way, if one of them gets hijacked, your other accounts remain safe. Never use the same password for your work and private email accounts. See section **Passwords** to learn more about how to secure yourself.

2.1.3 Reading Email in Public Places

One of the great conveniences of wireless networking and ‘cloud computing’ is the ability to work anywhere. You may often want to check your email in an Internet cafe, on open networks or public location. Spies, criminals and mischievous types are known to visit these locations in order to take advantage of the rich opportunities offered for ID theft, email snooping and hijacking bank accounts.

Here we find ourselves within an often underestimated risk of someone listening in on your communications using *network packet sniffing*. It matters little if the network itself is open or password secured. If someone joins the same encrypted network, s/he can easily capture and read all *unsecured* (see chapter **Secure Connection** for *TLS* and *VPN* solutions) traffic of all of other users within the same network. A wireless key can be acquired for the cost of a cup of coffee and gives those that know how to capture and read network packets the chance to read your password while you check your email if the connection to that service is not secured.

Here a simple general rule always applies: if the cafe offers a network cable connection, use it! Finally, just as at a bank machine, make sure no one watches over your shoulder when you type in the password.

2.1.4 Cache Cunning

Here again convenience quickly paves the road to bad places. Due to the general annoyance of having to type in your password over and over again, you ask the browser or local mail client to store it for you. This is not bad in itself, but when a notebook or phone gets stolen, it enables the thief to access the owner’s email account(s). The best practice is to clear this cache every time you close your browser. All popular browsers have an option to clear this cache on exit.

One basic precaution can justify you holding onto your convenient cache: disk encryption. If your notebook is stolen and the thief reboots the machine, they’ll be met with an encrypted disk. It is also wise to have a screen lock installed on your computer or phone. If the machine is taken from you while still running your existing browsing session, it cannot be accessed.

2.1 Basic Tips

2.1.5 Securing your communication

Whenever you write and send email in a browser or use an email program (Outlook Express, Mozilla Thunderbird, Mail.app or Mutt), you should always ensure to use encryption for the entire session. This is easily done due to the popular use of *TLS/SSL (Secure Socket Layer)* connections by email servers (See glossary **TLS/SSL**).

If using a browser to check your email, check to see if the mail server supports SSL sessions by looking for `https://` at the beginning of the URL. If not, be sure to turn it on in your email account settings, such as Gmail or Hotmail. This ensures that not just the login part of your email session is encrypted but also the writing and sending of emails. Furthermore check the certificate details and take *TLS pinning* into account and endorse browser extensions that warn about changing or disfunctional certificates (e.g. *Certificate Patrol*) and make use of TLS secured version of the website the default (e.g. *HTTPS everywhere*).

The email service provider you select, should provide you with the mail server details. These details can often be found in the settings option. If your email service provider does not offer you a cryptographic protocol (TLS/SSL) to encrypt your data on the network, then it is advised to stop using it. Even if your emails are not important, you might find yourself 'locked out' of your account one day with a changed password!

When using an email program to check your email, be sure that you are using TLS/SSL in the program options. For instance in Mozilla Thunderbird the option for securing your outgoing email is found in Tools -> Account Settings -> Outgoing Server (SMTP) and for incoming email in Tools -> Account Settings -> Server Settings. This ensures that the downloading and sending of email is encrypted, making it very difficult for someone on your network, or on any of the networks between you and the server, to read or log your email. Encrypting the email itself

Even if the line itself is encrypted using a system such as SSL, the email service provider still has full access to the email because they own and have full access to the storage device where you host your email. If you want to use a web service and be sure that your provider cannot read your messages, then you'll need to use something like *GPG* (Appendix for **GnuPG**) with which you can encrypt the email. The header of the email however will still contain the IP (Internet address) that the email was sent from alongside other compromising details. Worth mentioning here is that the use of *GPG* in webmail is not as comfortable as with a locally installed mail client, such as *Thunderbird* or *Outlook Express*.

2.1.6 DNSSEC & DANE

certificate information can be stored in DNS records and therefore be regarded more reliable. Check the availability of *DNSSEC* and especially regarding email services *DANE* with your service providers. Here again browser extensions (e.g. *DNSSEC/TLSA Validator*) can assist to control the availability of these security measures.

2.1.7 Account Separation

Due to the convenience of services like Gmail, it is increasingly typical for people to have only one email account. This considerably centralises the potential damage done by a compromised account.

More so, there is nothing to stop a disgruntled Google employee from deleting or stealing your email, let alone Google itself getting hacked. Hacks happen. A practical strategy is to keep your personal email, well, personal. If you have a work email then create a new account if your employers haven't already done it for you. The same should go for any clubs or organisations

you belong to, each with a unique password. Not only does this improve security, by reducing the risk of whole identity theft, but greatly reduces the likelihood of spam dominating your daily email.

2.1.8 A note about hosted email

Those that provide you with the service to host, send, download and read email are not encumbered by the use of TLS/SSL. As hosts, they can read and log your email in plain text. They can comply with requests by local law enforcement agencies who wish to access email. They may also study your email for patterns, keywords or signs of sentiment for or against brands, ideologies or political groups. It is important to read the EULA (End-user license agreement) of your email service provider and do some background research on their affiliations and interests before choosing what kind of email content they have access to. These concerns also apply to the hosts of your messages' recipients.

2.2 Types of Email

The use of email almost always comes in two forms:

- Email read, written and sent in the *browser* (webmail via HTTP) and stored on a providers server, and/or
- Email read, written, sent and stored using an *email program*, like e.g. Mozilla Thunderbird, Mail.App or Outlook Express by utilizing protocols like *SMTP*, *POP* and *IMAP*.

These two models might be mixed in practice, especially by using *IMAP*. Whilst the webmail solution is more convenient to use and easier to maintain for end users on different computers compared to the more powerful solution (less limits on storage, better search options and direct control over data) based on native applications.

2.2.1 Remotely hosted email ('webmail'), resourced using a web browser

Email sent using the *browser*, sometimes referred to as *webmail*, typically assumes an account with a remote email host like Google (Gmail), Microsoft (Hotmail) or Yahoo (Yahoo Mail). The business opportunities opened up by hosting other people's email

2.2 Types of Email

are many: contact with other services offered by the company, brand exposure and most importantly, mining your plain text email for patterns that can be used to evaluate your interests – something of great value to the advertising industry (alongside certain Governments). For the reason of datamining those companies have *no interest* in encouraging their users to use *encryption to secure privacy* and/or *signatures for integrity/authenticity* of communication.

2.2.2 Remotely hosted email, resourced using an email program or using a web browser

Email sent using an email program like Outlook, Thunderbird, Mail.App aso. can also be used with a webmail service like Gmail or your company's email service. In either case, email may still be downloaded onto your computer but is retained on the email server (e.g. Gmail). Done

this way, accessing email doesn't require the browser at all, but you are still using Gmail, Hotmail as a service. The difference between storing email on your computer with an email program and having it stored remotely on an email server (like Hotmail, Gmail or your University's service) on the Internet can appear confusing at first.

Finally, email can also be sent to an email server but not stored there at all, merely volleyed onto its' destination as soon as the email reaches the email forwarding server. Google and Microsoft do not allow for this sort of setup. Rather this is typically some- thing your university or company will provide for you. Bear in mind that this comes with the risk of the email administrator on that system still secretly copying the email as it reaches and leaves the server.

Generally, using webmail alongside downloading it using an email program is the best approach. This approach adds redundancy (local backups) alongside the option to delete all email from the remote server once downloaded. The latter option is ideal for content sensitive information where the possibility of account hijacking is high but risks total loss of email should the local machine go missing, without backups. Secondly, when using an email program, we have the option of using Email Encryption such as the popular OpenPGP implementation **GPG**, something not easily set up and used with browser-only webmail services. In any case, disk encryption on the local machine is highly advisable (Appendix **Disk Encryption**).

2.2.3 Context considerations

You may be a server administrator yourself and run your own email service. Or your email could be stored on your company or bosses' server. Finally you may be using a service provided by a corporation, like Google (Gmail) or Microsoft (Hotmail). Each comes with its own interesting mix of considerations that relates precisely to the basic fact that unless the email itself is encrypted, the administrator of the email server can still secretly copy the email the moment it reaches the server. It doesn't matter that you may be using *TLS/SSL* (Appendix **SSL**) to login and check your email as this only protects the connection between your local machine and the server.

As always, if you know the risks and feel concerned it is wise to listen to them - don't send sensitive email using a service you don't trust. Employer/Organisation

Your employer or an organisation that you are involved with is in a very good position to take advantage of your trust and read the emails of your business email account that is stored on their email server, perhaps in an effort to learn about you, your motivations, agendas and interests. Such cases of employer->employee spying are so typical they do not bear mention. Your only measure against it is to use an email encryption solution like GPG (Appendix GPG).

2.2.4 Email & Metadata

The actual content information of mails might be preserved utilizing *OpenPGP* or *S/MIME* but the metadata - the association of persons, addresses, time and used software/services - is stored by several stakeholders. Government services might store such data as well as any company involved in transmitting them. In regards of header information Email remains a risk for communication as long as the accounts used can be connected to individuals or groups.

2.2.5 Self-administered email server

Generally speaking this is the ideal hosting configuration, but requires a higher level of technical skill. Here, in general, the risks to privacy are not only in protecting your own email against

attempts at exploit (poor passwords, no SSL) but in that you have a responsibility, and perhaps a temptation, to read the emails of those you provide a service for.

2.2.6 'Free' email services

As mentioned above the risks of storing and sending your email using a service provided by a corporation are rather high if respect of your civil right to privacy is valued. The companies hosting your love letters, random expressions and diaries are always at risk of yielding to pressures from political, economic and law enforcement interests of the country to which they are legally subject. A Malaysian Gmail user, for instance, risks exposing her interests and intents to a government she did not elect, not to mention business partners of Google interested in expanding their market reach.

2.2.7 Non-profit

Several non-profit web hosts offer free email accounts to organisations that are themselves non-profit or philanthropic. Some of them even offer wikis, mailing lists, chats and social networks. A consideration for organisations working in a political field may be differences of interests between the state in which the email is hosted and the political interests of the organisation using that service. Such risks would ideally be reflected in the End User License Agreement.

2.2.8 Notes on email forwarding

Email forwarding services provide the great convenience of 'linking' one email account to another as the user sees fit. This of course is most commonly used when an account holder is on holiday and would like email forwarded from their work account to another used during travel or otherwise inaccessible outside the workplace. The risk with any external email forwarding service is the same as with remotely hosted emails through Gmail for instance: it can be copied and stored. Here email encryption using a system such as *GPG* (Appendix **GPG**) will ensure that if it is copied at least it cannot be read.

2.3 Fears

Who can read the email messages that I have already sent or received? Who can read the emails I send when they travel across the Internet? Can the people I send emails to share them with anybody?

Emails that are sent "in the clear" without any encryption (which means the vast majority of email sent and received today) can be read, logged, and indexed by any server or router along the path the message travels from sender to receiver. Assuming you use an encrypted connection (see glossary for TLS/SSL) between your devices and your email service provider (which everybody should), this means in practice that the following people can still read any given message:

1. You
2. Your email service provider
3. The operators and owners of any intermediate network connections (often ambiguous multinational conglomerates or even sovereign states)
4. The recipient's email service provider
5. The intended recipient

Many webmail providers (like Gmail) automatically inspect all of the messages sent and received by their users for the purpose of showing targeted advertisements. While this may be a reasonable compromise for some users most of the time (free email!), it is disturbing for many that even their most private communications are inspected and indexed as part of a hidden and potentially very insightful profile maintained by a powerful corporate giant with a profit motive.

Additionally, somebody who can legally pressure the groups above could request or demand:

1. logged meta-data about email (lists of messages sent or received by any user, subject lines, recipients), in some jurisdictions even without a warrant.
2. messages sent and received by a specific user or group, with a warrant or court order in some jurisdictions.
3. a dedicated connection to siphon off *all* messages and traffic, to be analyzed and indexed off site.

In cases where a user has a business or service relationship with their email provider, most governments will defend the privacy rights of the user against unauthorized and unwarranted reading or sharing of messages, though often it is the government itself seeking information, and frequently users agree to waive some of these rights as part of their service agreement. However, when the email provider is the user's employer or academic institution, privacy rights frequently do not apply. Depending on jurisdiction, businesses generally have the legal right to read all of the messages sent and received by their employees, even personal messages sent after hours or on vacation.

Historically, it was possible to “get away” with using clear text email because the cost and effort to store and index the growing volume of messages was too high: it was hard enough just to get messages delivered reliably. This is why many email systems do not contain mechanisms to preserve the privacy of their contents. Now the cost of monitoring has dropped much faster than the growth of internet traffic and large-scale monitoring and indexing of all messages (either on the sender or receiving side) is reasonable to expect even for the most innocuous messages and users. [CITE: corporate email archiving/spying, blue coat, Syrian monitoring, USA Utah data center, USA intercept scandals]

For more about legal protections of email messages “at rest” (technical term for messages stored on a server after having been delivered), especially regarding government access to your email messages, see:

- <https://ssd.eff.org/3rdparties/govt/stronger-protection> (USA)
- http://en.wikipedia.org/wiki/Data_Protection_Directive (EU)

Just like there are certain photos, letters, and credentials that you would not post “in the clear” on the Internet because you would not want that information to get indexed accidentally and show up in search results, you should never send email messages in the clear that you would not want an employer or disgruntled airport security officer to have easy access to.

2.3.1 Random abuse and theft by malicious hackers

What if somebody gets complete control of my email account?

*I logged in from an insecure location. . . how do I know now if my account has been hacked?
I've done nothing wrong. . . what do I have to hide? Why
would anybody care about me?*

Unfortunately, there are many practical, social, and economic incentives for malicious hackers to break into the accounts of random Internet individuals. The most obvious incentive

is identity and financial theft, when the attacker may be trying to get access to credit card numbers, shopping site credentials, or banking information to steal money. A hacker has no way to know ahead of time which users might be better targets than others, so they just try to break into all accounts, even if the user doesn't have anything to take or is careful not to expose his information.

Less obvious are attacks to gain access to valid and trusted user accounts to collect contact email addresses from and then distribute mass spam, or to gain access to particular services tied to an email account, or to use as a "stepping stone" in sophisticated social engineering attacks. For example, once in control of your account a hacker could rapidly send emails to your associates or co-workers requesting emergency access to more secured computer systems.

A final unexpected problem affecting even low-profile email users, is the mass hijacking of accounts on large service providers, when hackers gain access to the hosting infrastructure itself and extract passwords and private information in large chunks, then sell or publish lists of login information in online markets.

2.3.2 Targeted abuse, harassment, and spying

Something I wrote infuriated a person in power. . . how do I protect myself?

If you find yourself the individual target of attention from powerful organizations, governments, or determined individuals, then the same techniques and principles will apply to keeping your email safe and private, but additional care must be taken to protect against hackers who might use sophisticated techniques to undermine your devices and accounts. If a hacker gains control of any of your computing devices or gets access to any of your email accounts, they will likely gain immediate access both to all of your correspondence, and to any external services linked to your email account.

Efforts to protect against such attacks can quickly escalate into a battle of wills and resources, but a few basic guidelines can go a long way. Use specific devices for specific communication tasks, and use them only for those tasks. Log out and shutdown your devices immediately when you are done using them. It is best to use open software encryption tools, web browsers, and operating systems as they can be publicly reviewed for security problems and keep up to date with security fixes.

Be wary of opening PDF files using Adobe Reader or other proprietary PDF readers. Closed source PDF readers have been known to be used to execute malign code embedded in the PDF body. If you receive a .pdf as an attachment you should first consider if you know the supposed sender and if you are expecting a document from them. Secondly, you can use PDF readers which have been tested for known vulnerabilities and do not execute code via java script.

Linux: Evince, Sumatra PDF OS X:

Preview

Windows: Evince

Use short-term anonymous throw away accounts with randomly generated passwords whenever possible.

2.3.3 When Encryption Goes Wrong

What happens if I lose my "keys"? Do I lose my email?

Rigorous GPG encryption of email is not without its own problems.

If you store your email encrypted and lose all copies of your private key, you will be absolutely unable to read the old stored emails, and if you do not have a copy of your revocation certificate for the private key it could be difficult to prove that any new key you generate is truly the valid one, at least until the original private key expires.

If you sign a message with your private key, you will have great difficulty convincing anybody that you did not sign if the recipient of the message ever reveals the message and signature publicly. The term for this is *non-repudiation*: any message you send signed is excellent evidence in court. Relatedly, if your private key is ever compromised, it could be used to read all encrypted messages ever sent to you using your public key: the messages may be safe when they are in transit and just when they are received, but any copies are a liability and a gamble that the private key will never be revealed. In particular, even if you destroy every message just after reading it, anybody who snooped the message on the wire would keep a copy and attempt to decrypt it later if they obtained the private key.

The solution is to use a messaging protocol that provides *perfect forward secrecy* by generating a new unique session key for every conversation of exchange of messages in a random way such that the session keys could not be re-generated after the fact even if the private keys were known. The OTR chat protocol provides perfect forward secrecy (http://en.wikipedia.org/wiki/Perfect_forward_secrecy) for real time instant messaging, and the SSH protocol provides it for remote shell connections, but there is no equivalent system for email at this time.

It can be difficult to balance the convenience of mobile access to your private keys with the fact that mobile devices are much more likely to be lost, stolen, or inspected and exploited than stationary machines. An emergency or unexpected time of need might be exactly the moment when you would most want to send a confidential message or a signed message to verify your identity, but these are also the moments when you might be without access to your private keys if your mobile device was seized or not loaded with all your keys.

2.4 Secure Connections

2.4.1 Can other people read along when I check my email?

As discussed in the Chapter **Basic Tips**, whether you use webmail or an email program you should always be sure to use encryption for the entire session, from login to logout. This will keep anyone from spying on your communication with your email provider. Thankfully, this is easily done due to the popular use of *TLS/SSL* connections on email servers (See appendix **TLS/SSL**).

A *TLS/SSL* connection in the browser, when using webmail, will appear with `https` in the URL instead of the standard `http`, like so:

`https://gigglemail.com`

If your webmail host does not provide a *TLS/SSL* service then you should consider discontinuing use of that account; even if your emails themselves are not especially private or important, your account can very easily be hacked by “sniffing” your password! If it is

2.5 Secure Emails

not enabled already be sure to turn it on in your account options. At the time of writing, Google’s Gmail and Hotmail / Microsoft Live both automatically switch your browser to using a secure connection.

If you are using an email program like Thunderbird, Mail.app or Outlook, be sure to check that you are using TLS/SSL in the options of the program. See the chapter **Setting Up Secure Connections** in the section **Email Security**.

2.4.2 Notes

It's important to note that the administrators at providers like Hotmail or Google, that host, receive or forward your email can read your email even if you are using secure connections. It is also worth noting that the cryptographic keys protecting a TLS/SSL connection can be deliberately disclosed by site operators, or copied without their permission, breaching the confidentiality of that connection. It is also possible for a Certificate Authority to be corrupted or compromised so that it creates false certificates for keys held by eavesdroppers, making it much easier for a Man In The Middle Attack on connections using TLS/SSL (See Glossary for "Man in the Middle Attack"). An example of compromised E-mail providers is discussed here, implicating America's NSA and several email providers: <http://cryptome.info/0001/nsa-ssl-email.htm>

We also note here that a *Virtual Private Network* also a good way of securing your connections when sending and reading email but requires using a VPN client on your local machine connecting to a server. See the chapter **Virtual Private Networking** in the **Browsing** section.

2.5 Secure Emails

It is possible to send and receive secure email using standard current email programs by adding a few add-ons. The essential function of these add-ons is to make the message body (but not the To:, From:, CC: and Subject: fields) unreadable by any 3rd party that intercepts or otherwise gains access to your email or that of your conversation partner. This process is known as encryption.

Secure email is generally done using a technique called *Public-Key Cryptography*. Public-Key Cryptography is a clever technique that uses two code keys to send a message. Each user has a *public key*, which can only be used to encrypt a message but not to decrypt it. The public keys are quite safe to pass around without worrying that somebody might discover them. The *private keys* are kept secret by the person who receives the message and can be used to decode the messages that are encoded with the matching public key.

In practice, that means if Rosa wants to send Heinz a secure message, she only needs his public key which encodes the text. Upon receiving the email, Heinz then uses his private key to decrypt the message. If he wants to respond, he will need to use Rosa's public key to encrypt the response, and so on.

2.5.1 What software can I use to encrypt my email?

The most popular setup for public-key cryptography is to use *Gnu Privacy Guard (GPG)* to create and manage keys and an add-on to integrate it with standard email software. Using GPG will give you the option of encrypting sensitive mail and decoding incoming mail that has been encrypted but it will not force you to use it all the time. In years past, it was quite difficult to install and set up email encryption but recent advances have made this process relatively simple.

See section **Email Encryption** for working with GPG in the scope of your operating system and email program.

If you use a *webmail* service and wish to encrypt your email this is more difficult. You can use a GPG program on your computer to encrypt the text using your public key or you can use an add-on, like Lock The Text (<http://lockthetext.sourceforge.net/>). If you want to keep your messages private, we suggest using a dedicated email program like Thunderbird instead of webmail.

3 Understanding Browsing

3.1 Basic Tips

3.1.1 In Brief:

- When you visit a website you give away information about yourself to the site owner, unless precautions are taken.
- Your browsing on the Internet may be tracked by the sites you visit and partners of those sites. Use anti-tracking software.
- Visiting a website on the Internet is never a direct connection. Many computers, owned by many different people are involved. Use a secure connection to ensure your browsing can not be recorded.
- What you search for is of great interest to search providers. Use search anonymising software to protect your privacy.
- It is wiser to trust Open Source browsers like Mozilla Firefox as they can be more readily security audited.

3.1.2 Your browser talks about you behind your back

All browsers communicate information to the web server serving you a web page. This information includes name and version of the browser, referral information (a link on another site, for instance) and the operating system used.

Websites often use this information to customise your browsing experience, suggesting downloads for your operating system and formatting the web page to better fit your browser. Naturally however, this presents an issue and regards the user's own anonymity as this information becomes part of a larger body of data that can be used to identify you individually.

Stopping the chatter of your browser is not easily done. You can, however, falsify some of the information sent to web servers while you browse by altering data contained in the *User Agent*, the browser's identity. There is a very useful plugin for Firefox, for instance, called *User Agent Switcher* that allows you to set the browser identity to another profile selected from a drop down list of options.

3.1.3 Web sites can track you as you browse

Small files, called *cookies*, are often written onto your computer by web sites. Cookies present certain conveniences, like caching login data, session information and other data that makes your browsing experience smoother. These small pieces of data however

present a significant risk to your right to anonymity on the web: they can be used to identify you if you return to a site and also to track you as you move from site to site. Coupled with the User-Agent, they present a powerful and covert means of remotely identifying your person.

The ideal solution to this problem is deny all website attempts to write cookies onto your system but this can greatly reduce the quality of your experience on the web.

See the section **Tracking** for guides as to how to stop web servers tracking you.

3.1.4 Searching online can give away information about you

When we search online using services like Bing or Google our right to privacy is already at risk, vastly more so than asking a person at an Information Desk in an airport, for instance.

Combined with the use of cookies and User Agent data this information can be used to build an evolving portrait of you over time. Advertisers consider this information very valuable, use it to make assumptions about your interests and market you products in a targeted fashion.

While some customers may sing the praises of targeted advertising and others may not care, the risks are often misunderstood. Firstly, the information collected about you may be requested by a government, even a government you did not elect (Google, for instance, is an American company and so must comply with American judicial processes and political interests). Secondly there is the risk that merely searching for information can be misconstrued as intent or political endorsement. For instance an artist studying the aesthetics of different forms of Religious Extremism might find him or herself in danger of being associated with support for the organisations studied. Finally there is the risk that this hidden profile of you may be sold on to insurance agents, provided to potential employers or other customers of the company whose search service you are using.

Even once you've ensured your cookies are cleared, your *User Agent* has been changed (see above and chapter **Tracking**) you are still giving away one crucial bit of information: the Internet Address you are connecting from (see chapter **What Happens When You Browse**). To avoid this you can use an anonymising service like Tor (see chapter **Anonymity**). If you are a Firefox user be sure to install the excellent *Google Sharing* add-on, an anonymiser for Google search. Even if you don't consciously use Google, a vast number of web sites use a customised Google Search bar as a means of exploring their content.

With the above said, there are no reasons to trust Google, Yahoo or Bing. We recommend switching to a search service that takes your right to privacy seriously: Duck- DuckGo (<http://duckduckgo.com/>).

3.1.5 More eyes than you can see

The Internet is a big place and is not one network but a greater network of many smaller interconnected networks. So it follows that when you request a page from a server on the

Internet your request must traverse many machines before it reaches the server hosting the page. This journey is known as a *route* and typically includes at least 10 machines along the path. As packets move from machine to machine they are necessarily copied into memory, rewritten and passed on.

Each of the machines along a network route belongs to someone, normally a company or organisation and may be in entirely different countries. While there are efforts to standardise communication laws across countries, the situation is currently one of significant jurisdictional variation. So, while there may not be a law requiring the logging of your web browsing in your country, such laws may be in place elsewhere along your packet's route.

The only means of protecting the traffic along your route from being recorded or tampered with is using *end to end encryption* like that provided by TLS/Secure Socket Layer (See chapter **Encryption**) or a Virtual Private Network (See chapter **VPN**).

3.1.6 *Your right to be unknown*

Beyond the desire to minimise privacy leakage to specific service providers, you should consider obscuring the Internet Address you are connecting from more generally (see chapter **What Happens When You Browse**). The desire to achieve such anonymity spurred the creation of the *Tor Project*.

Tor uses an ever evolving network of nodes to route your connection to a site in a way that cannot be traced back to you. It is a very robust means of ensuring your Internet address cannot be logged by a remote server. See the chapter **Anonymity** for more information about how this works and how to get started with *Tor*.

3.2 **Fears**

3.2.1 *Social Networking - what are the dangers?*

The phenomenon of Internet based Social Networking has changed not just how people use the Internet but its very shape. Large data centers around the world, particularly in the US, have been built to cater to the sudden and vast desire for people to upload content about themselves, their interests and their lives in order to participate in Social Networking.

Social Networking as we know it with FaceBook, Twitter (and earlier MySpace) are certainly far from 'free'. Rather, these are businesses that seek to develop upon, and then exploit, a very basic anxiety: the fear of social irrelevance. As social animals we can't bear the idea of missing out and so many find themselves placing their most intimate expressions onto a businessman's hard-disk, buried deep in a data center in another country - one they will never be allowed to visit.

Despite this many would argue that the social warmth and personal validation acquired through engagement with Social Networks well out-weighs the potential loss of privacy. Such a statement however is only valid when the *full* extent of the risks are known.

The risks of Social Networking on a person's basic right to privacy are defined by:

- The scope and intimacy of the user's individual contributions.
- A user posting frequently and including many personal details constructs a body of information of greater use for targeted marketing.
- The preparedness of the user to take social risks.
- A user making social connections uncritically is at greater risk from predators and social engineering attacks.
- The economic interests and partners of the organisation providing the service.
- Commissioned studies from clients, data mining, sentiment analysis.
- Political/legal demands exerted by the State against the organisation in the jurisdiction(s) in which it is resident.
- Court orders for data on a particular user (whether civilian or foreigner).

- Surveillance agendas by law enforcement or partners of the organisation.
- Sentiment analysis: projections of political intent.

With these things in mind it is possible to chart a sliding scale between projects like Diaspora and Facebook: the former promises some level of organisational transparency, a commitment to privacy and a general openness, whereas Facebook proves to be an opaque company economically able to gamble with the privacy of their users and manage civil lawsuits in the interests of looking after their clients. As such there is more likelihood of your interactions with a large Social Network service affecting how an Insurance company or potential employer considers you than a smaller, more transparent company.

3.2.2 *Who can steal my identity?*

This question depends on the context you are working within as you browse. A weak and universal password presents a danger of multiple services from Social Networking, Banking, WebMail etc being account hijacked. A strong and universal password on a wireless network shared with others (whether open or encrypted) is just as vulnerable. The general rule is to ensure you have a strong password (see section on **Passwords**).

Wireless networks

Here we find ourselves amidst an often underestimated risk of someone listening in on your communications using *network packet sniffing*. It matters little if the network itself is open or password secured. If someone uses the same encrypted network, he can easily capture and read all unsecured traffic of other users within the same network. A wireless key can be acquired for the cost of a cup of coffee and gives those that know how to capture and read network packets the chance to read your password while you check your email.

A simple rule always applies: if the cafe offers a network cable connection, use it! Finally, just as at a bank machine, make sure no one watches over your shoulder when you type in the password.

The browser cache

Due to the general annoyance of having to type in your password repeatedly, you allow the browser or local mail client to store it for you. This is not bad in itself, but when a laptop or phone gets stolen, this enables the thief to access the owner's email account(s). The best practice is to clear this cache every time you close your browser. All popular browsers have an option to clear this cache on exit.

One precaution can justify you holding onto your convenient cache: disk encryption. If your laptop is stolen and the thief reboots the machine, they'll be met with an encrypted disk. It is also wise to have a screen lock installed on your computer or phone. If the machine is taken from you while still running your existing user session, it cannot be accessed.

Securing your line

Whenever you log into any service you should always ensure to use encryption for the entire session. This is easily done due to the popular use of *TLS/SSL (Secure Socket Layer)*.

Check to see the service you're using (whether Email, Social Networking or online-banking) supports TLS/SSL sessions by looking for **https://** at the beginning of the URL. If not, be sure to

turn it on in any settings provided by the service. To better understand how browsing the World Wide Web works, see the chapter **What Happens When I Browse?**

3.2.3 Can I get in trouble for Googling weird stuff?

Google and other search companies may comply with court orders and warrants targeting certain individuals. A web site using a customised Google Search field to find content on their site may be forced to log and supply all search queries to organisations within their local jurisdiction. Academics, artists and researchers are particularly at risk of being misunderstood, assumed to have motivations just by virtue of their apparent interests.

3.2.4 Who is keeping a record of my browsing and am I allowed to hide from them?

It is absolutely within your basic human rights, and commonly constitutionally protected, to visit web sites anonymously. Just as you're allowed to visit a public library, skim through books and put them back on the shelf without someone noting the pages and titles of your interest, you are free to browse anonymously on the Internet.

3.2.5 How to not reveal my Identity?

See the chapter on **Anonymity**.

3.2.6 How to avoid being tracked?

See the chapter on **Tracking**.

3.3 What happens when you browse

Browsing the web is communicating. You might not send as much text in terms of number of words, but it is always the browser which initiates and maintains the communication by requesting the bits and pieces which are woven into what is eventually displayed on your screen.

Browsers like Mozilla Firefox, Google Chrome, Opera, Safari & Internet Explorer all work in a similar manner. When we type a URL (e.g. "<http://happybunnies.com>") in the address bar, the browser requests the website (which is just a special kind of text) from a remote server and then transforms it into colored blocks, text and images to be displayed in the browser window. To see the text the way the browser sees it, one just has to click on the View --> Page source menu entry in the browser. What comes up is the same webpage but in HTML – a language mainly concerned with content, context and links to other resources (CSS and JavaScript) which govern the way these contents are displayed and behave.

When the browser tries to open a webpage – and assuming there are no proxies involved – the first thing it does is to check its own cache. If there is no past memories of such website, it tries to resolve the name into an address it can actually use. It is an internet program, so it needs an Internet Protocol address (IP address or just IP). To get this address it asks a DNS Server (kind of a telephone book for internet programs) which is installed in the router of your internet access by default. The IP address is a numerical label assigned to every device in the (global) network, like the address of a house in the postal system – and as the address of your home, you should be very careful to whom you hand out the IP address you are browsing from (by default this is: to everyone).

Once the IP address has been received, the browser opens a TCP (just a communication protocol) connection to the destination host and starts sending packages to a port at this address, typically no. 80 (ports are like doors to the servers, there are many but usually only a few are open), unless another path is specified. These packages travel through a number of servers on the internet (up to a couple of dozens depending on where the target address is located). The server then looks for the requested page and, if found, delivers it using the HTTP protocol. (To prevent others from reading or altering the data, TLS/SSL can be used to below HTTP to secure the connection)

When the HTTP response arrives, the browser can close the TCP connection or reuse it for subsequent requests. The response can be one of many things, from some sort of redirection or a classic Internal Server Error (500). Provided the response proceeds as expected the browser will store the page in a cache for further use, decode it (uncompress it

3.3 What happens when you browse

if compressed, rendered if video codec, etc) and display/play it according to instructions.

Now, the process can be illustrated in a little conversation between browser (B) and server (S):

B: "Hallo."

S: "Hey!"

B: "May I get that page with the happy bunnies, please?" S: "Well, here you are."

B: "Oh, maybe you could also give me a big version of that picture of that bunny baby cuddling a teddy bear."

S: "Sure, why not?"

[. . .]

B: "That's all for now. Thank you. Bye."

Note that there are lots of activities happening parallel to this TCP/IP exchange. Depending on how you have configured its options, your browser might be adding the page to browser history, saving cookies, checking for plugins, checking for RSS updates and communicating with a variety of servers, all while you're doing something else.

3.3.1 A topography of you: footprints

Most important: you will leave footprints. Some of them will be left on your own computer – a collection of cache data, browsing history and naughty little files with elephantine memory called cookies. They are all very convenient; speed up your browser's performance, reduce your data download or remember your passwords and preferences from Social Networks. They also snitch on your browsing habits and compile a record of everywhere you go and everything you do there. This should bother you if you are using a public computer station at a library, work at a cybercafe, or share your apartment with a nosy partner!

Even if you configure your browser to not keep a history record, reject cookies and delete cached files (or allocate zero MB of space for the cache), you would still leave breadcrumbs all over the Internet. Your IP address is recorded by default everywhere, by everyone and the packets sent are monitored by an increasing number of entities - commercial, governmental or criminal, along with some creeps and potential stalkers.

Democratic governments everywhere are redesigning regulations to require Internet providers to keep a copy of everything so they can have later access to it. In the USA, section 215 of the American PATRIOT act '*prohibits an individual or organization from revealing that it has given records to the federal government, following an investigation*'. That means that the company you pay every month as a customer to provide you with Internet access can be ordered to turn over your browsing and email records without your knowledge.

Most of the time, though, surveillance is not a 1984 affair. Google collects your searches along with your browser identification (*user agent*), your IP and a whole bunch of data that can eventually lead to your doorstep, but the ultimate aim is usually not political repression but market research. Advertisers don't fuss about advertising space any more, they just want to know everything about you. They want to know your dietary and

medication habits, how many children you have and where you take them on holidays; how you make your money, how much you earn and how you like to spend it. Even more: they want to know how you *feel* about stuff. They want to know if your friends respect those feelings enough so that you can convince them to change their consumption habits. This is not a conspiracy, but rather the nature of Information Age capitalism. To paraphrase a famous observation of the current situation, the best minds of our generation are thinking about how to make people click ads.⁴

Some people think ads can be ignored or that having advertisers cater for our specific needs is a win-win situation, because at least they are spammed with things they may actually want. Even if that was the case (it isn't): should we trust Google with such intimate details of our life? Even if we trust Google to 'do no evil', it can still be bought by someone we do not trust; benevolent Larry Page and Sergey Brin could be overruled by their own Board, or their data base be sequestered by a fascistic government. One of their 30,000 employees worldwide could cut loose and run with our data. Their servers can be hacked. And in the end, they are just interested in their customers, *the companies paying for advertising*. We are just the product being sold.

Moreover; in the Social Networks our browsing habits are generating a Permanent Record, a collection of data so vast that the information that Facebook keeps about a given user alone can fill 880 pages. Nobody will be surprised to learn that Facebook's purpose is not to make us happy – again: if you are not paying for it, you're not the customer, you're the product. But even if you don't care about their commercial goals, consider this: the platform has publicly admitted hackers break into hundreds of thousands of Facebook accounts every day.

For a taste of what lurks behind the curtains of the websites you visit, install a plugin/add-on called *Ghostery* to your browser. It's like an x-ray-machine which reveals all the surveillance technology which might be (and often is) embedded in a web page, normally invisible to the user. In the same line, *Do Not Track Plus* and *Trackerblock* will give you further control over online tracking, through cookie blocking, persistent opt-out cookies, etc. Our following chapter **Tracking** will equip you with expertise in such topics.

Even in between your computer and the router, your packages can easily be intercepted by anyone using the same wireless network in the casual environment of a cafe. It is a jungle out there, but still we choose passwords like "password" and "123456", perform economic transactions and buy tickets on public wireless networks and click on links from unsolicited emails. It is not only our right to preserve our privacy but also our responsibility to defend that right against the intrusions of governments, corporations and anyone who attempts to dispossess us. If we do not exercise those rights today, we deserve whatever happens tomorrow.

- 1.If you are a Unix user, you can use the tcpdump command in the bash and view real time dns traffic. It's loads of fun! (and disturbing) ^
- 2.See list of TCP and UDP port numbers (http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers)
- 3.If this exchange is happening under an HTTPS connection, the process is much

3.4 Accounts and Security

more complicated and also much safer, but you will find out more about that in a most fascinating chapter called Encryption. ^

4. This Tech Bubble Is Different ([http://www.businessweek.com/magazine/content/ 11_17/b4225060960537.htm](http://www.businessweek.com/magazine/content/11_17/b4225060960537.htm)), Ashlee Vance (Businessweek magazine) ^

3.4 Accounts and Security

When you browse, you may be logged into various services, sometimes at the same time. It may be a company website, your email or a social networking site. Our accounts are important to us because highly sensitive information about us and others is stored on machines elsewhere on the Internet.

Keeping your accounts secure requires more than just a strong password (see section **Passwords**) and a secure communication link with the server via TLS/SSL (see chapter **Secure Connection**). Unless specified otherwise, most browsers will store your login data in tiny files called cookies, reducing the need for you re-type your password when you reconnect to those sites. This means that someone with access to your computer or phone may be able to access your accounts without having to steal your password or do sophisticated snooping.

As smart phones have become more popular there has been a dramatic rise in account hijacking with stolen phones. Laptops theft presents a similar risk. If you do choose to have the browser save your passwords then you have a few options to protect yourself:

- Use a screen lock. If you have a phone and prefer an unlock pattern system get in the habit of wiping the screen so an attacker can not guess the pattern from finger smears. On a Laptop, you should set your screensaver to require a password as well as a password on start-up.
- Encrypt your hard disk. TrueCrypt is an open and secure disk encryption system for Windows 7/Vista/XP, Mac OS X and Linux. OSX and most Linux distributions provide the option for disk encryption on install.
- Android Developers: do not enable USB debugging on your phone by default. This allows an attacker using the Android *adb shell* on a computer to access your phone's hard disk without unlocking the phone.

3.4.1 Can malicious web sites take over my accounts?

Those special cookies that contain your login data are a primary point of vulnerability. One particularly popular technique for stealing login data is called click-jacking, where the user is tricked into clicking on a seemingly innocuous link, executing a script that takes advantage of the fact you are logged in. The login data can then be stolen, giving the remote attacker access to your account. While this is a very complicated technique, it has proven effective on several occasions. Both Twitter and Facebook have seen cases of login sessions being stolen using these techniques.

It's important to develop a habit for thinking before you click on links to sites while logged into your accounts. One technique is to use another browser entirely that is not

logged into your accounts as a tool for testing the safety of a link. Always confirm the address (URL) in the link to make sure it is spelled correctly. It may be a site with a name very similar to one you already trust. Note that links using URL shorteners (like <http://is.gd> and <http://bit.ly>) present a risk as you cannot see the actual link you are requesting data from.

If using Firefox on your device, use the add-on [NoScript](#) as it mitigates many of the *Cross Site Scripting* techniques that allow for your cookie to be hijacked but it will disable many fancy features on some web sites.

3.5 Tracking

When you browse the web tiny digital traces of your presence are left behind. Many web sites harmlessly use this data to compile statistics and see how many people are looking at their site and which pages are popular, but some sites go further and use various techniques to track individual users, even going as far as trying to identify them personally. It doesn't stop there however. Some firms store data in your web browser which can be used to track you on other web sites. This information can be compiled and passed on to other organizations without your knowledge or permission.

This all sounds ominous but really who cares if some big company knows about a few web sites that we have looked at? Big web sites compile and use this data for "behavioral advertising" where ads are tailored to fit your interests exactly. That's why after looking at say, the Wikipedia entry for Majorca, one may suddenly start seeing lots of ads for packaged vacations and party hats. This may seem innocent enough, but after doing a search for "Herpes Treatments" or "Fetish Communities" and suddenly seeing listings for relevant products, one may start to feel that the web is getting a bit too familiar.

Such information is also of interest to other parties, like your insurance company. If they know you have been looking at skydiving sites or forums for congenital diseases, your premiums may mysteriously start going up. Potential employers or landlords may turn you down based on their concerns about your web interests. In extreme instances, the police or tax authorities may develop an interest without you ever having committed a crime, simply based on suspicious surfing.

3.5.1 How do they track us?

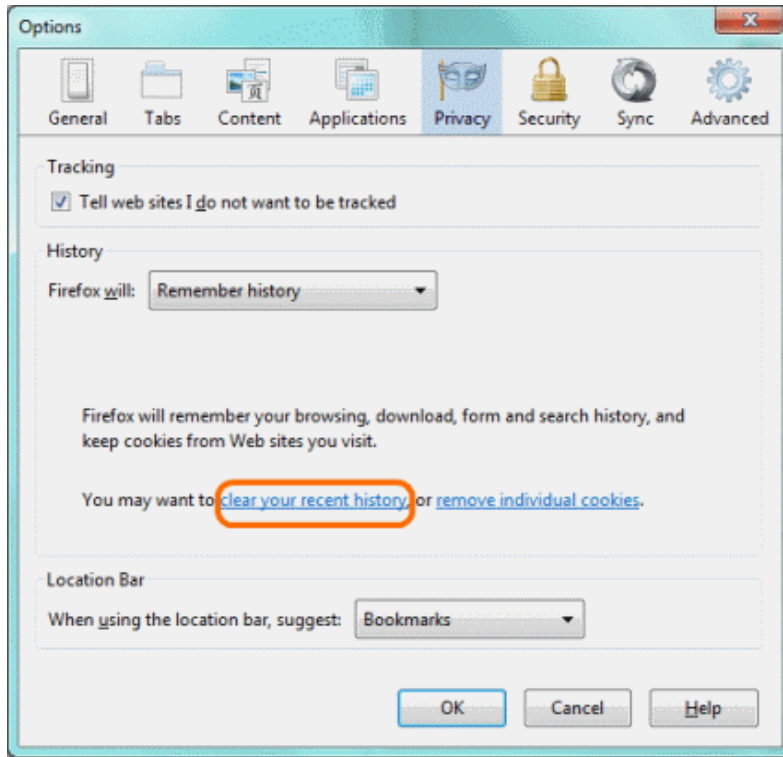
Every time you load a web page, the server software on the web site generates a record of the page viewed in a log file. This is not always a bad thing. When you log in to a website, there is a need for a way to establish your identity and keep track of who you are in order to save your preferences, or present you with customized information. It does this by passing a small file to your browser and storing a corresponding reference on the web server. This file is called a *cookie*. It sounds tasty but the problem is that this information stays on your computer even after leaving the web site and may phone home to tell the owner of the cookie about other web sites you are visiting. Some major sites, like Facebook and Google have been caught using them to keep track of your browsing even after you have logged out.

Supercookies / Evercookie / Zombie Cookies?

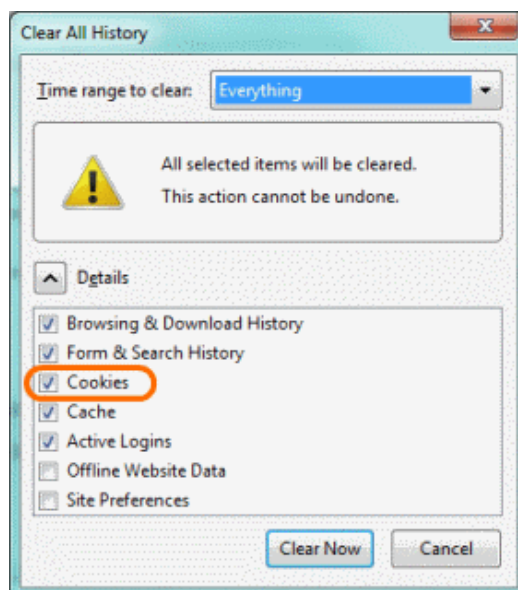
3.5.2 How can I prevent tracking?

The simplest and most direct way to deal with tracking is to delete the cookie files in your browser:

In **Firefox**: 1. Click the **Firefox menu**. 2. Click **Options**. 3. Click **Privacy**. 4. Click **Clear your recent history**.



to clear is set to **Everything**. 6. Tick **Cookies**.



Click **Clear now**.

In **Chrome**: 1. Click the **Chrome menu**. 2. Click **Tools**. 3. Click **Clear browsing data**. 4. Make sure **Obliterate the following items from** is set to **The beginning of time**. 5. Tick **Delete cookies and other site and plug-in data**. 6. Click **Clear browsing data**.

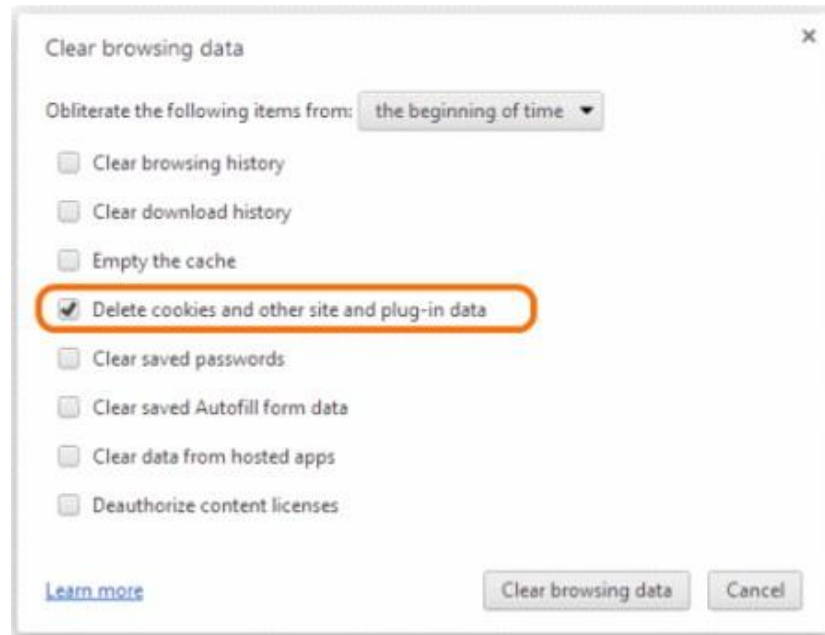


Figure 3.1:Delete Chrome Cookies

In **Internet Explorer**: 1. Click the **Tools** button (shaped like a gear). 2. Click **Safety**. 3. Click **Delete Browsing History**. 4. Tick **Cookies**. 5. Click **Delete**.

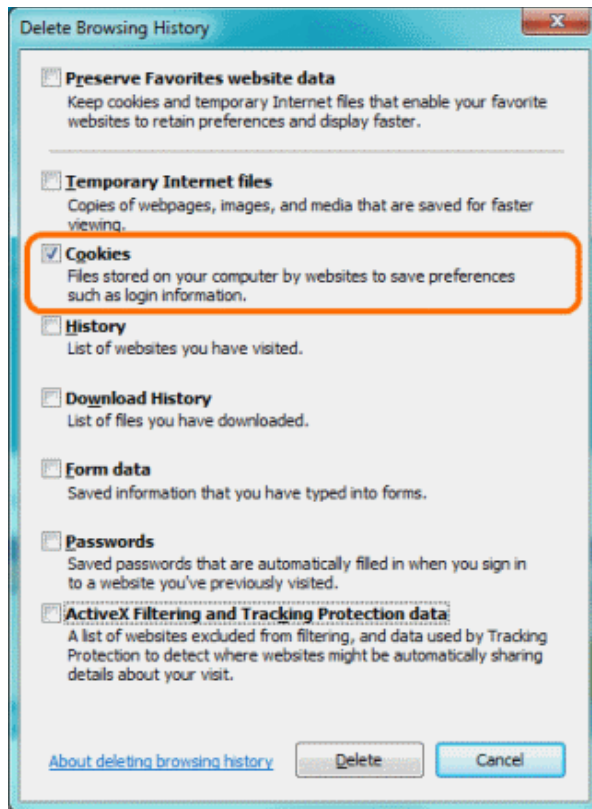
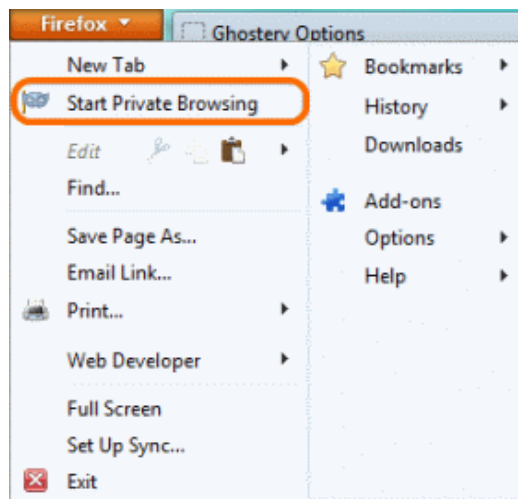


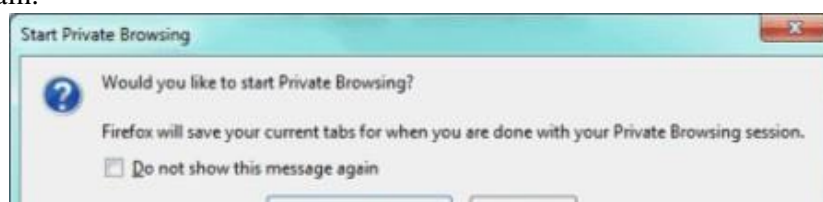
Figure 3.2:Delete IE Cookies

The limitation to this approach is that you will receive new cookies as soon as you return to these sites or go to any other pages with tracking components. The other disadvantage is that you will lose all of your current login sessions for any open tabs, forcing you to type in usernames and passwords again. A more convenient option, supported by current browsers is private browsing or incognito mode. This opens a temporary browser window that does not save the history of pages viewed, passwords, downloaded files or cookies. Upon closing the private browsing window, all of this information is deleted. You can enable private browsing:

In **Firefox**: 1. Click the **Firefox menu**. 2. Click **Start Private Browsing**.



When prompted, click **Start Private Browsing** again.



button turns purple, showing that private browsing is on.

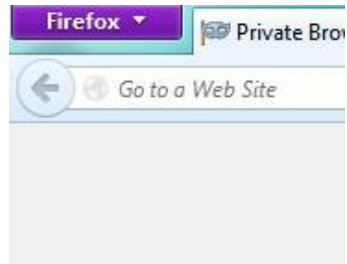
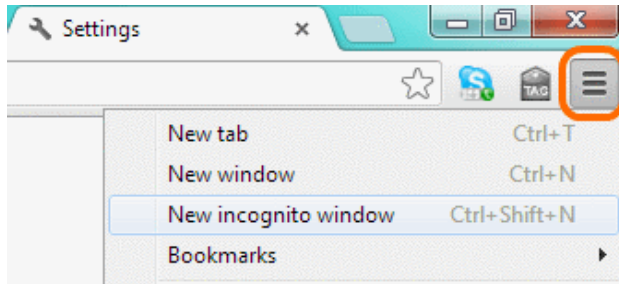


Figure 3.3:Firefox Private Browsing

4. The Firefox menu

In **Chrome**: 1. Click the **Chrome menu**. 2. Click **New incognito window**.



3. The **spy icon** in the top-left of the browser window shows that private browsing is on.

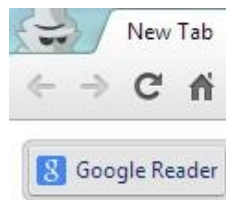
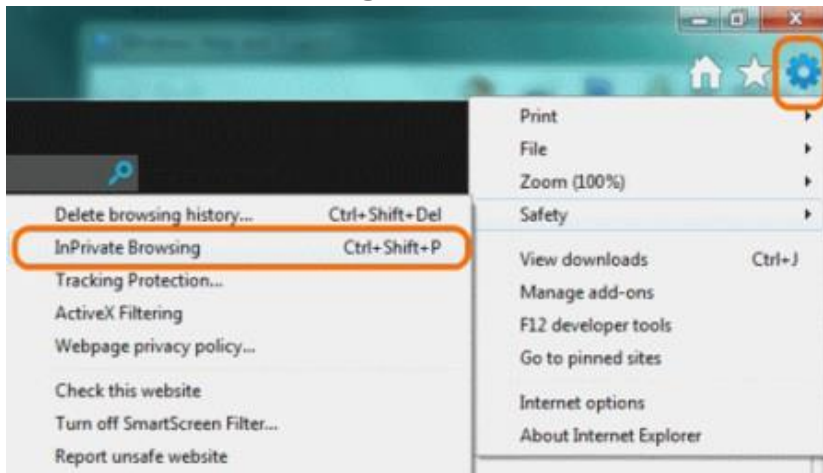


Figure 3.4: Chrome Private Browsing

In **Internet Explorer**: 1. Click the **Tools** menu, shaped like a gear. 2. Click **Safety**. 3. Click **InPrivate Browsing**.



4. The **InPrivate** logo appears in the top-left of your browser window, showing that private browsing is on.

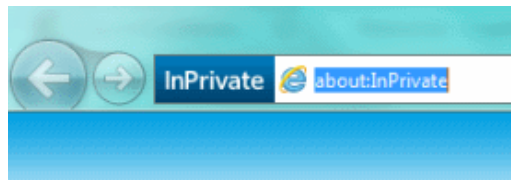
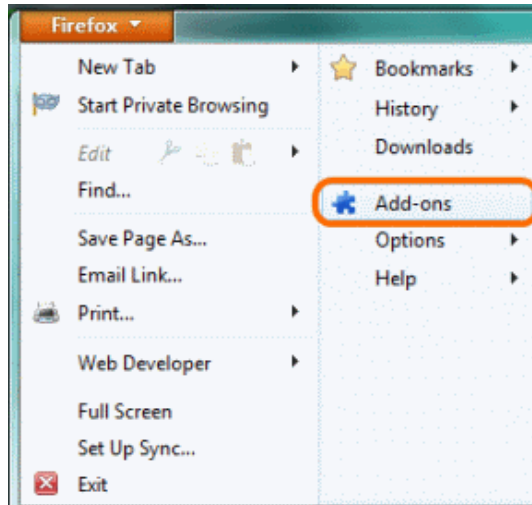


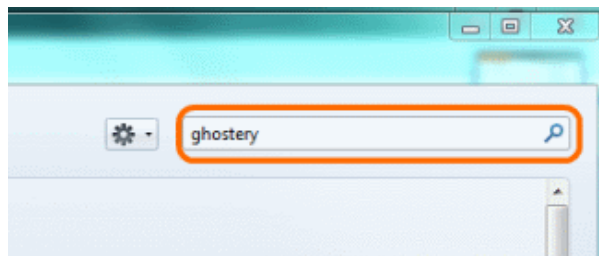
Figure 3.5: IE Private Browsing

This solution also has its limitations. We cannot save bookmarks, remember passwords, or take advantage of much of the convenience offered by modern browsers. Thankfully, there are several plugins specially designed to address the problems of tracking. The most extensive, in terms of features and flexibility, is Ghostery. The plugin allows you to block categories or individual services that track users. Here's how you install Ghostery:

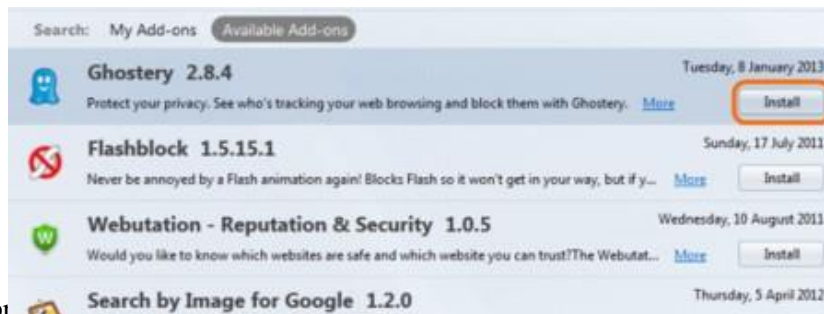
1. In Firefox, click the **Firefox** menu and select **Add-ons**.



the **Search** box, type “ghostery”, then click the **Search** icon or press **Enter**.



and Ghostery in the list of Add-ons, and click **Install**.



4. Restart your browser by clicking **Restart Now**.



5. Click the **Ghostery**

toolbar and select **Options**. Do the walkthrough and/or play with Ghostery's settings, if you want.



6. V and have a look at its trackers.

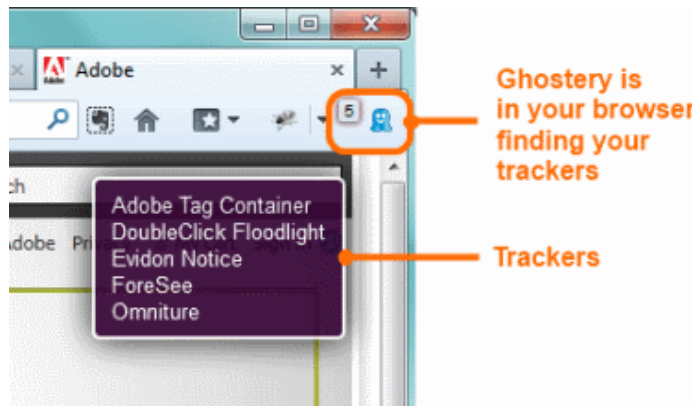


Figure 3.6: Ghostery

Another option is to install an ad-blocking plugin like AdblockPlus. This will automatically block many of the tracking cookies sent by advertising companies but not those used by Google, Facebook and other web analytics companies. [expand on this maybe, explain “web analytics”]

How can I see who is tracking me?

The easiest way to see who is tracking you is to use the Ghostery plugin. There is a small icon on the upper right or lower right corner of your browser window that will tell you which services are tracking you on particular web sites.

{Suggestion: Add Abine.com's Do Not Track add-on. I suggest using both Ghosterly and DNT, as occasionally they block a different cookie. Abine also has Privacy Suite, recently developed which can give a proxy telephone and proxy email, similar to 10 Minute Mail or Guerrilla Mail for fill-in emails for forms.}

3.5.3 A word of warning

If you block trackers, you will have a higher level of privacy when surfing the net. However, government agencies, bosses, hackers and unscrupulous network administrators will still be able to intercept your traffic and see what you are looking at. If you want to secure your connections you will need to read the chapter on encryption. Your identity may also be visible to other people on the internet. If you want to thoroughly protect your identity while browsing, you will need to take steps toward online anonymity which is explained in another section of this book.

3.6 Anonymity

3.6.1 Intro

Article 2 of the Universal Declaration of Human Rights states:

“Everyone is entitled to all the rights and freedoms set forth in this Declaration, without distinction of any kind, such as race, colour, sex, language, religion, political or other opinion, national or social origin, property, birth or other status.

Furthermore, no distinction shall be made on the basis of the political, jurisdictional or international status of the country or territory to which a person belongs, whether it be independent, trust, non-self-governing or under any other limitation of sovereignty.“.

One way of enforcing this basic right in hostile environments is by means of anonymity, where attempts to connect an active agent to a specific person are blocked.

Acting anonymously is also a great way to help others with a high need for protection – the bigger the herd of sheep, the harder it is to target a specific one. An easy way to do so is by using Tor, a technique which routes internet traffic between users of a special software, thus making it untraceable to any specific IP address or person without having control over the whole network (and nobody has that yet in the case of the internet). A highly functional means to protect ones own identity is by using anonymous proxy servers and Virtual Private Networks (VPN).

3.6.2 Proxy

“An **anonymizer** or an **anonymous proxy** is a tool that attempts to make activity on the Internet untraceable. It is a proxy [server] computer that acts as an intermediary and privacy shield between a client computer and the rest of the Internet. It accesses the Internet on the user’s behalf, protecting personal information by hiding the client computer’s identifying information.”
(<http://en.wikipedia.org/wiki/Anonymizer>)

3.6 Anonymity

The main purpose behind using a proxy is to hide or to change Internet address (IP address) assigned to user’s computer. There can be a few reasons for needing to do so, for example:

- To anonymize access to particular server(s) and/or to obfuscate traces left in the log files of a web-server. For instance a user might need/want to access sensitive materials online (special materials, research topics or else) without triggering authorities attention.
- To break through firewalls of corporations or repressive regimes. A corporation/government can limit or completely restrict Internet access for a particular IP address or a range of IP addresses. Hiding behind a proxy will help to trick these filters and access otherwise forbidden sites.
- To watch online video and streams banned in your country due to legal issues.
- To access websites and/or materials available only for IP addresses belonging to a specific country. For example, a user wants to watch a BBC video stream (UK-only) while not residing in the UK.
- To access the Internet from a partially banned/blocked IP address. Public IP addresses

can often have “bad reputation” (bandwidth abuse, scam or unsolicited email distribution) and be blocked by some web-sites and servers.

While a usual scenario would be to use proxy for accessing the Web (HTTP), practically Internet protocol can be proxied - i.e. sent via a remote server. Unlike a router, proxy server is not directly forwarding remote user requests but rather mediates those requests and echos responses back to remote user’s computer.

Proxy (unless setup as “transparent”) does not allow direct communication to the Internet thus applications such as browsers, chat-clients or download applications need to be made aware of the proxy server (see **Safer Browsing/Proxy settings** chapter)

3.6.3 Tor

- Tor prevents anyone from learning your location or browsing habits.
- Tor is for web browsers, instant messaging clients, remote logins, and more.
- Tor is free and open source for Windows, Mac, Linux/Unix, and Android.
(<https://www.torproject.org>)

Tor is a system intended to enable online anonymity, composed of client software and a network of servers which can hide information about users’ locations and other factors which might identify them. Imagine a message being wrapped in several layers of protection: every server needs to take off one layer, thereby immediately deleting the sender information of the previous server.

Use of this system makes it more difficult to trace Internet traffic to the user, including visits to Web sites, online posts, instant messages, and other communication forms. It is intended to protect users’ personal freedom, privacy, and ability to conduct confidential

business, by keeping their internet activities from being monitored. The software is open-source and the network is free of charge to use.

Tor cannot and does not attempt to protect against monitoring the traffic entering and exiting the network. While Tor does provide protection against traffic analysis, it cannot prevent traffic confirmation (also called end-to-end correlation). *End-to-End Correlation* is a way of matching an online identity with a real person.

A recent case of this involved the FBI wanting to prove that the man Jeremy Hammon was behind an alias known to be responsible for several Anonymous attacks. Sitting outside his house, the FBI were monitoring his wireless traffic alongside a chat channel the alias was known to visit. When Jeremy went online in his apartment, inspection of the wireless packets revealed he was using Tor at the same moment the suspected alias associated with him came online in the surveilled chat channel. This was enough to incriminate Jeremy and he was arrested.

See section **Safer Browsing/Using Tor** for setup instructions.

3.7 VPN

The way your data makes it to the desired server and back to your laptop computer or a mobile device is not as straightforward as it might first seem. Say, you are connected to a wireless network at home and opening a wikipedia.org page. The path your request (data) takes will consist of multiple middle points or “hops” - in network-architect terminology. At each of these

hops (which are likely to be more than 5) your data can be scooped, copied and potentially modified.

- Your wireless network (your data can be sniffed from the air)
- Your ISP (in most countries they are obliged to keep detailed logs of user activity)
- Internet Exchange Point (IXP) somewhere on another continent (usually more secure than any other hop)
- ISP of the hosting company that hosts the site (is probably keeping logs)
- Internal network to which the server is connected
- And multiple hops between. . .

Any person with physical access to the computers or the networks which are on the way from you to the remote server, intentionally or not, can collect and reveal the data that's passing from you to the remote server and back. This is especially true for so called 'last mile' situations - the few last leaps that an internet connection makes to reach a user. That includes domestic and public wireless or wired networks, telephone and mobile networks, networks in libraries, homes, schools, hotels. Your ISP can not be considered a safe, or 'data-neutral' instance either - in many countries state agencies do not even require a warrant to access your data, and there is always the risk of intrusion by paid attackers working for a deep-pocketed adversaries.

VPN - a Virtual Private Network - is a solution for this 'last-mile' leakage. VPN is a technology that allows the creation of a virtual network on top of an existing infrastructure. Such a VPN network operates using the same protocols and standards as the

3.7 VPN

underlying physical network. Programs and OS use it transparently, as if it was a separate network connection, yet its topology or the way how network nodes (you, the VPN server and, potentially, other members or services available on VPN) are interconnected in relation to the physical space is entirely redefined.

Imagine that instead of having to trust your data to every single middle-man (your local network, ISP, the state) you have a choice to pass it via a server of a VPN provider whom you trust (after a recommendation or research) - from which your data will start its journey to the remote location. VPN allows you to recreate your local and geo-political context all together - from the moment your data leaves your computer and gets into the VPN network it is fully secured with TLS/SSL type encryption. And as such it will appear as pure random noise to any node who might be spying after you. It is as if your data was traveling inside a titanium-alloy pipe, unbreakable on all the way from your laptop to the VPN server. Of course one could argue that eventually, when your data is outside the safe harbour of VPN it becomes just as vulnerable as it was - but this is only partially true. Once your data exits the VPN server it is far away from you - way beyond the reach of some creeps sniffing on the local wireless network, your venal ISP or a local government obsessed with anti-terrorism laws. A serious VPN provider would have their servers installed at a high-security Internet exchange location, rendering any physical human access, tapping or logging a difficult task.

“Today everything you do on the Internet is monitored and we want to change that. With our fast VPN service you get totally anonymous on the Internet. It's also possible to surf censored web sites, that your school, ISP, work or country are blocking. [DarkVPN] will not only help people to surf anonymously, it also helps people in countries like China to be able to surf censored web pages. Which is your democratic right. DarknetVPN gives all VPN users an anonymous IP address. All electronic tracks will end up with us. We do not save any log files in order to

achieve maximum anonymity. With us you always surfing anonymously, secure and encrypted.” (<http://www.darknetvpn.com/about.php>)

Another interesting and often underrated features of VPN is encoded in its name - besides being **V**irtual and **P**rivate it is also a **N**etwork. VPN allows one not only to connect via the VPN server to the rest of the world but also to communicate to other members of the same VPN network without ever having to leave the safety of encrypted space. Through this functionality Virtual Private Network becomes something like a *DarkNet* (in a broader sense of the definition) - a network isolated from the Internet and inaccessible to “others”. Since a connection to VPN server, and thus the private network it facilitates, require a key or a *certificate*, only “invited” users are allowed. There is no chance that Internet stranger would gain access to what’s on a VPN without enrolling as a user or stealing someones keys. While not referred to as such, any corporate Intranet type of network is a DarkNet too.

“A virtual private network (VPN) extends a private network across a public network, such as the Internet. It enables a computer to send and receive data

across shared or public networks as if it were directly connected to the private network, while benefitting from the functionality, security and management policies of the private network.”(http://en.wikipedia.org/wiki/Virtual_private_network)

Many commercial VPN providers stress the anonymity that their service provides. Quoting IPredator.org page (a VPN service started by the people behind The Pirate Bay project):

“You’ll exchange the IP address you get from your ISP for an anonymous IP address. You get a safe/encrypted connection between your computer and the Internet”. (<https://www.ipredator.se>)

Indeed, when you access the Internet via a VPN connection it does appear as if the connection is originating from the IP address of IPredator servers.

4 Publishing And Distribution

4.1 Publishing Anonymously

Whether you are an activist operating under a totalitarian regime, an employee determined to expose some wrongdoings in your company or a vengeful writer composing a bitchy portrait of your ex-wife, you need to protect your identity. If you are not collaborating with others, the focus lies on anonymity and not encryption or privacy.

If the message is urgent and the stakes are high, one easy way to just get it out quickly is going to an internet cafe one usually does not frequent, create accounts specifically set up for the task, deliver the data and discard those accounts right after that. If you are in a hurry, consider MintEmail (<http://www.mintemail.com/>) or FilzMail (<http://www.filzmail.com/>), where your address will expire from 3 to 24 hours respectively. Do not do anything else while you’re there; don’t check your Gmail account, do not have a quick one on Facebook and clear all cache, cookies and history and close the browser before you leave.

If you keep these basic rules, the worst – though highly improbable – thing that could happen would be that the offered computer is compromised and logging keystrokes, revealing passwords or even your face, in case an attached webcam is remotely operated. Don't do this at work or in a place where you are a registered member or a regular visitor, like a club or a library.

If you want to maintain a constant stream of communication and maybe even establish an audience, this method quickly becomes quite cumbersome, and you might also run out of unused internet cafes. In this case you can use a machine you own, but, if you cannot dedicate one especially to this purpose, boot your computer with a different operating system (OS). This can be easily done by using a USB stick to boot a live operating system like Tails (<https://tails.boum.org/>), which comes with Tor enabled by default and includes state-of-the-art cryptographic tools. In any case, use Tor to disguise your IP.

Turn off all cookies, history and cache options and never use the same profile or the same browser for other activities. Not only would that add data to your topography as a user in the Net, but it also opens a very wide window for mistakes. If you want extra support, install *Do Not Track Plus* and *Trackerblock* or *Ghostery* in your browser add-ons menu.

Use passwords for different accounts and choose proper passwords or even passphrases (more about that in the basic tips section). Protect your entire system with a general password, change it often and do not share it with anyone, *especially* not your lover. Install a keystroke logger to see if someone sneaks into your email, especially your lover. Set up your preferences everywhere to log out of every service and platform after 5

minutes of non-use. Keep your superhero identity to yourself.

If you can maintain such level of discipline, you should even be capable of using your own internet connection. But consider this: not using a dedicated system makes it incredibly difficult to keep all the different identities separated in a safe way, and the feeling of safety often leads to carelessness. Keep a healthy level of neurosis.

Today there are many publishing possibilities, from cost-free blogging sites (Blogspot, Tumblr, WordPress, Identi.ca) to PasteBins (see glossary) and some specifically catered to anonymous users like BlogACause. Global Voices Advocacy recommends using Word-Press through the Tor network. Keep a sane level of cynicism; they all act in commercial interests that you use for 'free' and so cannot be trusted at all, especially in that they may be bound to the demands of a legal jurisdiction that is not your own. All providers are, when it comes down to it, traitors.

If registration with these services requires a working email address, create one dedicated solely to this purpose. Avoid Gmail, Yahoo, Hotmail and other big commercial platforms with a history of turning over their users and go for an specialized service like Hushmail (<https://www.hushmail.com/>). For more on anonymous email, please find the chapter Anonymous email in the previous section.

4.1.1 Several Don'ts

Don't register a domain. There are services that will protect your identity from a simple who is query, like Anonymous Speech or Silent Register, but they will know who you are through your payment data. Unless you have the chance to purchase one in BitCoins, limit yourself to one of the domains offered by your blogging platform like `yourblogname.blogspot.com` and choose a setting outside your native country. Also, find a name that doesn't give you away easily. If you have problems with that, use a blog name generator online.

Don't open a social network account associated to your blog. If you must, keep the level of hygiene that you keep for blogging and never ever login while using your regular

browser. If you have a public social network life, avoid it all together. You will eventually make a mistake.

Don't upload video, photo or audio files without using an editor to modify or erase all the meta data (photos contain information up to the GPS coordinates of the location the photo was taken at) that standard digital cameras, SmartPhones, recorders and other devices add by default. The *Metadata Anonymisation Toolkit* or *ExifTool* might help you with that.

Don't leave a history. Add X-Robots-Tag to your http headers to stop the searching spiders from indexing your website. That should include repositories like the Wayback Machine from archive.org. If you don't know how to do this, search along the lines of "Robots Text File Generator".

Don't leave comments. If you must, maintain the levels of hygiene that you use for blogging and always logout when you're done and for god sakes do not troll around. Hell hath no fury like a blogger scorned.

Don't expect it to last. If you hit the pot and become a blogging sensation (like
4.2 Anonymous Email

Belle de Jour, the British PhD candidate who became a sensation and sold a book and mused two TV shows about her double life as a high escort) there will be a legion of journalists, tax auditors and obsessive fans scrutinizing your every move. You are only human: they will get to you.

Don't linger. If you realize you have already made any mistakes but nobody has caught you yet, do close all your accounts, cover your tracks and start a totally new identity. The Internet has infinite memory: one strike, and you're out of the closet.

4.2 Anonymous Email

Every data packet traveling through the Internet contains information about its sender and its recipient. This applies to email as well as any other network communication. There are several ways to reduce identifying information but no way to remove it completely.

4.2.1 Sending From Throw-away Email Accounts

One option is to use a throw-away email account. This is an account set up at a service like Gmail or Hotmail, used once or twice for anonymous exchange. When signing up for the account, you will need to provide fake information about your name and location. After using the account for a short amount of time, say 24 hours, you should never log in again. If you need to communicate further, then create a new account.

It is very important to keep in mind that these services keep logs of the IP addresses of those using them. If you are sending highly sensitive information, you will need to combine a throw away email account with Tor in order keep your IP address hidden.

If you are not expecting a reply, then an anonymous remailer like AnonEmail or Silentsender may be a useful solution. A remailer is a server that receives messages with instructions on where to send the data and acts as a relay, forwarding it from a generic address without revealing the identity of the original sender. This works best when combined with an email provider like Hushmail or RiseUp who are specially set up for secure email connections.

Both of these methods are useful, but only if you always remember that the intermediary himself knows where the original message came from and can read the messages as they come in. Despite their claims to protect your identity, these services often have user agreements that

indicate their right “to disclose to third parties certain registration data about you” or they are suspected to be compromised by secret services. The only way to safely use this technique is to not trust these services at all, and apply extra security measures: send via Tor using a throw-away email address.

If you only need to receive email, services like Mailinator and MintEmail give you an email address that destroys itself after a few hours. When signing up for any account, you should provide fake information about your name and location and protect yourself by using Tor.

4.2.2 Be Careful about what you say!

The content of your message can give away your identity. If you mention details about your life, your geography, social relations or personal appearance, people may be able to determine who is sending the message. Even word choice and style of writing can be used to guess who might be behind anonymous emails.

You should not use the same user name for different accounts or use a name that you are already linked to like a childhood nickname or a favorite book character. You should never use your secret email for normal personal communication. If someone knows your secrets, do not communicate with that person using this email address. If your life depends on it, change your secret email address often as well as between providers.

Finally, once you have your whole email set up to protect your identity, vanity is your worst enemy. You need to avoid being distinct. Don't try to be clever, flamboyant or unique. Even the way you break your paragraphs is valuable data for identification, especially these days when every school essay and blog post you have written is available in the Internet. Powerful organizations can actually use these texts to build up a database that can “fingerprint” writing.

4.3 File Sharing

The term *File Sharing* refers to the practice of sharing files on a network, often with widest possible distribution in mind. Unfortunately in recent years the term has come to be popularly associated with the distribution of content registered under certain copy-right licenses that disallow the distribution of copies (eg. supposed criminal activity). Regardless of this new association, file sharing remains a vital tool for many world wide: from academic groups to scientific networks and open source software communities.

In this book we wish to help you learn to privately distribute files, with other consenting people, without the content of that exchange known to others or the transaction stopped by an external party. Your basic right to anonymity and to not be spied upon protects that. Suspicions that those things *might* have been stolen and are not yours to give does not undermine that same and original right to privacy.

The history of the internet is littered with attacks of different types on publication and distribution nodes, conducted by different means (court order, Distributed Denial of Service attacks). What such events have demonstrated is that if one wants information to be persistently available and robust against attack, it is a mistake to rely upon a single node which can be neutralised.

This has recently been demonstrated by the takedown of the direct download service Megaupload, whose disappearance led to the loss of massive amounts of its users' data, much of it extraneous even to the alleged copyright infringements which formed the pretext for its closure. In similar vein ISPs will often take down web sites containing disputed material merely because it is cheaper for them to do so than to go to court and have a judge decide. Such policies leave the door open to groundless bullying by all manner of companies, organisations and individuals ready and willing to

make aggressive use of legal letters. Both direct download services and ISPs are examples of centralised

structures which cannot be relied upon both because they are a single point of failure for attack, and because their commercial interests are not aligned with those of their users. Spreading files through distribution, decentralising the data, is the best way to defend against such attacks. In the following section two realms of filesharing are profiled. The first are standard p2p technologies whose technical design is determined by the efficiency of the networks in enabling speed of distribution and discovery of content through associated search mechanisms. The second focuses on I2P as an example of a so-called darknet, its design prioritises security and anonymity over other criteria offering a robust, if less resource efficient, path to persistent availability.

The means of sharing files mentioned below are just some examples of the many P2P technologies that were developed since 1999. BitTorrent and Soulseek have very different approaches, both however were designed for easy usability by a wide public and have significant user communities. I2P is of more recent development and has a small user base.

BitTorrent has become the most popular P2P file-sharing system. The controversy that surrounds it nowadays ironically seems to help the community grow, while police, lobbied by powerful copyright holders seize torrent-tracker server hardware and pursue their operators, sometimes to the point of jailing them as in the case of The Pirate Bay.

Soulseek - while it has never been the most popular file-sharing platform, neither did it ever have the ambition. Soulseek focuses on the exchange of music between enthusiasts, underground producers, fans and researchers. The system and the community around it is completely isolated from the Web: Soulseek files can't be linked to. They are kept exclusively on the hard-disks of Soulseek users. The content of the network fully depends on how many members are connected and what they share. Files are transferred only between two users at a time and nobody but those two users are involved. Because of this 'introverted' character - and the specificity of its content - Soulseek has stayed out of sight of legislation and non-pro-copy copyright advocates.

I2P is one of several systems developed to resist censorship (others include FreeNet and Tor) and has a much smaller user community, it is highlighted here because of its inclusion of Bit Torrent functionality within its basic installation. These systems can also be used to provide hidden services, amongst others, enabling you to publish web pages within their environments.

4.3.1 *BitTorrent*

BitTorrent is a peer-to-peer (P2P) protocol that facilitates distribution of data stored across multiple nodes/participants of the network. There are no central servers or hubs, each node is capable of exchanging data with any other node, sometimes hundreds of them simultaneously. The fact that data is exchanged in parts between numerous nodes allows for great download speeds for popular content on BitTorrent networks, making it quickly the de facto P2P file-sharing platform.

If you are using BitTorrent to circulate material of ambiguous legality, you should know that enforcement agents typically collect information on allegedly infringing peers by participating in torrent swarms, observing and documenting the behaviour of other

peers. The large number of users creates a difficulty for the enforcement system simply at the level of scaling up - there simply are not the resources to pursue every user. Any court case will require actual evidence of data transfer between your client and another (and usually evidence of you uploading), it is enough that you provide even part of the file, not the file in its entirety, for a

prosecution to have legs. But if you prefer to lean towards greater caution, you should use a VPN to route your BitTorrent traffic, as detailed in the **Using VPN** chapter.

Leeching (downloading) of a file from BitTorrent network begins with a *torrent file* or *magnet link*. A torrent file is a small file containing information on the larger files you want to download. The torrent file tells your torrent client the names of the files being shared, a URL for the *tracker* and a *hash* code, which is a unique code representing, and derived from, the underlying file - kind of like an ID or catalog number. The client can use that hash to find others seeding (uploading) those files, so you can download from their computers and check the authenticity of the chunks as they arrive.

A *Magnet Link* does away with the need for a torrent file and is essentially a hyperlink containing a description for that torrent, which your torrent client can immediately use to start finding people sharing the file you are willing to download. Magnet links don't require a tracker, instead they rely on *Distributed Hash Table (DHT)* - which you can read more about in the Glossary - and *Peer Exchange*. Magnet links do not refer to a file by its location (e.g. by IP addresses of people who have the file, or URL) but rather defines search parameters by which this file can be found. When a magnet link is loaded, the torrent client initiates an availability search which is broadcast to other nodes and is basically a shout-out "who's got anything matching this hash?!". Torrent client then connects to the nodes which responded to the shout-out and begins to download the file. BitTorrent uses encryption to prevent providers and other man-in-the-middle from blocking and sniffing your traffic based on the content you exchange. Since BitTorrent swarms (flocks of seeders and leechers) are free for everyone to join it is possible for anyone to join a swarm and gather information about all connected peers. Using magnet links will not prevent you from being seen in a swarm; any of the nodes sharing the same file must communicate between each-other and thus, if just one of the nodes in your swarm is rogue, it will be able to see your IP address. It will also be able to determine if you are seeding the data by sending your node a download request.

One important aspect of using BitTorrent is worth a special mention. Every chunk of data that you receive (leech) is being instantly shared (seeded) with other BitTorrent users. Thus, a process of downloading transforms into a process of (involuntary) publishing, using a legal term - *making available* of that data, before the download is even complete. While BitTorrent is often used to re-distribute freely available and legitimate software, movies, music and other materials, its "making available" capacity created a lot of controversy and led to endless legal battles between copyright holders and facilitators of BitTorrent platforms. At the moment of writing this text, the co-founder of *The Pirate Bay* Gottfrid Svartholm is being detained by Swedish police after an international warrant was issued against him.

For these reasons, and a public relations campaign by copyright holders, use of BitTorrent platforms has become practically analogous to piracy. And while the meaning

of terms such as piracy, copyright and ownership in digital context is yet to be settled, many ordinary BitTorrent users have been already prosecuted on the basis of breaking copyright laws.

Most torrent clients allow you to block IP addresses of known copyright trolls using blacklists. Instead of using public torrents one can also join closed trackers or use BitTorrent over VPN or Tor.

In situations when you feel that you should be worried about your BitTorrent traffic and it's anonymity go through the following check-list:

- Check if your torrent client supports peer-blacklists.
- Check if the peer-blacklist definitions are updated on a daily basis.
- Make sure your client supports all recent protocols - DHT, PEX and Magnet links.

- Choose a torrent client that supports encrypted peers and enable it.
- Upgrade or change your torrent client if any of the above mentioned options is not available.
- Use VPN connection to disguise your BitTorrent traffic from your ISP. Make sure your VPN provider allows P2P traffic. See more tips and recommendations in Using VPN chapter.
- Do not leech and seed stuff you don't know much about.
- Be suspicious of high ratings and overly-positive comments regarding particular torrent link.

4.3.2 SoulSeek

As a peer to peer (P2P) file sharing program, the content available is determined by the users of the Soulseek client, and what files they choose to share. The network has historically had a diverse mix of music, including underground and independent artists, unreleased music, such as demos and mix-tapes, bootlegs, etc. It is entirely financed by donations, with no advertising or user fees.

“Soulseek does not endorse nor condone the sharing of copyrighted materials. You should only share and download files which you are legally allowed to, or have otherwise received permission to, share.” (<http://www.soulseekqt.net>)

Soulseek network depends on a pair of central servers. One server supports the original client and network, and the other supporting the newer network. While these central servers are key to coordinating searches and hosting chat rooms, they do not actually play a part in the transfer of files between users, which takes place directly between the users concerned.

Users can search for items; the results returned being a list of files whose names match the search term used. Searches may be explicit or may use wildcards/patterns or terms to be excluded. A feature specific to the Soulseek search engine is the inclusion of the folder names and file paths in the search list. This allows users to search by folder name.

The list of search results shows details, such as the full name and path of the file, its size, the user who is hosting the file, together with that users' average transfer rate, and, in the case of mp3 files, brief details about the encoded track itself, such as bit rate, length, etc. The resulting search list may then be sorted in a variety of ways and individual files (or folders) chosen for download.

Unlike BitTorrent, Soulseek does not support multi-source downloading or “swarming” like other post-Napster clients, and must fetch a requested file from a single source.

While the Soulseek software is free, a donation scheme exists to support the programming effort and cost of maintaining the servers. In return for donations, users are granted the privilege of being able to jump ahead of non-donating users in a queue when downloading files (but only if the files are not shared over a local area network). The Soulseek protocol search algorithms are not published, as those algorithms run on the server. However several Open Source implementations of server and client software exist for Linux, OS X and Windows.

Regarding privacy and copyright issues Soulseek stand quite far away from BitTorrent too. Soulseek has been taken to court only once, in 2008, but even that did not go anywhere. There are no indications of Soulseek users ever being brought to court or accused of illegal distribution of copyrighted materials or any other ‘digital-millennium’ crimes.

If you want to use the Soulseek network with some degree of real anonymity, you will need to use it over a VPN.

4.3.3 I2P

I2P began as a fork from the Freenet project, originally conceived as a method for censorship-resistant publishing and distribution. From their website:

The I2P project was formed in 2003 to support the efforts of those trying to build a more free society by offering them an uncensorable, anonymous, and secure communication system. I2P is a development effort producing a low latency, fully distributed, autonomous, scalable, anonymous, resilient, and secure network. The goal is to operate successfully in hostile environments

- even when an organization with substantial financial or political resources attacks it.

All aspects of the network are open source and available without cost, as this should both assure the people using it that the software does what it claims, as well as enable others to contribute and improve upon it to defeat aggressive attempts to stifle free speech. (<http://www.i2p2.de/>)

For a guide to installing the software and configuring your browser see section on Secure Filesharing - Installing I2P. Once complete, on launch you will be brought to a console page containing links to popular sites and services. In addition to the usual webpages (referred to as eePsites) there are a range of applications services available ranging from the blogging tool Syndie to a built in BitTorrent client which functions through a web interface.

5 Secure Calls And Sms

5.1 Secure Calls

Phone calls made over the normal telecommunications system have some forms of protection from third party interception, i.e. GSM mobile phones calls are encrypted. GSM calls are not encrypted end-to-end however and telephone providers are increasingly forced to give governments and law enforcement organisations access to your calls. In addition to this the encryption used in GSM has been cracked and now anyone with enough interest and capital can buy the equipment to intercept calls. A GSM Interceptor (<http://en.intercept.ws/catalog/2087.html>) is an off the shelf device to record mobile phone conversations when in the vicinity of the call. Centralised or proprietary systems like Skype also encrypt calls but have built in backdoors for secret services and governments and are at the behest of their owner (in Skype's case Microsoft). Additionally, there are a whole classification of devices called IMSI catchers which can further gather information about mobile phones, including the content of your communication.

However, there are a variety of tools you can use to secure your phone using end-to-end encryption.

5.1.1 iOS - Installing Signal

From the makers of TextSecure is a free and open source tool named Signal. <https://itunes.apple.com/us/app/signal-private-messenger/id874139669?mt=8> Signal uses similar same encryption methods as SilentCircle but provides their service with free and using open

source tools. Additionally, the GUI is extremely easy to use. Signal will transparently detect if you are calling a fellow Signal user and ask if you wish to make a “secure call” (with Signal) or “insecure call” (without end-to-end encryption).

5.1.2 Android - Installing RedPhone

Also from the makers of Signal, there is a free and open source tool named Redphone. <https://play.google.com/store/apps/details?id=org.thoughtcrime.redphone&hl=en> Again, RedPhone uses the similar encryption methods as SilentCircle but provides their service with free and using open source tools. Again, the GUI will transparently detect if you are calling a fellow Signal or RedPhone user and ask if you wish to make a “secure call” (with RedPhone) or “insecure call” (without end-to-end encryption). Unfortunately, RedPhone requires the Google Play framework so it will not work on phones without it (stock Cyanogenmod or similar ROMs).

5 Secure Calls And Sms

5.2 Secure Messaging

SMS are short messages sent between mobile phones. The text is sent without encryption and can be read and stored by mobile phone providers and other parties with access to the network infrastructure to which you’re connected. To protect your messages from interception you need to use end-to-end encryption on your text messages.

5.2.1 Android

- **TextSecure** - WhisperSystems provide an SMS encryption system for Android called TextSecure, based on public key cryptography which ensures that messages are encrypted on the wire and are also stored in an encrypted database on the device, however to ensure encryption on the wire, both parties must be using the application. It is [Open Source](#) and available through the [Play Store](#)

The encryption technology behind it (named //axolotl//) extends the OTR protocol so that messages can be encrypted and send even if not all of the communicating parties are online.

6 Basic Email Security

6.1 Start Using Thunderbird



Figure 6.1:Thunderbird

In upcoming sections, we will be using Mozilla’s Thunderbird e-mail program to show you how to configure your e-mail client for maximum security. Similar to Mozilla’s Firefox

browser, Thunderbird has many security advantages over its counterparts like Apple Mail and Outlook.

Thunderbird is a so-called “mail user agent” (MUA). This is different from web-based e-mail services like Google’s Gmail. You must install the Thunderbird application on your computer. Thunderbird has a nice interface and features that enable you to manage multiple mailboxes, organize messages into folders, and search through mails easily.

Thunderbird can be configured to work with your existing e-mail account, whether that account is through your Internet Service Provider (such as Comcast) or through an web-based email provider (such as Gmail).

Using Thunderbird has many advantages over using web-based e-mail interfaces. These will be discussed in the following chapter. To summarize, though, Thunderbird enables much greater privacy and security than web-based e-mail services.

This section provides information on how to install Thunderbird on Windows, Mac OS X, and Ubuntu.

6.1.1 Installing Thunderbird on Windows

Installing Thunderbird involves two steps: first, download the software and then run the installation program.

1. Use your web browser to visit the Thunderbird download page at <http://www.mozillamessaging.com/en-US/thunderbird/>. This page detects your computer’s operating system and language, and recommends the best version of Thunderbird for you to use.

If you want to use Thunderbird in a different language or with a different operating system, click the *Other Systems and Languages* link on the right side of the page and select the version that you need.



Figure 6.2:Thunderbird Install

2. Click the download button to save the installation program to your computer.

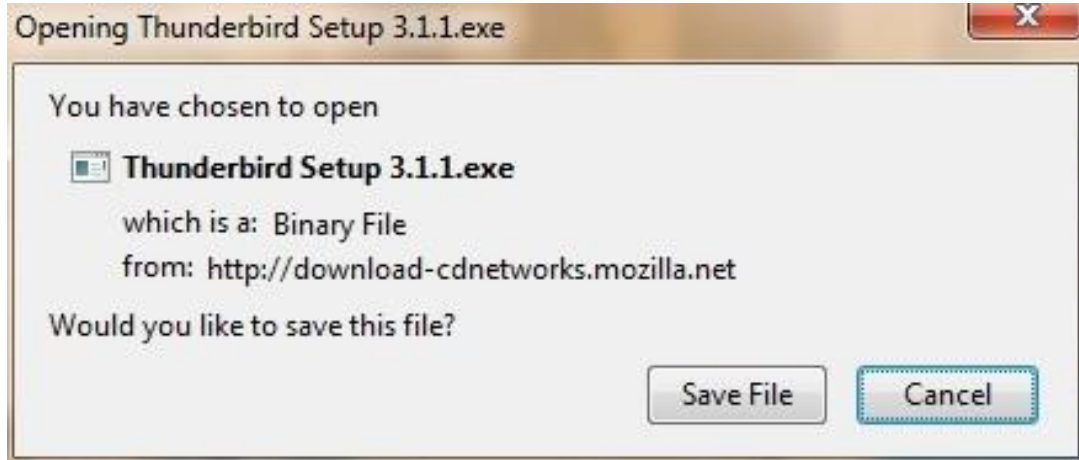


Figure 6.3:Thunderbird Install

- Click the **Save** button to save the Thunderbird Setup file to your computer. 3.Close all applications running on your computer.
- 4.Find the setup file on your computer (it's usually in the Downloads folder or on your desktop) and then double-click it to start the installation. The first thing that the installer does is display the **Welcome to the Mozilla Thunderbird Setup Wizard** screen.



Figure 6.4:Thunderbird Install

Click the **Next** button to start the installation. If you want to cancel it, click the **Cancel** button.

5. The next thing that you see is the **Setup Type** screen. For most users the Standard setup option is good enough for their needs. The Custom setup option is recommended for experienced users only. Note that Thunderbird installs itself as your default mail application. If you do not want this, clear the checkbox labeled **Use Thunderbird as my default mail application**.

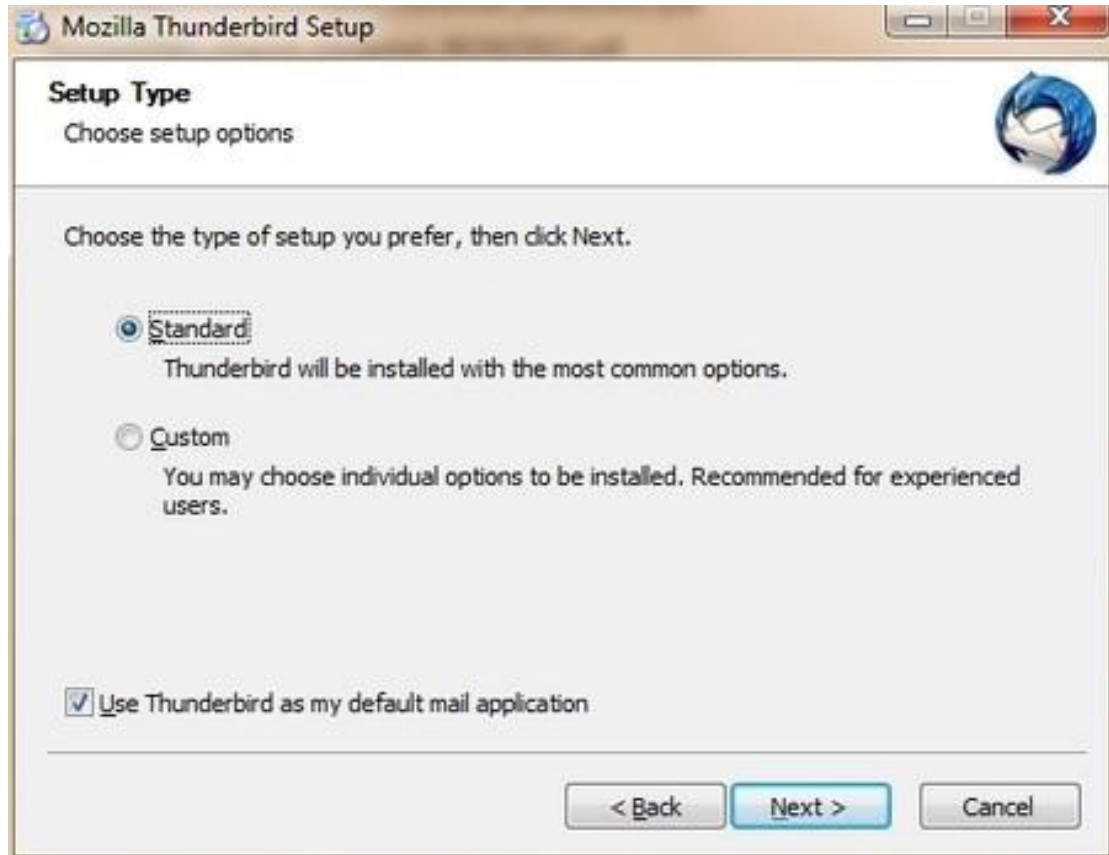


Figure 6.5:Thunderbird Install Click

the **Next** button to continue the installation.

6. After Thunderbird has been installed, click the **Finish** button to close the setup wizard.

If the **Launch Mozilla Thunderbird** now checkbox is selected, Thunderbird starts after it has been installed.

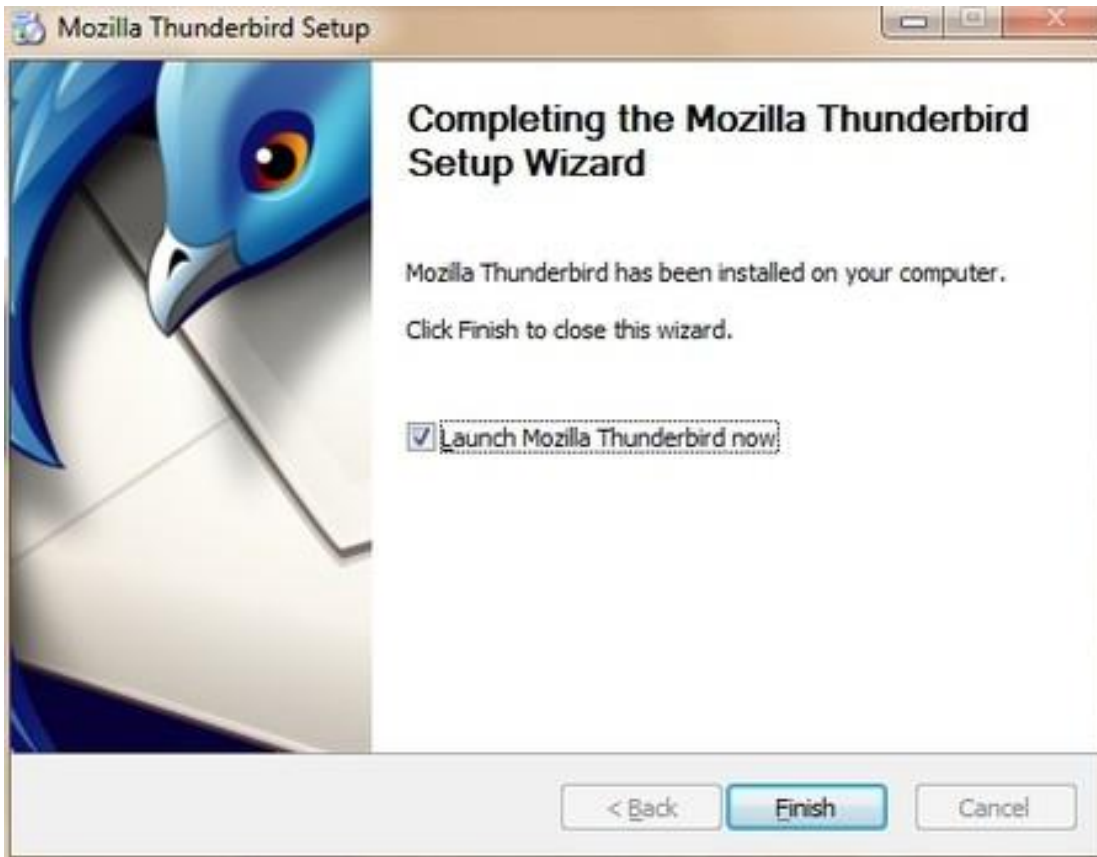


Figure 6.6:Thunderbird Install

6.1.2 Installing Thunderbird on Ubuntu

There are two different procedures for installing Thunderbird on Ubuntu: one for version 10.04 or later, and one for earlier versions of Ubuntu. We describe both below.

Thunderbird will not run without the following libraries or packages installed on your computer:

- GTK+ 2.10 or higher
- GLib 2.12 or higher
- Pango 1.14 or higher
- X.Org 1.0 or higher

Mozilla recommends that a Linux system also has the following libraries or packages installed:

- NetworkManager 0.7 or higher
- DBus 1.0 or higher
- HAL 0.5.8 or higher
- GNOME 2.16 or higher

6.1.3 Installing Thunderbird on Ubuntu 12.04 or newer

If you're using Ubuntu 12.04 or newer, the easiest way to install Thunderbird is through the Ubuntu Software Center.

1. Type Software in the Untiy search window.



Figure 6.7:Thunderbird Install

2.Click on ‘Ubuntu Software Center’

3.Type “Thunderbird” in the search box and press the Enter on your keyboard. The Ubuntu Software Center finds Thunderbird in its list of available software.

4.Click the **Install** button. If Thunderbird needs any additional libraries, the Ubuntu Software Center alerts you and installs them along with Thunderbird.

You can find the shortcut to start Thunderbird in the Internet option under the Applications menu:



Figure 6.8:Thunderbird Install

6.1.4 Installing Thunderbird on Mac OS X

To install Thunderbird on your Mac, follow these steps:

1. Use your web browser to visit the Thunderbird download page at <http://www.mozilla.com/en-US/thunderbird/>. This page detects your computer's operating system and language, and it recommends the best version of Thunderbird for you to use.
2. Download the Thunderbird disk image. When the download is complete, the disk image may automatically open and mount a new volume called *Thunderbird*.



Figure 6.9:Thunderbird Install

If the volume did not mount automatically, open the Download folder and double-click the disk image to mount it. A Finder window appears:

3. Drag the Thunderbird icon into your Applications folder. You've installed Thunderbird!
4. Optionally, drag the Thunderbird icon from the Applications folder into the Dock. Choosing the Thunderbird icon from the Dock lets you quickly open Thunderbird from there.

Note: When you run Thunderbird for the first time, newer versions of Mac OS X (10.5 or later) will warn you that the application Thunderbird.app was downloaded from the Internet. If you downloaded Thunderbird from the Mozilla site, click the **Open** button.



Figure 6.10:Thunderbird Install



Figure 6.11:Thunderbird Install



Figure 6.12:Thunderbird Install

6.1.5 Starting Thunderbird for the first time

After you have installed Thunderbird for the first time you will be guided through the configuration of your mail account. These settings are defined by your e-mail provider (your Internet Service Provider or web-based e-mail service provider). The next chapter describes how to set up your account and configure it for maximum security.

6.2 Setting up secure connections

There is a right (secure) way to configure your connection to your provider's mail servers and a wrong (insecure) way. The most fundamental aspect of e-mail security is the type of connection that you make to your e-mail provider's mail server.

Whenever possible, you should connect using the **SSL** (Secure Socket Layer) and **TLS** (Transport Layer Security) protocols. (**STARTTLS**, which is another option available when configuring an account, is a variation of SSL / TLS.) These protocols prevent your own system (beyond Thunderbird) and any points between your system and the mail server from intercepting and obtaining your password. SSL / TLS also prevent eavesdroppers from reading the content of your messages.

These protocols, however, only secure the connection between your computer and the mail server. They do not secure the information channel all the way to the message recipient. Once the mail servers forward the message for delivery, the message may be intercepted and read by points in between the mail server and the recipient.

This is where **PGP** (Pretty Good Privacy) comes in, which is described in the next chapter.

The first step in establishing e-mail security is a secure connection between your system and the mail servers. This chapter describes how to set up your e-mail account the right way.

6.2.1 Configuration requirements

When you configure an account, Thunderbird attempts to determine (from the email account and the account details that you provide) the connection parameters to your email provider. While Thunderbird knows the connection parameters for many email providers, it does not know them all. If the parameters are not known to Thunderbird, you will need to provide the following information to configure your account:

- **Your username**
- **Your password**
- **Incoming server:** name (such as `imap.example.com`), protocol (POP or IMAP), port (by default, 110), and security protocol
- **Outgoing server:** name (such as `smtp.example.com`), port (by default, 25), and security protocol

You should have received this information from your hosting provider. Alternatively, you can usually find this information on the support pages on the website of your hosting provider. In our example we will be using the Gmail server configuration. You can use Thunderbird with your Gmail account. To do so, you must change a configuration setting in your account. If you are not using a Gmail account, skip the next section.

6.2.2 Preparing a Gmail account for use with Thunderbird

Log in to your Gmail account in your browser. Select **Settings** from options in the top right, then go to the tab **Forwarding and POP/IMAP**. Click **Enable IMAP** and then **Save Changes**.



Figure 6.13:Gmail enable IMAP

6.2.3 Configuring Thunderbird to use SSL/TLS

When you start up Thunderbird for the first time, you will enter a step-by-step configuration procedure for setting up your first account. (You can invoke the account setup interface any time by selecting **File | New | Mail Account**). On the first screen, you will be asked for your name, your email-address and your password. The value you enter for your name does not have to be your real name. It will be shown to the recipient of your messages. Enter the information and click **Continue**.

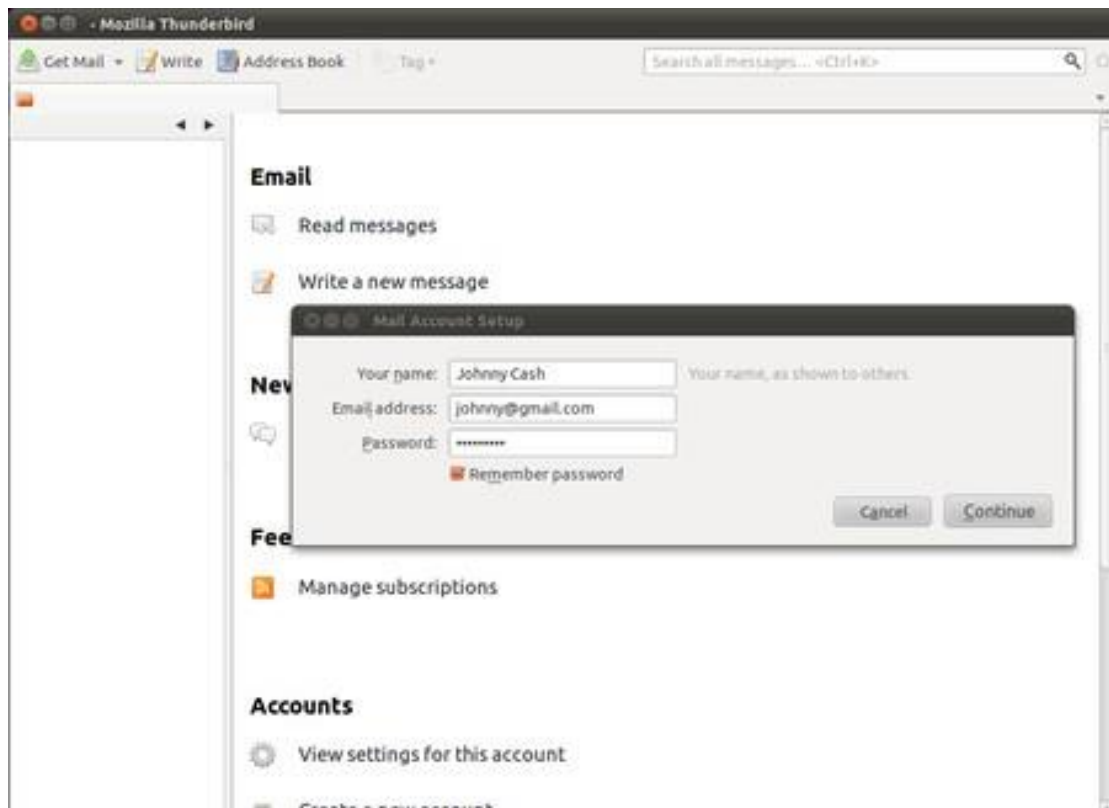


Figure 6.14:Thunderbird Configure

On the next screen, Thunderbird will attempt to determine the server names based on your email address. This may take some time, and will only work if Thunderbird knows the settings for the mail servers for your email provider. In either case you will be presented with a window where you can modify the settings. In the example below, Thunderbird has detected the settings automatically. You can see the protocol at the right side of the server names. This should be either **SSL/TLS** or **STARTTLS**. *Otherwise your connection is insecure and you should attempt manual setup.*

When you are finished, click **Create account**. If Thunderbird could not determine your server settings, click on **Manual setup** to configure the server names yourself.

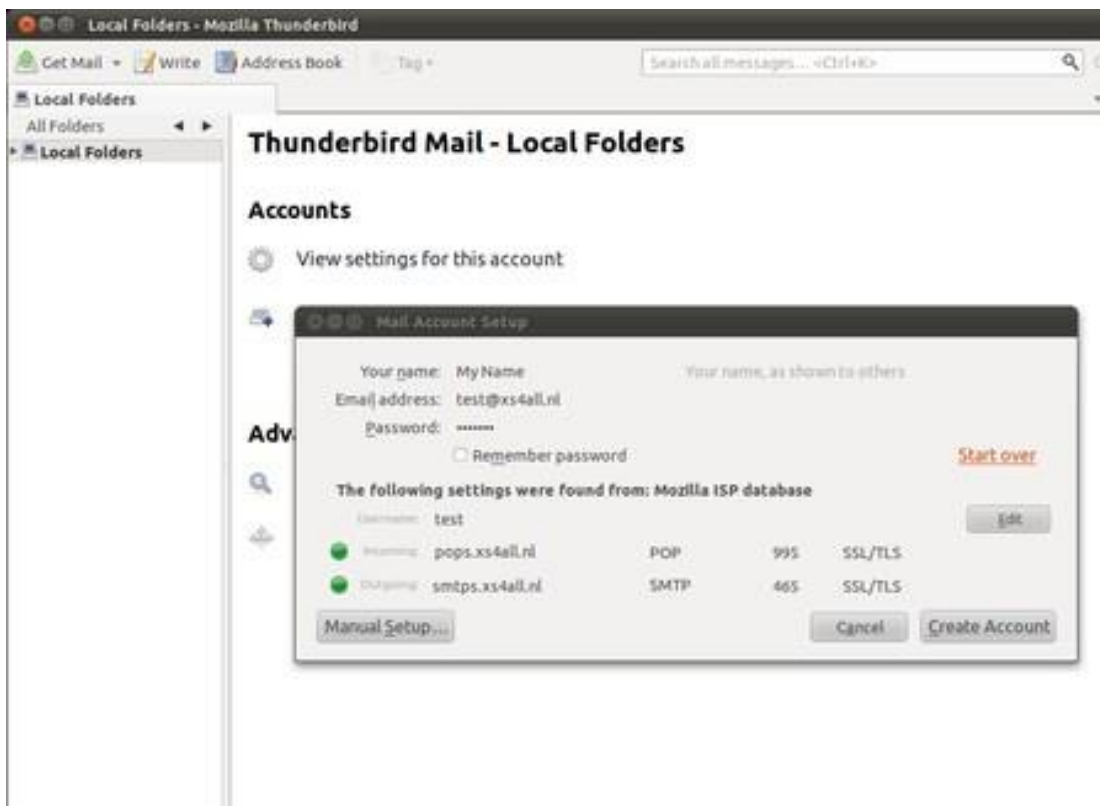


Figure 6.15:Thunderbird Install

6.2.4 Manual setup

Use the Account Settings interface to manually configure accounts in Thunderbird. The Account Settings dialog will automatically open if you select **Manual setup** in the configuration wizard. In this case we are only interested in the incoming and outgoing mail server names, and the protocol we use to connect with them. As you can see in the examples below, we enter the Gmail server names and we force them to use **TLS/SSL**, a secure method to connect to the servers.

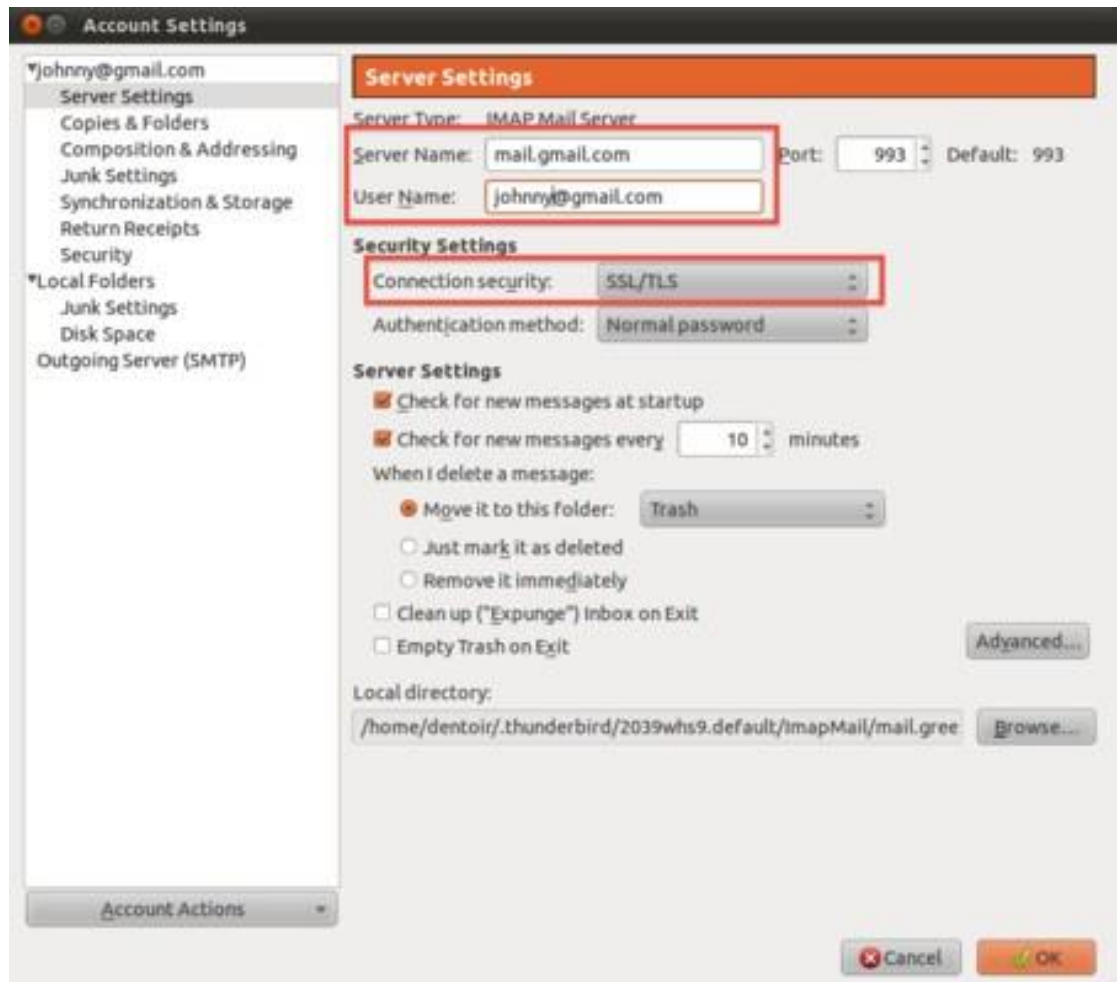


Figure 6.16:Thunderbird Install

Under 'Server Settings', we will find only the incoming (**IMAP**) server and its settings for that specific account.

After **Server Name** enter the name of the IMAP server, in this case mail.gmail.com. As you can see we have selected '**SSL/TLS**' under the connection security setting. This enforces encryption. Do not be scared by the authentication method **Normal**

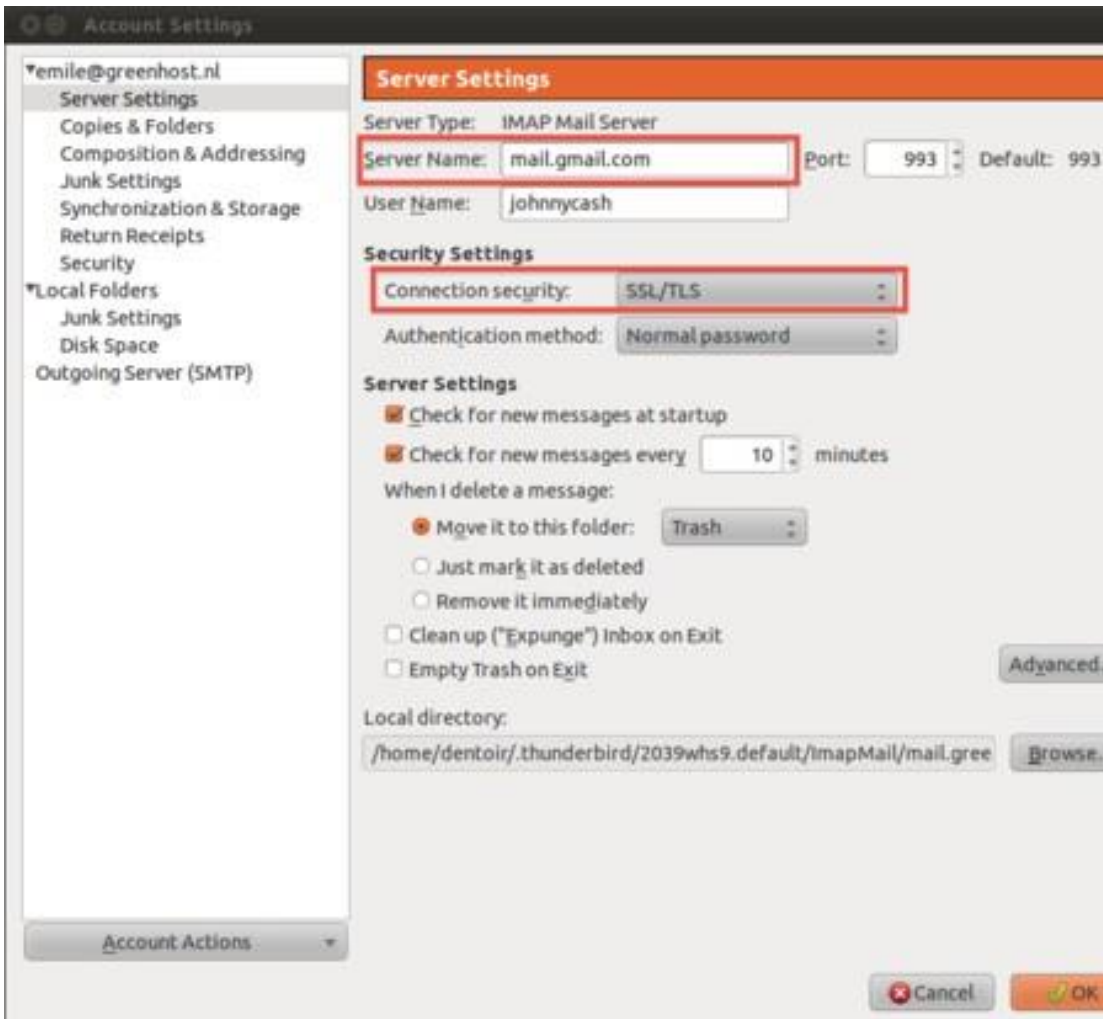


Figure 6.17:Thunderbird Install

password. The password will be automatically encrypted due to our secured connections to the server.

Finally, configure the outgoing server for the account. Click on **Outgoing Server (SMTP)** in the left panel.

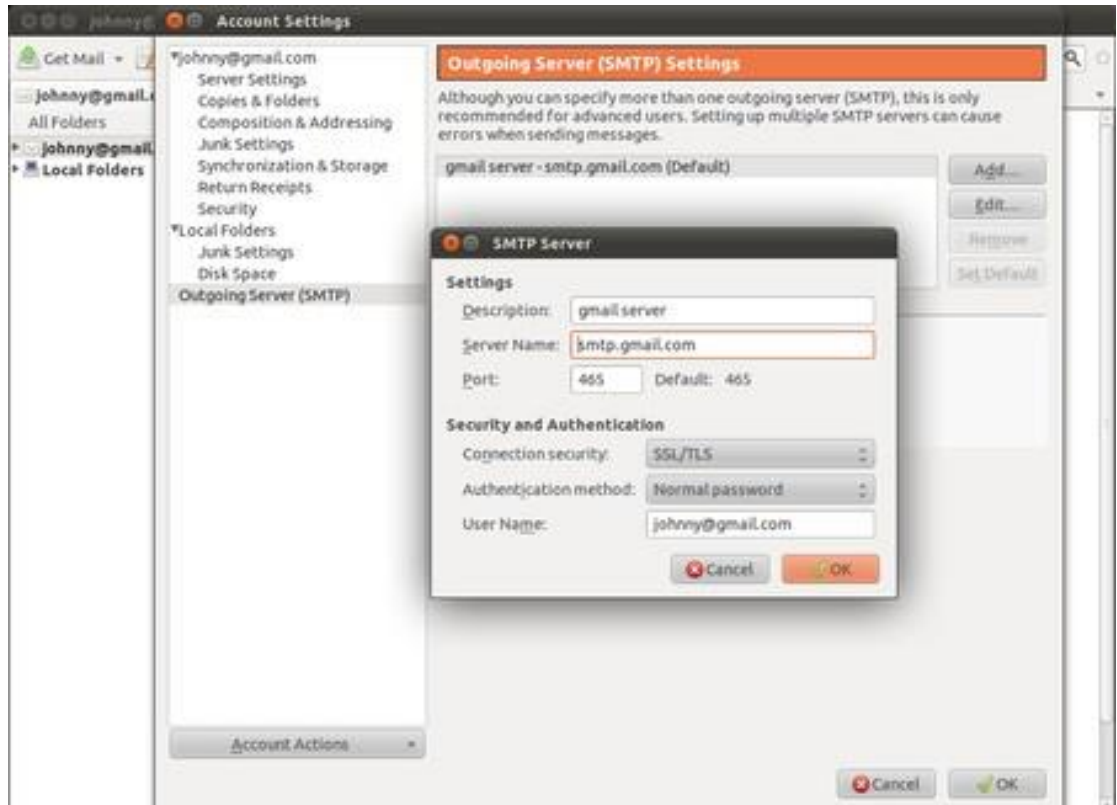


Figure 6.18:Thunderbird Install

Again, we have selected **SSL/TLS** under **Connection security**. The port will default to 465 and this should generally not have to be changed.

6.2.5 Finishing the setup, different encryption methods

Test your Thunderbird setup by trying to send and receive mails. Some email hosting providers may not support the SSL/TLS protocol, which is the preferred choice. You will get an error message saying the authentication protocol is not supported by the server. You may then switch to using STARTTLS instead. In the above two screens, select 'STARTTLS' under 'Connection security'. If this method also fails, contact your email hosting provider and ask them if they provide another way to securely connect to their servers. If they do not allow you to securely connect to their servers, then you should complain and seriously consider switching to a different provider.

6.2.6 Returning to the configuration screens

At any time you can reconfigure your email accounts by going to the Thunderbird menu bar and clicking **Edit | Account Settings** (Linux), **Tools | Account Settings** (Windows and Mac OS X).

6.3 Some Additional Security Settings

Thunderbird provides additional security measures to protect you from junk mail, identity theft, viruses (with the help of your anti-virus software, of course), intellectual property theft, and malicious web sites.

We will look at the following Thunderbird security features. First a little background on why you need to consider some of these measures:

- **Adaptive junk mail controls.** Adaptive junk mail controls allow you to train Thunderbird to identify junk email (SPAM) and remove it from your inbox. You can also mark messages as junk mail manually if your email provider's system misses the junk mail and lets it go through.
- **Integration with anti-virus software.** If your anti-virus software supports Thunderbird, you can use that software to quarantine messages that contain viruses or other malicious content. If you're wondering what anti-virus software works with Thunderbird, you can find a list here: http://kb.mozillazine.org/Antivirus_software.
- **Master password.** For your convenience, you can have Thunderbird remember each of your individual passwords of your e-mail accounts. You can specify a master password that you enter each time you start Thunderbird. This will enable Thunderbird to open all your email accounts with your saved passwords.
- **Restrictions on cookies.** Some blogs and websites attempt to send cookies (a piece of text that stores information from Web sites on your computer) with their RSS feeds. These cookies are often used by content providers to provide targeted advertising. Thunderbird rejects cookies by default, but you can configure Thunderbird to accept some or all cookies.

In the Security Preferences section of Thunderbird's Options/Preferences dialog box you can set up the preferences for these features.

- In Windows and Mac OS X, go to the 'Tools' menu and click 'Options'.
- On Ubuntu or other versions of Linux, go to the 'Edit' menu and click 'Preferences'.

6.3.1 Junk mail settings

1. In the Preferences/Options dialog box, click 'Security' and then click the 'Junk' tab.

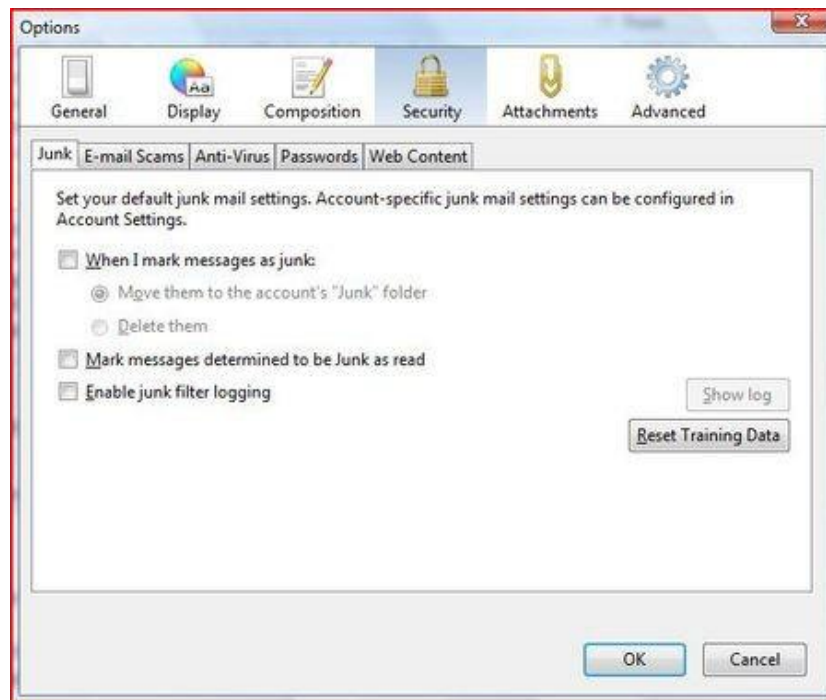


Figure 6.19:Thunderbird Security

2.Do the following:

- To tell Thunderbird that it should handle messages marked as junk, select the check box labelled 'When I mark message as junk'.
- To have Thunderbird move these messages to a junk folder, select the 'Move them to account's 'Junk' folder' radio button.
- To have Thunderbird delete junk mail upon receiving it, select the 'Delete them' radio button.

3. Thunderbird will mark junk message as read if you select the check box labeled 'Mark messages determined to be Junk as read'.
4. If you want to keep a log of junk mail received, select the 'Enable junk filter logging' check box.
- 5.Click the 'OK' button to close the 'Options/Preferences' dialog box.

6.3.2 Scam detection and warning system

1. In the Preferences/Options dialog box, click 'Security' and then click the 'E-mail Scams' tab.

2.To have Thunderbird warn you about possible email scams, select the check box

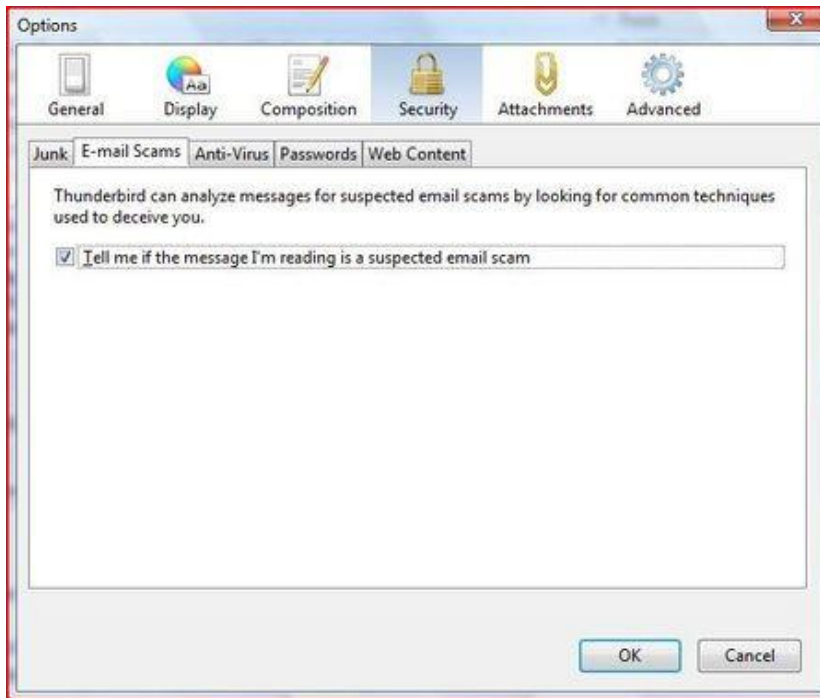


Figure 6.20:Thunderbird Security

labelled 'Tell me if the message I'm read is a suspected email scam'. To turn off this feature, deselect this check box.

3.Click the 'OK' button to close the 'Options/Preferences' dialog box.

6.3.3 Anti-virus integration

1.In the Preferences/Options dialog box, click 'Security' and then click the 'Anti- Virus' tab.

2.To turn on anti-virus integration, select the check box labeled 'Allow anti-virus clients to quarantine individual incoming messages'. To turn off this feature, dese- lect this check box.

3.Click the 'OK' button to close the 'Options/Preferences' dialog box.

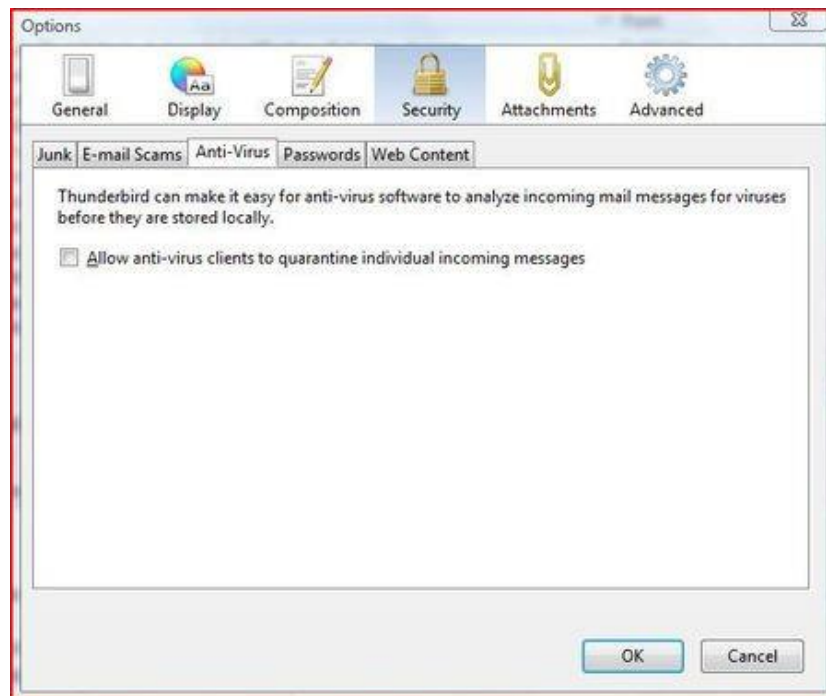


Figure 6.21:Thunderbird Security

6.3.4 Set a master password

1. In the Preferences/Options dialog box, click 'Security' and then click the 'Passwords' tab.
2. Select the check box labeled 'Use a master password'.
3. Enter your password into the 'Enter new password' and 'Re-enter password' fields.
4. Click the 'OK' button to close the Change Master Password dialog box.
5. If you want to see the passwords that you have saved in Thunderbird, click the 'Saved Passwords' button. This will open the 'Saved Passwords' dialog box.
6. To see the passwords, click the 'Show Passwords' button.
7. Click the 'Close' button to close 'Saved Passwords' dialog box.
8. Click the 'OK' button to close the 'Options/Preferences' dialog box.

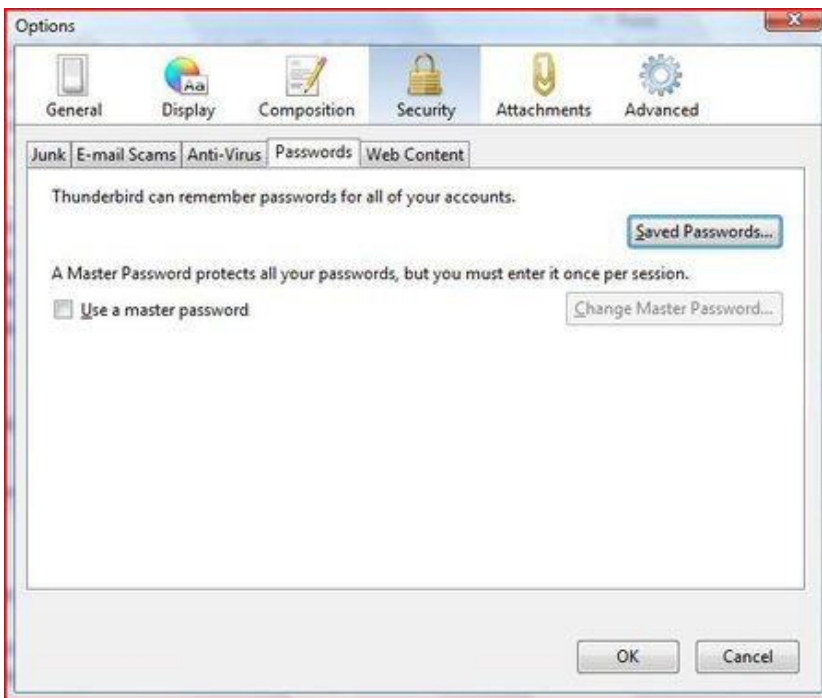


Figure 6.22:Thunderbird Security



Figure 6.23:Thunderbird Security

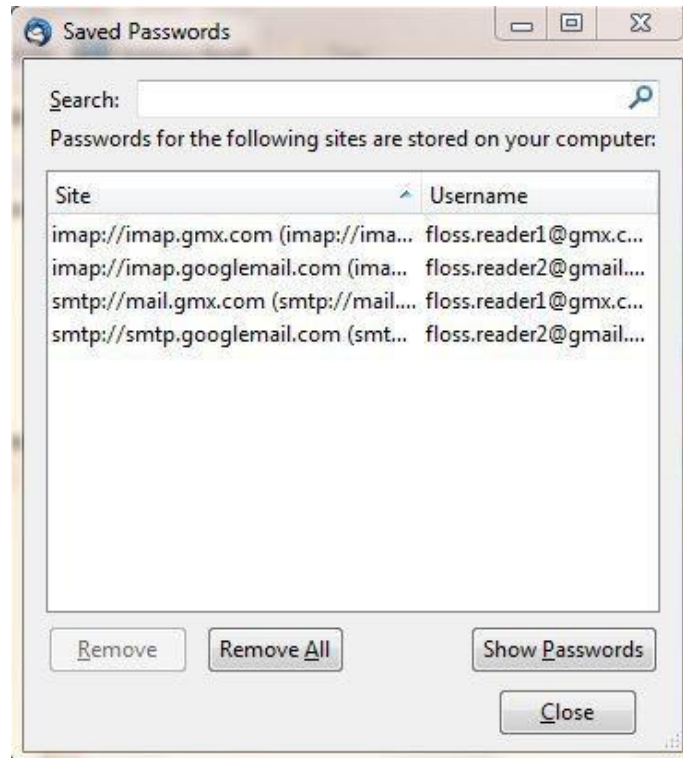


Figure 6.24:Thunderbird Security

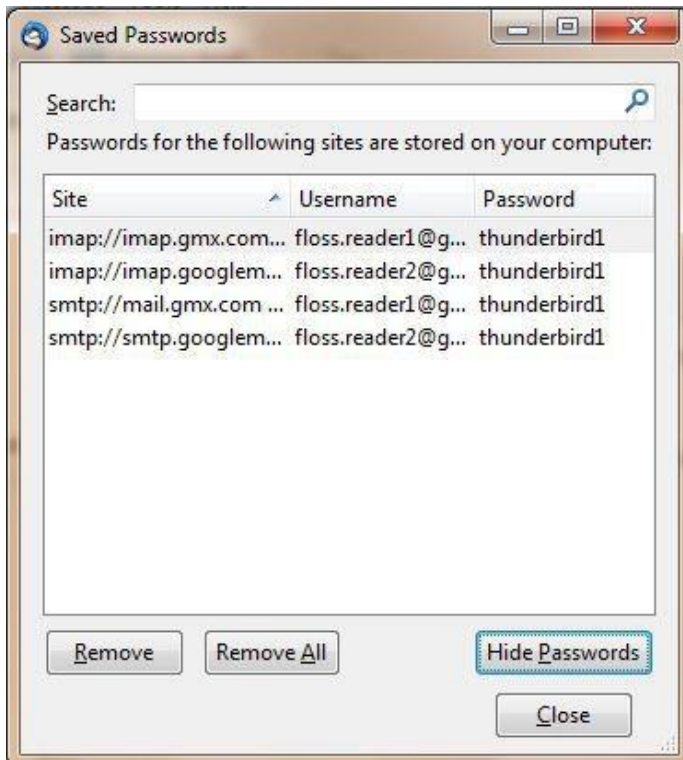


Figure 6.25:Thunderbird Security

6.3.5 Adaptive junk mail controls

You need to first open Account Settings window. Note that settings configured in the Account Settings window apply only to the account that you select in the Folders pane. You must configure local folders separately.

1. In the Folders pane right-click on an account name and select 'Settings'.

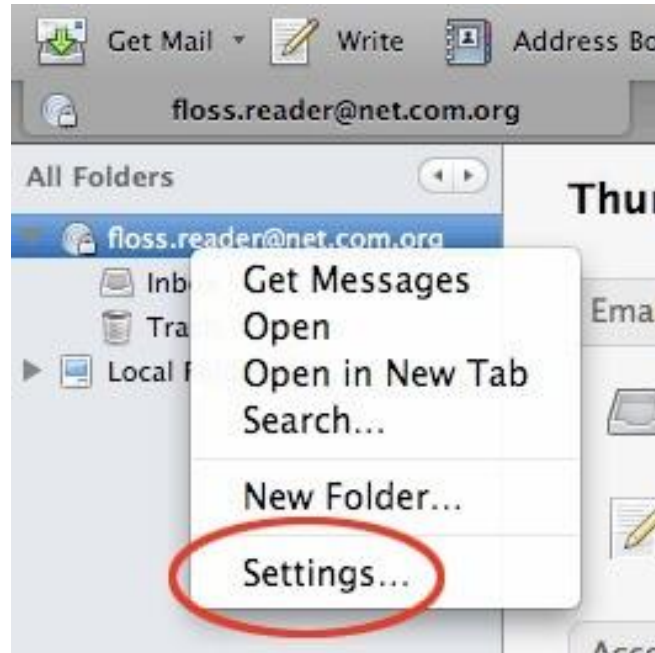


Figure 6.26:Thunderbird Security

2. In Windows or Mac go to the 'Tools' menu and select 'Account Settings'. In Linux, go to the 'Edit menu' and select 'Account Settings'.
3. To set adaptive junk mail controls for a specific account, pick an account and click 'Junk Settings'.
4. To turn on the controls, select the check box labeled 'Enable adaptive junk mail controls for this account'. To turn them off, deselect this check box.
5. If you want the controls to ignore mail from senders in your Address Book, select the check boxes next to any of the listed address books.
6. To use a mail filter such as SpamAssassin or SpamPal, select the check box labelled 'Trust junk mail headers sent by:' and pick a filter from the menu.
7. Select the check box labeled 'Move new junk messages to' if you want to move junk mail to a specified folder. Then select the destination folder to be either at your email provider or a local folder on your computer.

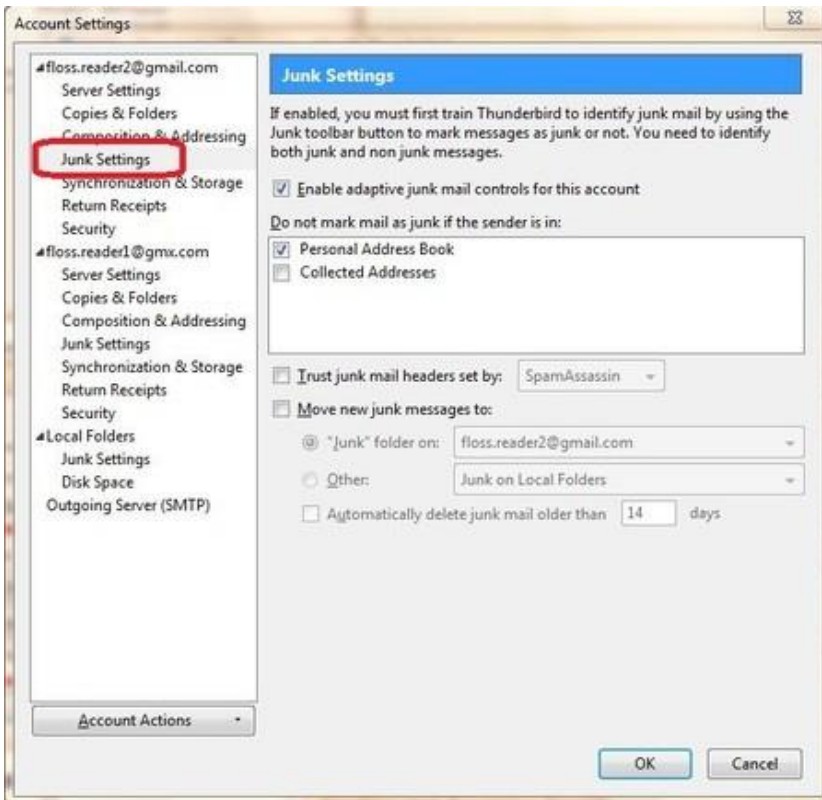


Figure 6.27:Thunderbird Security

8. Select the 'Automatically delete junk mail other 14 days' check box to have Thunderbird regularly remove junk mail. To change the time period for this process, enter a different number (in days) in the text box.
9. Click 'OK' to save your changes.

7 Email Encryption

7.1 Introducing mail encryption (PGP)



Figure 7.1:PGP

This chapter will introduce you to some basic concepts behind mail encryption. It is important to read to get some feeling of how mail encryption actually works and what its caveats and limitations are. **PGP** (Pretty Good Privacy) is the protocol we shall use for e-mail encryption. This protocol allows us to digitally sign and encrypt mail messages. It works on an end-to-end basis: messages will be encrypted on your own computer and will only be decrypted by the recipient of the message.

There is no possibility for a ‘man-in-the-middle’ to decipher the contents of your encrypted message. This *excludes* the subject lines and the ‘from’ and ‘to’ addresses, which unfortunately are not encrypted in this protocol.

After having introduced these basic concepts, the next chapters will give you a hands-on guide to install the necessary tools on your operating system and get encryption up and running. We will focus on using Enigmail which is an extension for Thunderbird that helps you manage PGP encryption for your email. The installation process for Enigmail / PGP is different for Mac OSX, Windows and Ubuntu so please see the appropriate chapters in this section for instructions.

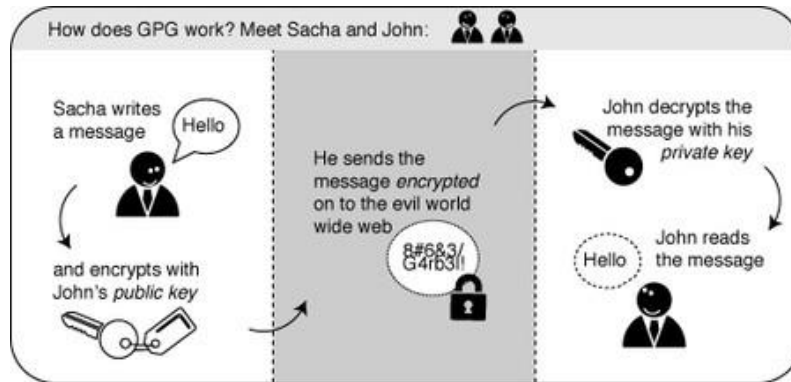


Figure 7.2:GPG Schema

7.1.1 Using a key-pair to encrypt your mail

A crucial concept in mail encryption is the usage of so-called *key-pairs*. A key-pair is just two separate files sitting on your harddisk or USB stick. Whenever you want to encrypt mails for a certain mail-account, you will need to have these files available to yourself in some form. If they are sitting at home on your computer, you will not be able to decrypt mail at the office. Putting them on a USB stick should provide a solution to this problem.

A key-pair consists of the two different keys: a public key and a secret key.

The public key: you can give this key to other people, so they can send you encrypted mails. This file does not have to be kept secret.

The secret key: this basically is your secret file to decrypt emails people send to you. It should *never* be given to someone else.

7.1.2 Sending encrypted mails to other people: you need their public key

I have five colleagues at work and I want to send encrypted mails to them. I need to have public keys for each of their addresses. They can send me these keys using ordinary mail, or they can give them to me in person, or put them on a USB stick, or they can have their keys on a website. It doesn't matter, as long as I can trust those keys really belong to the person I want to correspond with. My software puts the keys on my 'keyring', so my mail application knows how to send them encrypted mails.

7.1.3 Receiving encrypted mails from other people: they need my public key

For my five (or thirty) colleagues to be able to send *me* encrypted mails, the process goes the other way around. I need to distribute my public key to each of them.

7.1.4 Conclusion: encryption requires public key distribution!

All the people in a network of friends or colleagues wanting to send each other encrypted emails, need to distribute their public keys to each other, while keeping their secret keys a closely guarded secret. The software described in this chapter will help you do this key management.

7.2 Installing PGP on Windows

To complicate matters a little - PGP is the protocol used for encrypting e-mail by various softwares. To get PGP to work with Thunderbird we need to install GPG - a free software implementation of PGP *and* Enigmail - an extension of Thunderbird that allows you to use GPG. . . Confused?! Don't worry about it, all you have to know is how to encrypt your email with PGP and you need to install *both* GPG and Enigmail. Here is how to do it. . .

7.2.1 Installing PGP (GPG) on Microsoft Windows

The GNU Privacy Guard (GnuPG) is software which is required to send PGP encrypted or signed emails. It is necessary to install this software before being able to do any encryption.

Head to the website of the Gpg4win project. Go to <http://gpg4win.org/>. On the left side of the website, you will find a 'Download' link. Click on it.



Figure 7.3:GPG Windows

This will take you to a page where you can download the Gpg4Win. Click on the button which offers you the latest stable version (not beta) of Gpg4Win.



Figure 7.4:GPG Windows

This will download you an .exe file. Depending on your browser, you may have to double-click on this downloaded file (named something like gpg4qin-2.1.0.exe) before something happens. Windows will ask you if you are sure you want to install this program. Answer yes.

Then complete the installation by agreeing to the license, choosing appropriate language and accepting the default options by clicking 'Next', unless you have a particular reason not to.

The installer will ask you where to put the application on your computer. The default setting should be fine but make a note of it as we may need this later. Click on 'Next' when you agree.

7.2.2 Installing with the Enigmail extension

After you have successfully installed the **PGP** software as we described above you are now ready to install the **Enigmail** add-on.

Enigmail is a Thunderbird add-on that lets you protect the privacy of your email conversations. Enigmail is simply an interface that lets you use PGP encryption from

within Thunderbird.

Enigmail is based on public-key cryptography. In this method, each individual must generate her/his own personal key pair. The first key is known as the private key. It is protected by a password or passphrase, guarded and never shared with anyone.

The second key is known as the public key. This key can be shared with any of your correspondents. Once you have a correspondent's public key you can begin sending encrypted emails to this person. Only she will be able to decrypt and read your emails, because she is the only person who has access to the matching private key.

Similarly, if you send a copy of your own public key to your e-mail contacts and keep the matching private key secret, only you will be able to read encrypted messages from those contacts.

Enigmail also lets you attach digital signatures to your messages. The recipient of your message who has a genuine copy of your public key will be able to verify that the e-mail comes from you, and that its content was not tampered with on the way. Similarly, if you have a correspondent's public key, you can verify the digital signatures on her messages.

7.2.3 Installation steps

To begin installing Enigmail, perform the following steps:

1. Open **Thunderbird**, then Select Tools > Add-ons to activate the *Add-ons* window; the Add-ons window will appear with the default *Get Add-ons* pane enabled.
2. Enter enigmail in the search bar, like below, and click on the search icon.

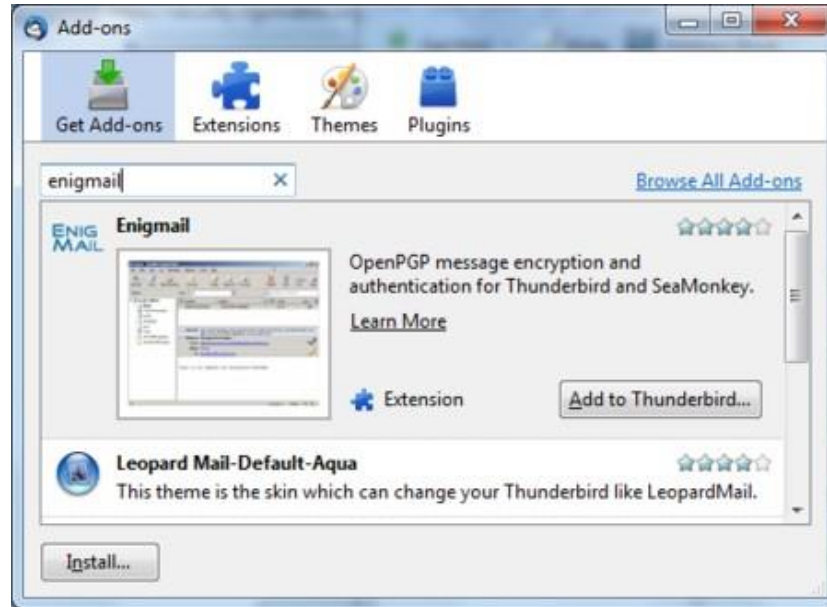


Figure 7.5:Enigmail Install

3. Simply click on the 'Add to Thunderbird' button to start the installation.
4. Thunderbird will ask you if you are certain you want to install this add-on. We trust this application so we should click on the 'Install now' button.

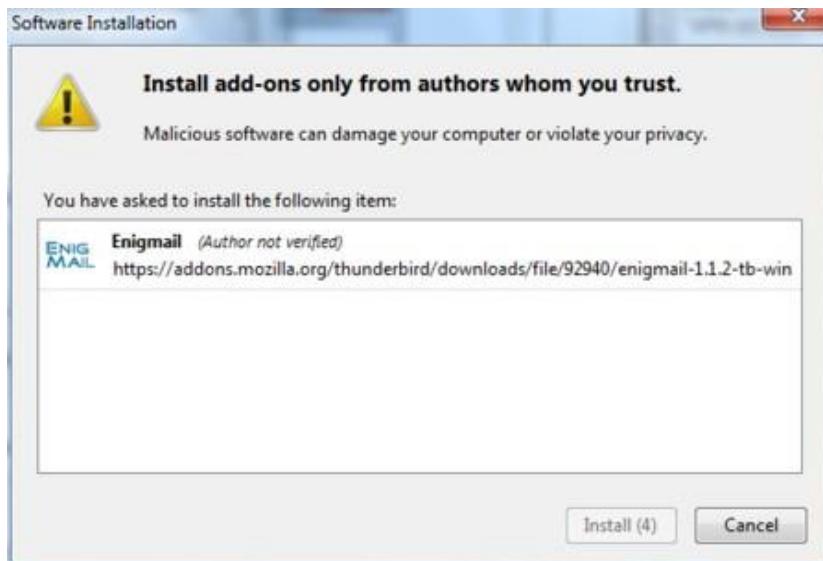


Figure 7.6:Enigmail Install

5. After some time the installation should be completed and the following window should appear. Please click on the 'Restart Thunderbird' button.

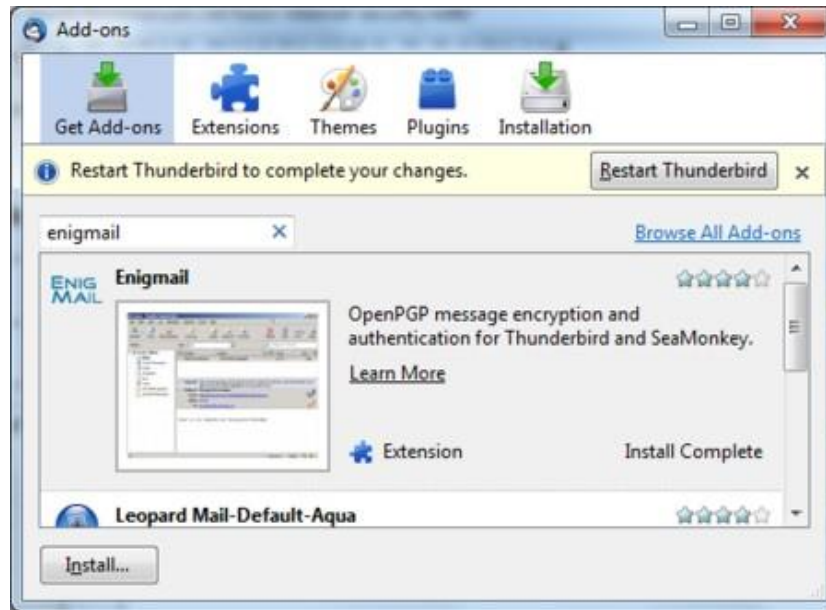


Figure 7.7: Enigmail Install

7.3 Installing PGP on OSX

The GNU Privacy Guard (GnuPG) is software which enables you to send PGP encrypted or signed emails. It is necessary to install this software before being able to do any encryption. This chapter covers the installation steps required to install GnuPG on Mac OSX.

7.3.1 Getting started

For this chapter we assume you have the latest version of:

- OSX installed (10.6.7)
- Thunderbird (3.1.10)

Note on OSX Mail: It is possible to use PGP with the build-in mail program of OSX. But we do not recommend this because this option relies on a hack of the program which is neither open or supported by its developer and breaks with every update of the mail program. So unless you really have no other option we advice you to switch to Mozilla Thunderbird as your default mail program if you want to use PGP.

7.3.2 Downloading and installing the Software

1. For OSX there is a bundle available which will install everything you need in one installation. You can get it by directing your browser to <http://www.gpgtools.org/> and clicking on the big blue disk with “Download GPGTools Installer” written under

it. It will redirect you to another page on <http://www.gpgtools.org/installer/index.html> where you can actually download the software.

(nb. We are using the latest version Firefox for this manual, so the screens might look a little bit different if you are using a different browser)

2. Download the software by choosing 'Save File' and clicking 'OK' in the dialogue.

3. Navigate to the folder where you normally store your downloads (Mostly the desk-top or the downloads folder surprisingly) and double click the '.DMG' file to open the virtual disk containing the installer.

4. Open the installer by double-clicking on the icon.

5. The program will check your computer to see if it can run on the computer.

(Note, if your Mac is bought before 2006 it will not have an intel processor required to run this software and the installation will fail. Sadly it is beyond the scope of this manual to also take into account computers over five years old)

You will be guided by the program through the next steps like accepting the license agreement. But stop pressing all the OK's and Agrees as soon as you come to the 'Installation Type' screen:

6. Clicking 'Customize' will open this screen where you see several options of programs and software to install. You can click on each one of them to get a little bit of information on what it is, what it does and why you might need it.

As said in the intro; we advise against using Apple Mail in combination with PGP. Therefore you won't be needing 'GPGMail', as this enables PGP on Apple Mail, and you can uncheck it.

'**Enigmmail**' on the other hand is very important as it is the component that will enable Thunderbird to use PGP. In the screen shot here it is greyed out as the installer wasn't able to identify my installation of Thunderbird. Since this seems to be a bug. You can also install Enigmmail from within Thunderbird as is explained in another chapter.

If the option is not greyed out in your installation, you should tick it.

After you checked all the components you want to install click 'Install' to proceed. The installer will ask you for your password and after you enter that the installation will run and complete; Hooray!



Figure 7.8:GPG Install

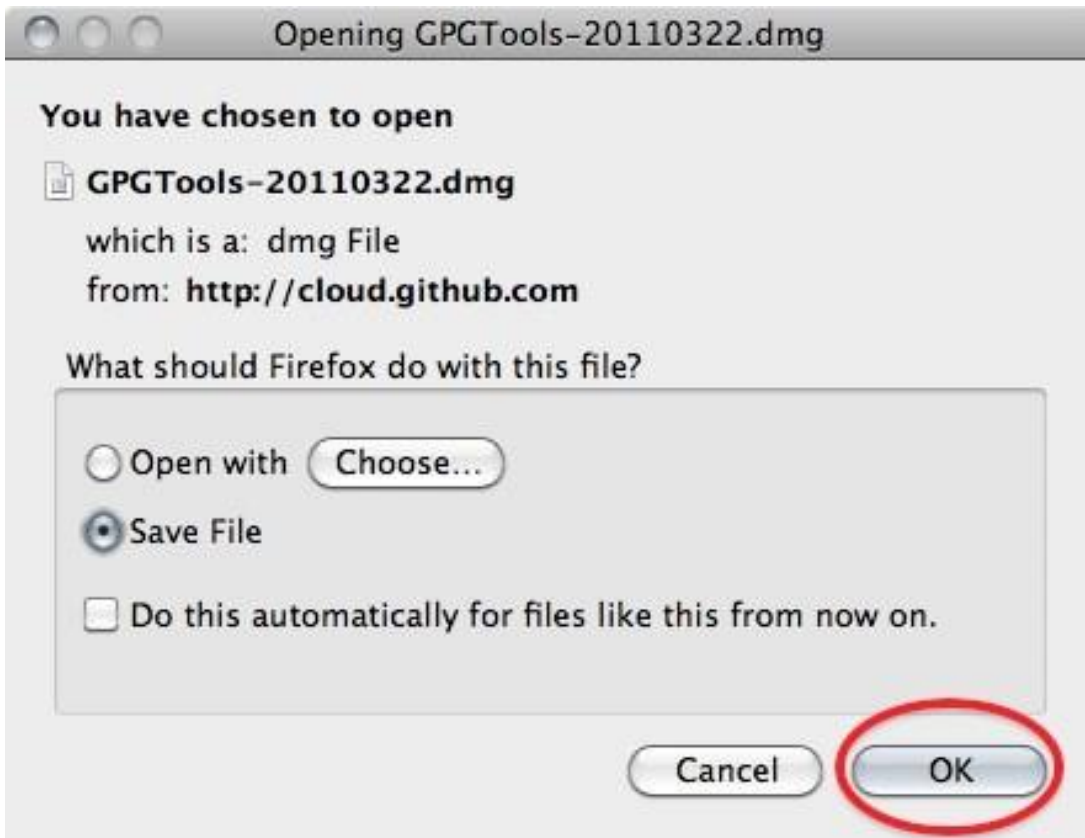


Figure 7.9:GPG Install



Figure 7.10:GPG Install



Figure 7.11:GPG Install



Figure 7.12:GPG Install



Figure 7.13:GPG Install



Figure 7.14:GPG Install



Figure 7.15:GPG Install

7.3.3 Installing up Engimail

1. Open **Thunderbird**, then Select Tools > Add-ons to activate the *Add-ons* window; the Add-ons window will appear with the default *Get Add-ons* pane enabled.

In the Add-On window, you can search for ‘Enigmail’ and install the extension by clicking ‘Add to Thunderbird . . .’

2. After you open the Add-On window, you can search for ‘Enigmail’ and install the extension by clicking ‘Add to Thunderbird . . .’

3. Click on ‘Install Now’ to download and install the extension.

Be aware that you will have to restart Thunderbird to use the functionality of this extension!

Now that you have successfully downloaded and installed Enigmail and PGP you can go on to the Chapter that deals with setting up the software for use.



Figure 7.16:GPG Install

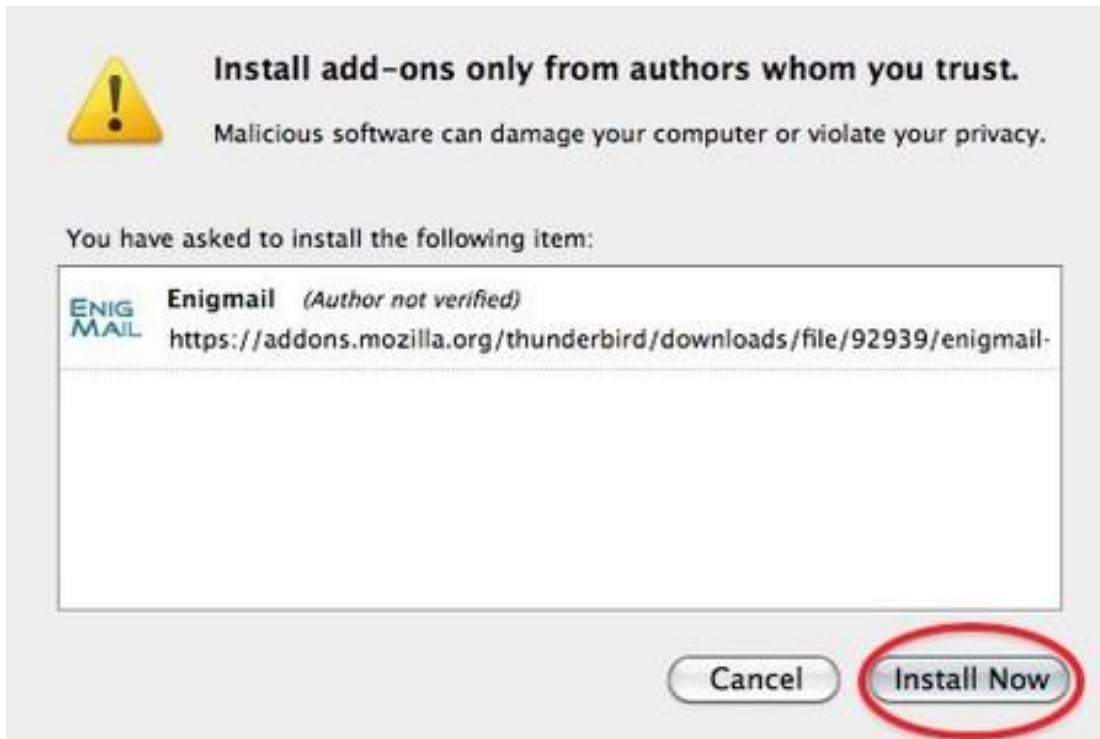


Figure 7.17:GPG Install

7.4 Installing PGP on Ubuntu

We will use the Ubuntu Software Centre for installing PGP (Enigmail and accessories). First open the Ubuntu Software Center through the Unity menu by typing ‘software’ into the Unity search area



Figure 7.18:PGP Install Click on the

‘Ubuntu Software Center’.

Type into the search field ‘Enigmail’ and search results should be returned automatically: Highlight the Enigmail item (it should be highlighted by default) and click ‘Install’ and you will be asked to authenticate the installation process.

Enter your password and click ‘Authenticate’. The installation process will begin.

When the process is completed you get very little feedback from Ubuntu. The progress bar at the top left disappears. The 'In Progress' text on the right also disappears. Enigmail should now be installed.



Figure 7.19:PGP Install

7.5 Installing GPG on Android

With the growing usage of mobile phones for e-mail, it's interesting to be able to use GPG also on your mobile. This way you can still read the messages sent to you in GPG on your phone and not only on your computer.

Install the *Android Privacy Guard (APG)* and *K-9 Mail* applications to your Android device from the Google Play Store or another trusted source.

1. Generate a new private key that uses DSA-Elgamal with your PC's GPG installation (You can only create keys with up to 1024bit key length on Android itself).
2. Copy the private key to your Android device.
3. Import the private key to APG. You may wish to have APG automatically delete the plaintext copy of your private key from your Android device's filesystem.
4. Set-up your e-mail accounts in *K-9 Mail*.
5. In the settings for each account, under *Cryptography*, make sure that *K-9 Mail* knows to use APG. You can also find options here to make *K-9 Mail* automatically sign your messages and/or encrypt them if APG can find a public key for the recipient(s).
6. Try it out.

7.5.1 APG

This is a small tool which makes GPG encryption possible on the phone. You can use APG to manage your private and public keys. The options in the application are quite

straightforward if you have a little knowledge of GPG in general.

Management of keys is not very well implemented yet. The best way is to manually copy all your public keys to the SD card in the APG folder. Then it's easy to import your keys. After you've imported your public and private keys, GPG encrypting, signing and decrypting will be available for other applications as long as these applications have integrated encryption/GPG.

7.5.2 GPG enabled e-mail on Android: K-9 Mail

The default mail application does not support GPG. Luckily there is an excellent alternative: K-9 Mail. This application is based on the original Android mail application but with some improvements. The application can use APG as its GPG provider. Setting up K-9 Mail is straightforward and similar to setting up mail in the Android default mail application. In the settings menu there is an option to enable "Cryptography" for GPG mail signing.

If you want to access your GPG mails on your phone this application is a must have. Please note, due to some small bugs in K-9 Mail and/or APG, it's very advisable to disable HTML mail and use only Plain text. HTML mails are not encrypted nicely and are often not readable.

7.6 Creating your PGP keys

Enigmail comes with a nice wizard to help you create a public/private key pair (see the chapter introducing PGP for an explanation). You can start the wizard at any time within Thunderbird by selecting OpenPGP > Setup Wizard from the menu on top.

1. This is what the wizard looks like. Please read the text on every window carefully. It provides useful information and helps you setup PGP to your personal preferences. In the first screen, click on Next to start the configuration.
2. The wizard asks you whether you want to sign all your outgoing mail messages. Signing all your messages is a good choice. If you choose not to, you can still manually decide to sign a message when you are composing it. Click on the 'Next' button after you have made a decision.
3. On the following screen, the wizard asks you whether you want to encrypt *all* your outgoing mail messages. Unlike signing of mails, encryption requires the recipient to have PGP software installed. You should probably answer 'no' to this question, so that you will send normal (unencrypted) mail by default. After you have made your decision, click on the 'Next' button.

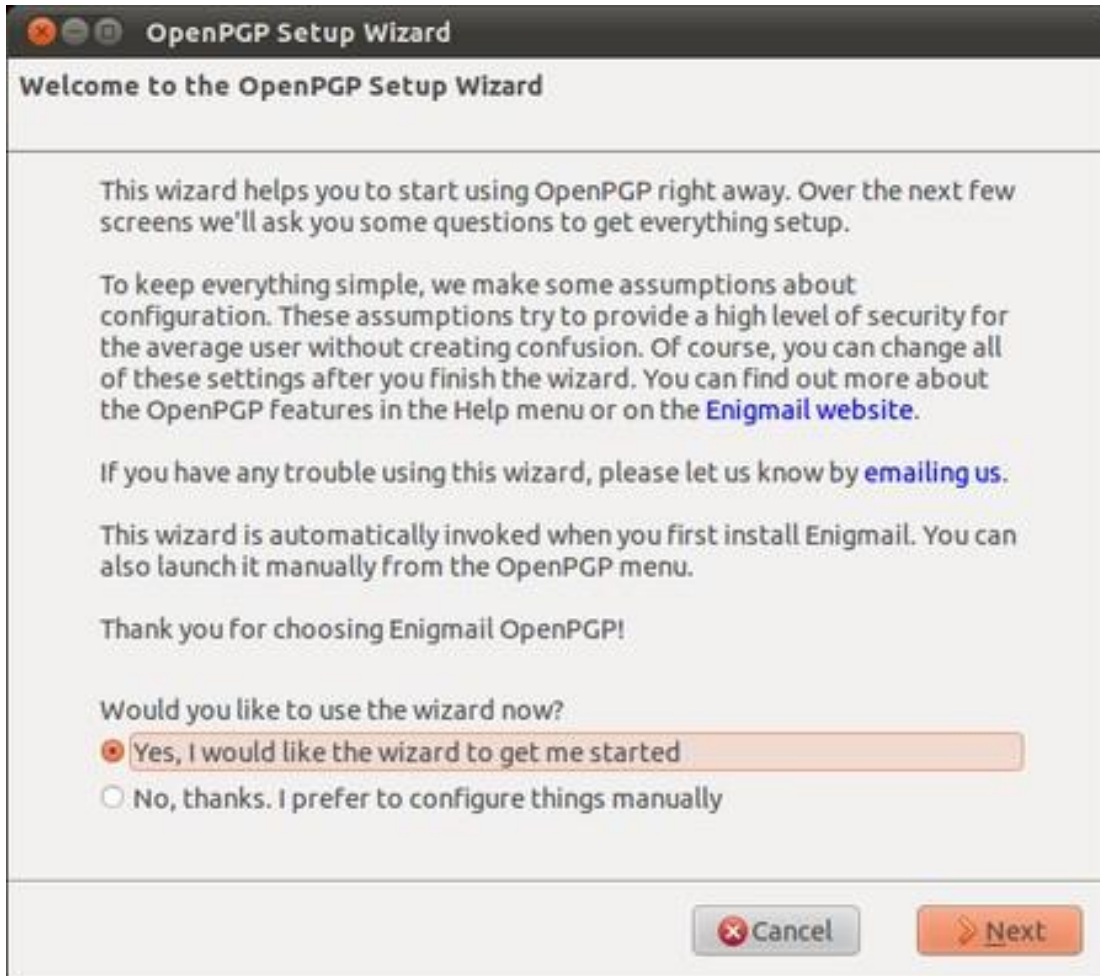


Figure 7.20:GPG Keys



Figure 7.21:GPG Keys



Figure 7.22:GPG Keys

4. On the following screen the wizard asks if it can change some of your mail formatting settings to better work with PGP. It is a good choice to answer 'Yes' here. This will mean that by default, mail will be composed in plain text rather than HTML. Click on the 'Next' button after you have made your decision.

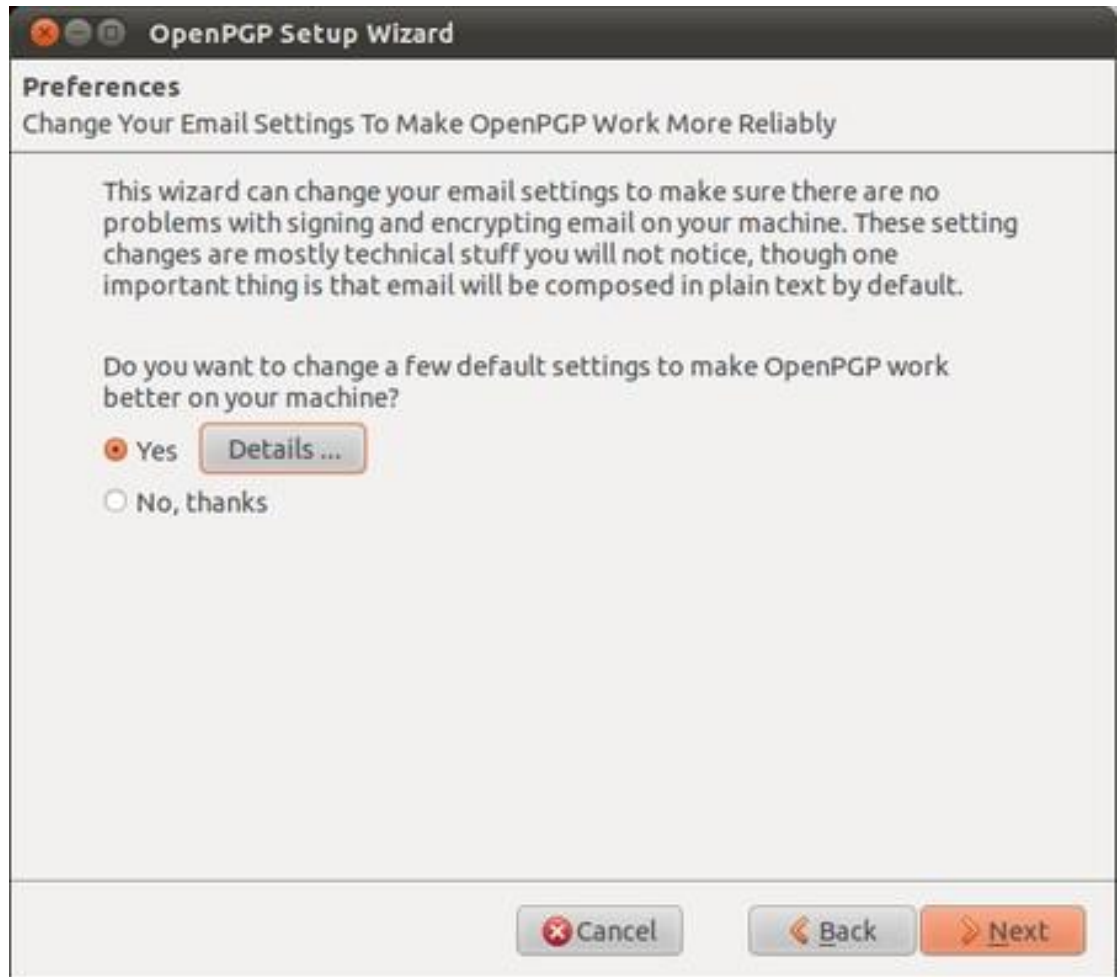


Figure 7.23:GPG Keys

5. In the following screen, select one of your mail accounts; the default is selected for you if you only have one. In the 'Passphrase' text box you must enter a password. This is a *new* password which is used to protect your private key. It is **very important** to remember this password, because you cannot read your own encrypted emails if you forget it. Make it a **strong** password, ideally 20 characters or longer. Please see the chapter on passwords for help on creating unique, long and easy to remember passwords. After you have selected your account and created a passphrase, click on the 'Next' button.



Figure 7.24:GPG Keys

6. In the following screen the wizard summarizes the actions it will take to enable PGP encryption for your account. If you are satisfied, click the 'Next' button.



Figure 7.25:GPG Keys

7. Your keys will be created by the wizard, which will take some time. When completed, click on the 'Next' button.
8. You now have your own PGP key-pair. The wizard will ask you if you also want to create a 'Revocation certificate'. This is a file which can be used to inform everyone if your private key is compromised, for example if your laptop is stolen. Think of it as a 'kill switch' for your PGP identity. You may also wish to revoke the key simply because you have generated a new one, and the old one is obsolete.



Figure 7.26:GPG Keys

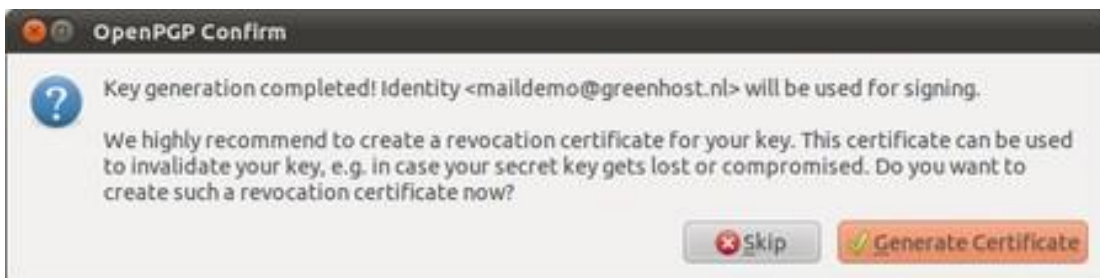


Figure 7.27:GPG Keys

9.If you decided to generate a revocation certificate, the wizard will ask you where the file should be saved. The dialog will look different depending on which operating system you use. It is a good idea to rename the file to something sensible like my_revocation_certificate. Click on ‘Save’ when you you have decided on a location.

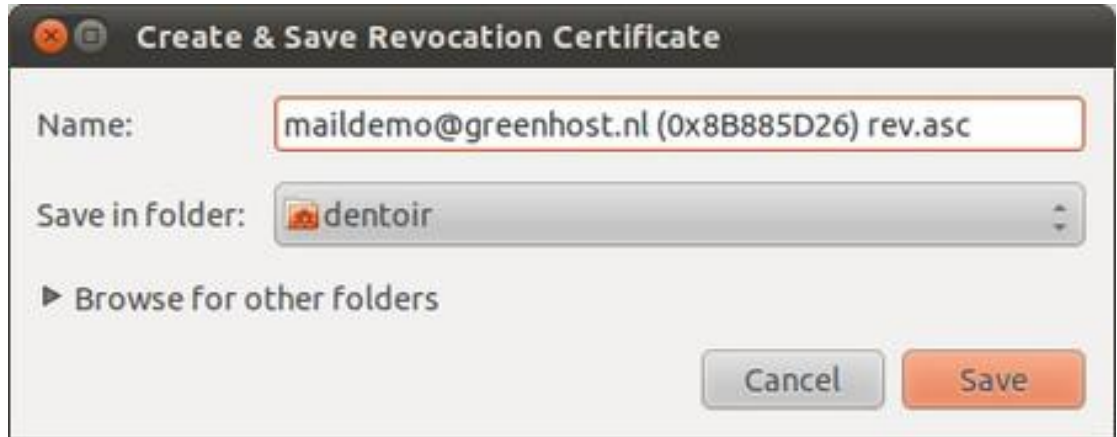


Figure 7.28:GPG Keys

10. If you decided to generate a revocation certificate, the wizard informs you it has been successfully stored. You may want to print it out or burn it to a CD and keep it in a safe place.



Figure 7.29:GPG Keys

11. The wizard will inform you it has completed.

Congratulations, you now have a fully PGP-configured mail client. In the next chapter we will explain how to manage your keys, sign messages and do encryption. Thunderbird can help you do a lot of these things automatically.

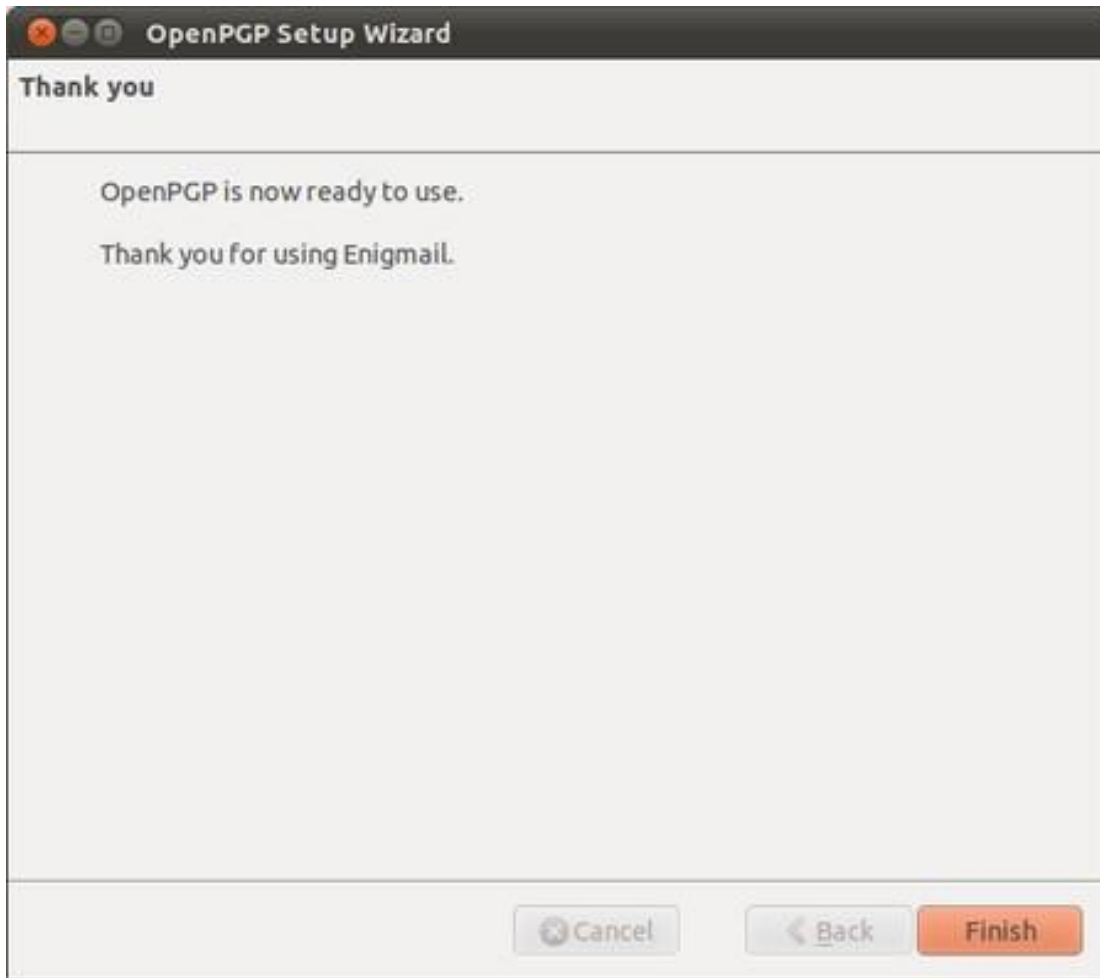


Figure 7.30:GPG Keys

7.7 Daily PGP usage

In the previous chapters we have explained how to set up a secure mail environment using Thunderbird, GPG and Enigmail. We assume you have installed the software and have successfully followed the wizard instructions to generate an encryption key-pair as described in the previous chapter. This chapter will describe how to use your secured Thunderbird in daily life to protect your e-mail communication. In particular we will focus on:

1. Encrypting attachments
2. Entering your pass-phrase
3. Receiving encrypted e-mail
4. Sending and receiving public keys
5. Receiving public keys and adding them to your key ring
6. Using public key servers
7. Signing e-mails to an individual
8. Sending encrypted e-mails to an individual
9. Automating encryption to certain recipients
10. Verifying incoming e-mails
11. Revoking your GPG key pair

12.What to do when you have lost your secret key, or forgot your passphrase 13.What to do when your secret key has been stolen, or compromised 14.Backing up your keys

First we shall explain two dialog windows that will inevitably appear after you start using Thunderbird to encrypt your emails.

7.7.1 *Encrypting attachments*

The dialog window below will pop-up whenever you are sending an encrypted email with attachments for the first time. Thunderbird asks a technical question on how to encrypt attachments to your mail. The second (default) option is the best choice, because it combines security with the highest compatibility. You should also select the ‘Use the selected method for all future attachments’ option. Then click ‘OK’ and your mail should be sent with no further delay.



Figure 7.31:Daily GPG Usage

7.7.2 *Entering your pass-phrase*

For security reasons, the pass-phrase to your secret key is stored temporarily in memory. Every now and then the dialog window below will pop-up. Thunderbird asks you for the pass-phrase to your secret key. This should be different from your normal email password. It was the pass-phrase you have entered when creating your key-pair in the previous chapter. Enter the pass-phrase in the text-box and click on ‘OK’

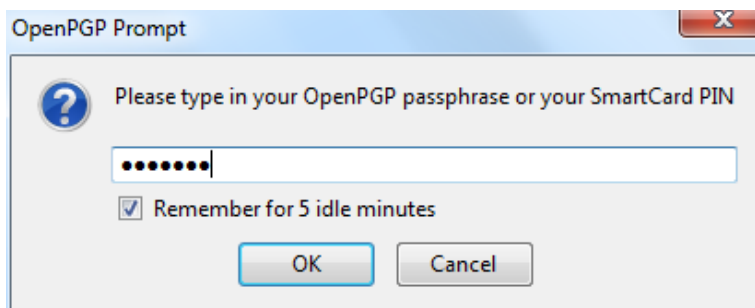


Figure 7.32:Daily GPG Usage

7.7.3 Receiving encrypted e-mails

The decryption of e-mails is handled automatically by Enigmail, the only action that may be needed on your behalf is to enter the pass-phrase to your secret key. However, in order to have any kind of encrypted correspondence with somebody, you will first need to exchange public keys.

7.7.4 Sending and receiving public keys

There are multiple ways to distribute your public key to friends or colleagues. By far the simplest way is to attach the key to a mail. In order for your friend to be able to trust that the message actually came from you, you should inform them in person (if possible) and also require them to reply to your mail. This should at least prevent easy forgeries. You have to decide for yourself what level of validation is necessary. This is also true when receiving emails from third-parties containing public keys. Contact your correspondent through some means of communication other than e-mail. You can use a telephone, text messages, Voice over Internet Protocol (VoIP) or any other method, but you must be absolutely certain that you are really talking to the right person. As a result, telephone conversations and face-to-face meetings work best, if they are convenient and if they can be arranged safely.

Sending your public key is easy.



1. In Thunderbird, click on the icon.

2. Compose a mail to your friend or colleague and tell them you are sending them your PGP public key. If your friend does not know what that means, you may have to explain them and point them to this documentation.

3. Before actually sending the mail, click to OpenPGP > Attach My Public Key option on the menu bar of the mail compose window. Next to this option a marked sign will appear. See the example below.



4. Send your mail by clicking on the button.

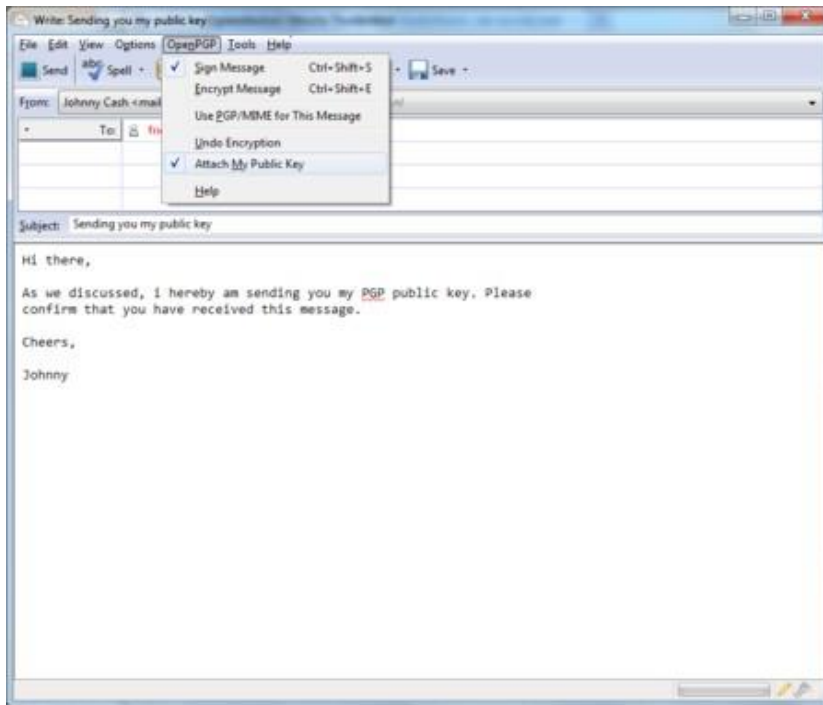


Figure 7.33:Daily GPG Usage

7.7.5 Receiving public keys and adding them to your keyring

Lets say we receive a public key from a friend by mail. The key will show up in Thunderbird as an *attached file*. Scroll down the message and below you will find tabs with one or two file names. The extension of this public key file will be `.asc`, different from the extension of an attached GPG signature, which ends with `.asc.sig`

Look at the example email in the next image, which is a received, signed GPG message containing an attached public key. We notice a yellow bar with a warning message: 'OpenPGP: Unverified signature, click on 'Details' button for more information'. Thunderbird warns us that the sender is not known yet, which is correct. This will change once we have accepted the public key.

What are all those strange characters doing in the mail message? Because Thunderbird does not yet recognize the signature as valid, it prints out the entire raw signature, just as it has received it. This is how digitally signed GPG messages will appear to those recipients who do not have your public key.

The most important thing in this example is to find the attached GPG public key. We mentioned it is a file that ends with `.asc`. In this example it's the first attachment on the left, in the red circle. Double-clicking on this attachment will make Thunderbird recognize the key.

After we have clicked on the attachment, the following pop-up will appear. Thunderbird has recognized the GPG public key file. Click on 'Import' to add this key

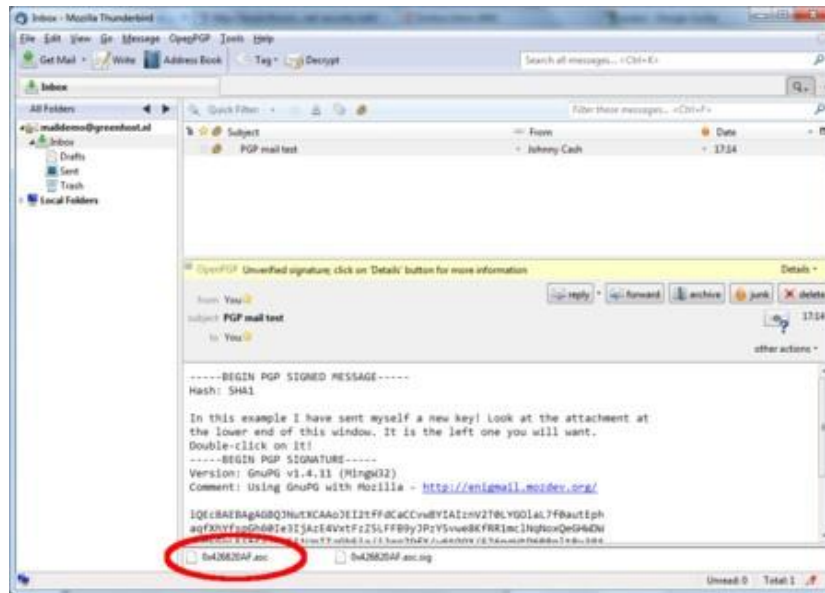


Figure 7.34:Daily GPG Usage

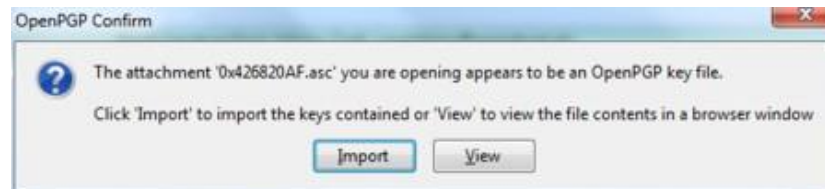


Figure 7.35:Daily GPG Usage

to your keyring. The following pop-up should appear. Thunderbird says the operation was successful. Click on 'OK' and you are almost done.

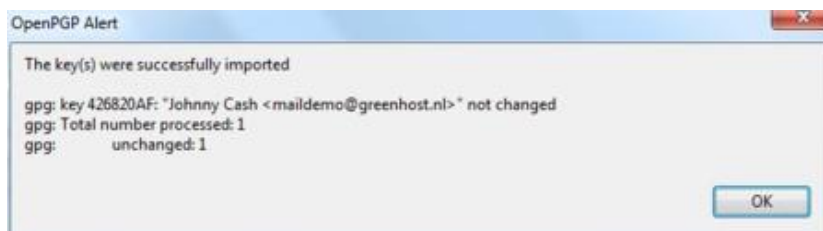


Figure 7.36:Daily GPG Usage

We are back in the main Thunderbird screen and we refresh the view on this particular example message, by clicking on some other message and back for example. Now the body of the message looks different (see below). This time Thunderbird *does* recognize the signature, because we have added the public key of the sender.

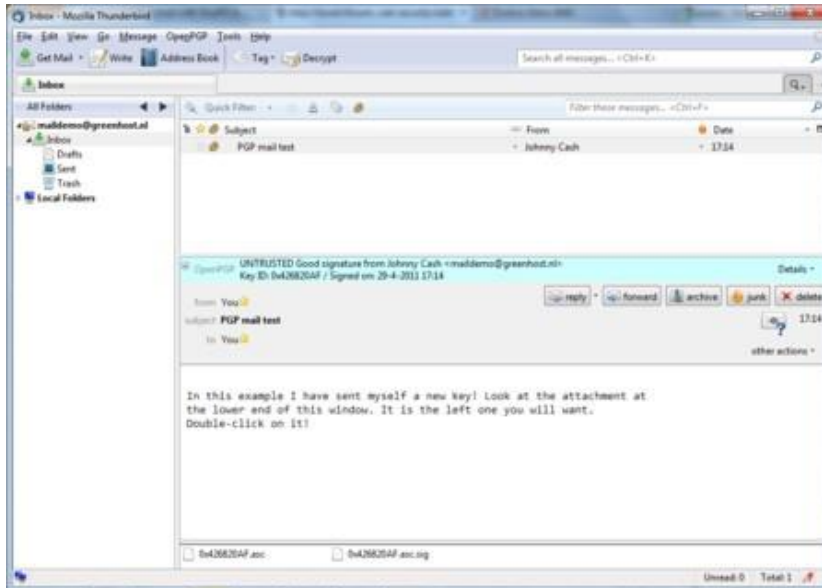


Figure 7.37:Daily GPG Usage

There is still one thing that remains. While Thunderbird now recognizes the signature, we should explicitly trust that the public key really belongs to the sender in real life. We realize this when we take a closer look at the green bar (see below). While the signature is good, it is still UNTRUSTED.

We will now decide to trust this particular public key and the signatures made by it. We can do this immediately by clicking on 'Details'. A small menu will appear (see below). From this menu we should click on the option 'Sign Sender's Key . . . '.

After we have selected 'Sign Sender's Key . . . ' we will get another selection window



Figure 7.38:Daily GPG Usage



Figure 7.39:Daily GPG Usage

(see below). We are requested to state how carefully we have checked this key for validity. The explanation of levels of trust and trust networks in GPG falls outside the scope of this document.

We will not use this information, therefore we will just select the option ‘I will not answer’. Also select the option ‘Local signature (cannot be exported)’. Click on the ‘OK’ button to finishing signing this key. This finishes accepting the public key. You can now send encrypted mail to this individual.

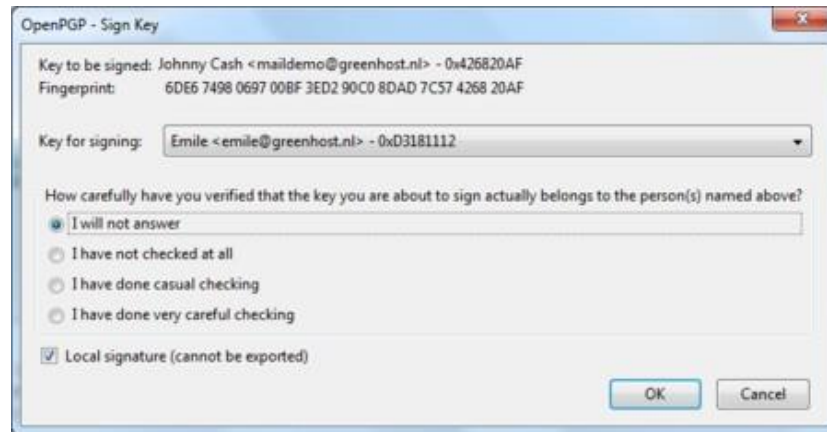


Figure 7.40:Daily GPG Usage

7.7.6 Using public key servers

Another method of distributing public keys is by putting them on a public key server. This allows anyone to check whether your email address has GPG support, and then download your public key.

To put your own key on a keyservers, take the following steps.

- 1.Head to the key manager by using the Thunderbird menu and click on OpenPGP > Key Management

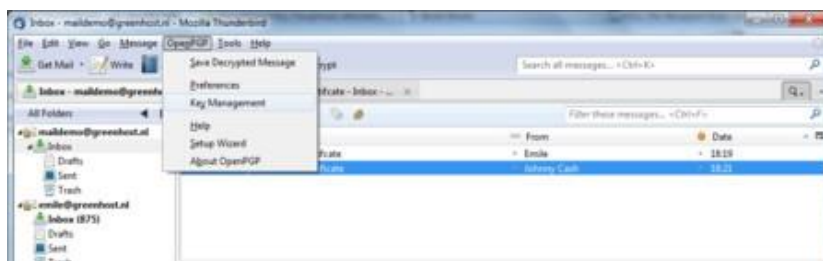


Figure 7.41:Daily GPG Usage

- 2.The key management window will be displayed and looks like this:

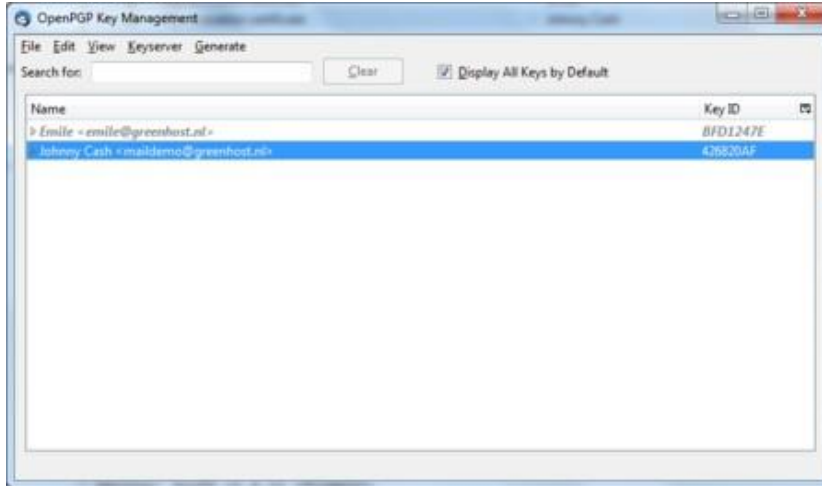


Figure 7.42:Daily GPG Usage

3. You need to have selected the 'Display All Keys by Default' option to get a list of all your keys. Look up your own email address in the list and right click on the address. A selection window will appear with some options. Select the option 'Upload Public Keys to Keyserver'.

4. You will see a small dialog window like below. The default server to distribute your keys to is good. Press 'OK' and distribute your public key to the world.

To look up whether some email address has a public key available on a server, take the following steps.

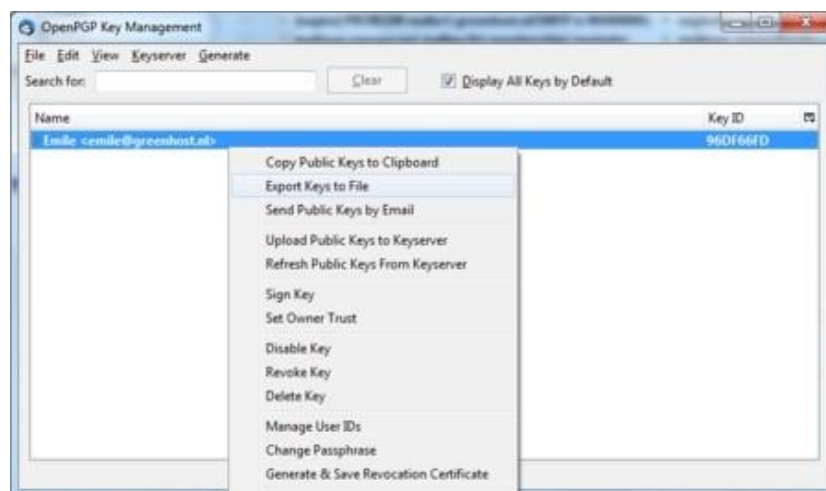


Figure 7.43:Daily GPG Usage

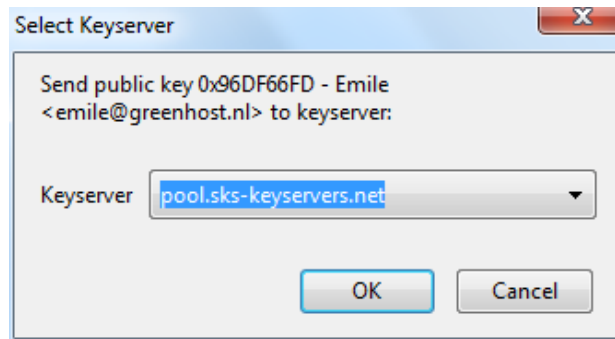


Figure 7.44:Daily GPG Usage

- 1.Head to the key manager by using the Thunderbird menu and click on OpenPGP > Key Management
- 2.In the key manager window menu bar, select Keyserver > Search for Keys

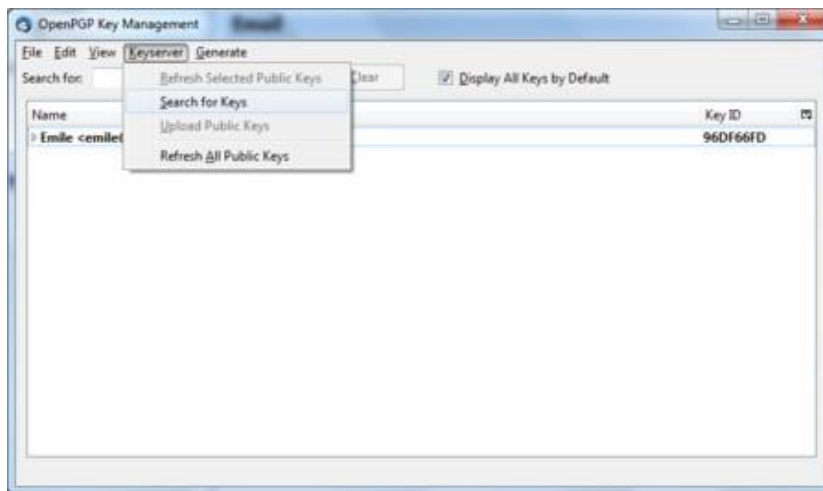


Figure 7.45:Daily GPG Usage

- 3.In this example we will look-up up the key for the creator of PGP software, Philip Zimmermann. After we have entered the email address, we click on 'OK'.

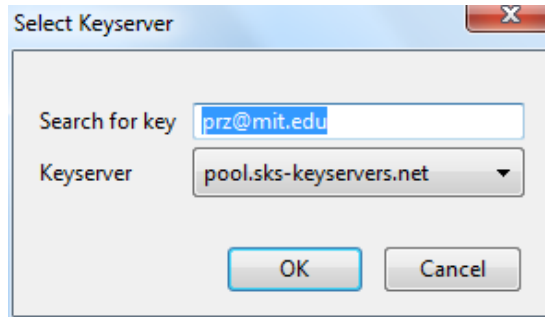


Figure 7.46:Daily GPG Usage

- 4.The next window displays the result of our search. We have found the public key. It is automatically selected. Just click on 'OK' to import the key.
- 5.Importing the key will take some time. On completion you should see a pop-up window like below.

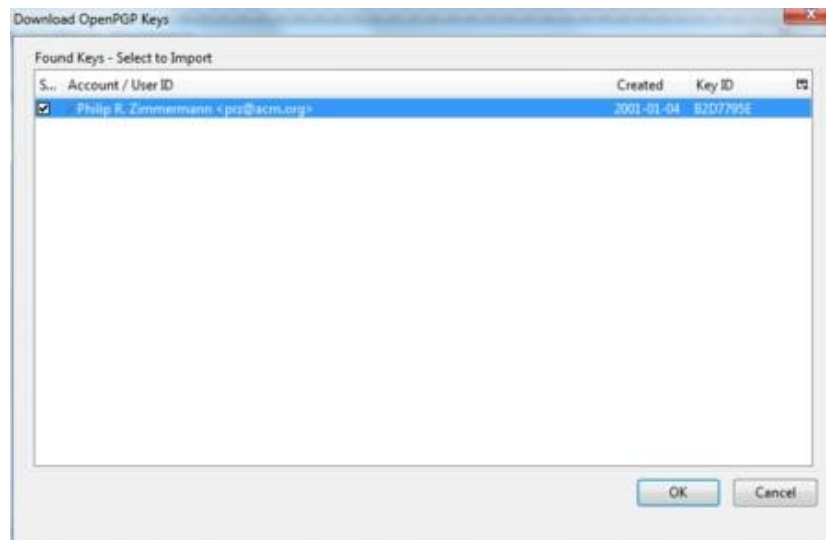


Figure 7.47:Daily GPG Usage

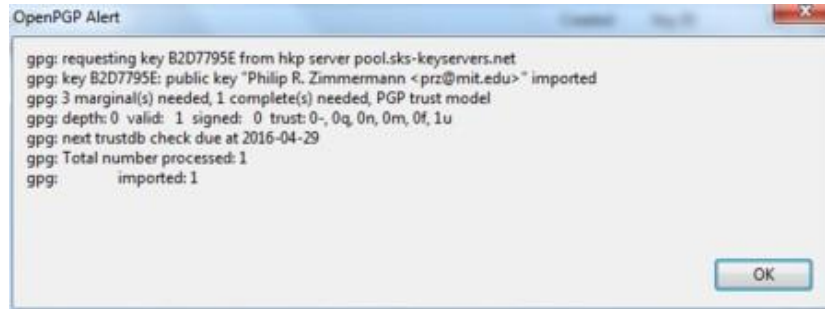


Figure 7.48:Daily GPG Usage

6.The final step is to locally sign this key, to indicate that we trust it. When you are back in the key manager, make sure you have selected the ‘Display All Keys by Default’ option. You should now see the newly imported key in the list. Right-click on the address and select the option ‘Sign Key’ from the list.

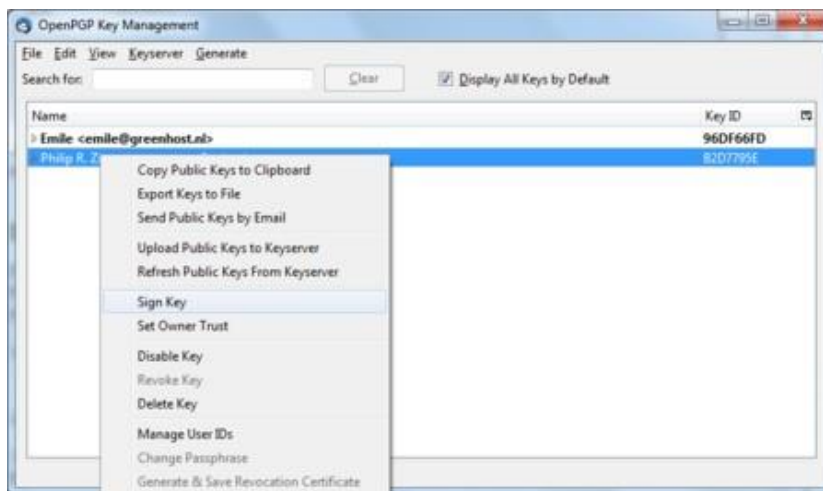


Figure 7.49:Daily GPG Usage

7.Select the options ‘I will not answer’ and ‘Local signature (cannot be exported)’, then click on ‘OK’. You are now finished and can send Philip Zimmermann encrypted mail.

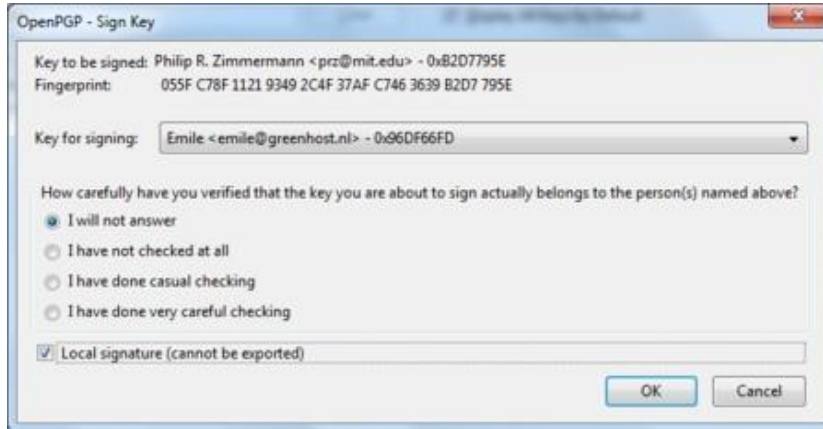



Figure 7.50:Daily GPG Usage

7.7.7 Signing emails to an individual

Digitally signing email messages is a way to prove to recipients that you are the actual sender of a mail message. Those recipients who have received your public key will be able to *verify* that your message is authentic. However, take note that signing an email will make it very hard (if not impossible) to deny that you are the author of the message.

1. Offer your friend your public key, using the method described earlier in this chapter.



2. In Thunderbird, click on the  icon.

3. Before actually sending the mail, enable the OpenPGP > Sign Message option via the menu bar of the mail compose window, if it is not enable already. Once you have enabled this option, by clicking on it, a marked sign will appear. Clicking again should disable encryption again. See the example below.

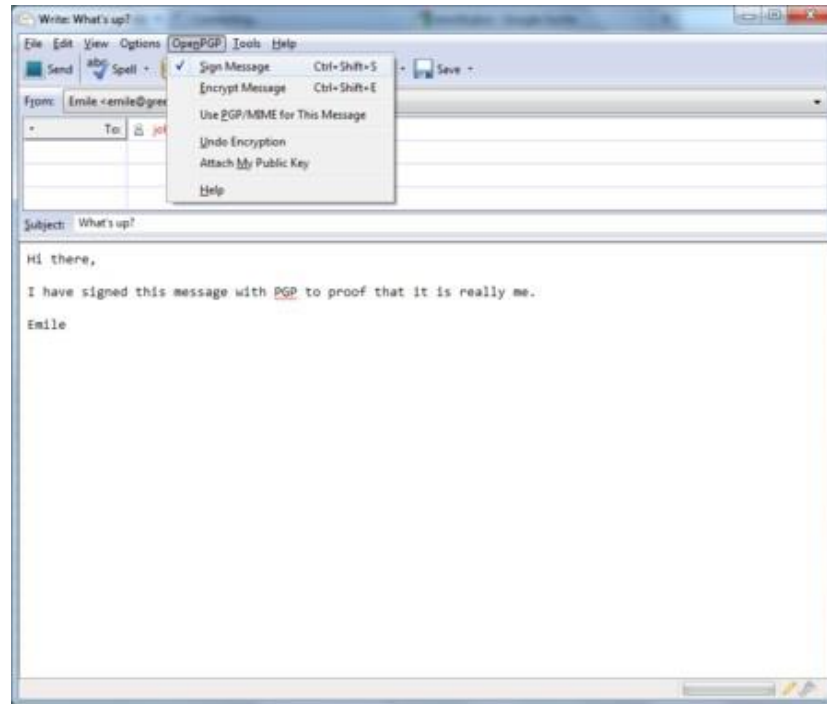




Figure 7.51:Daily GPG Usage

4. Click on the  button and your signed mail will be sent.

7.7.8 Sending encrypted mails to an individual

1. You should have received the public key from the friend or colleague you want to email and you should have accepted their public key, using the method describe earlier in this chapter.

2. In Thunderbird, click on the  icon.

3. Compose a mail to the friend or colleague, from who you have previously received their public key. **Remember the subject line of the message will not be encrypted**, only the message body itself, and any attachments.
4. Before actually sending the mail, enable the OpenPGP > Encrypt Message option via the menu bar of the mail compose window, if it is not enabled already. Once you have enabled this option, by clicking on it, a marked sign will appear. Clicking again should disable encryption again. See the example below.

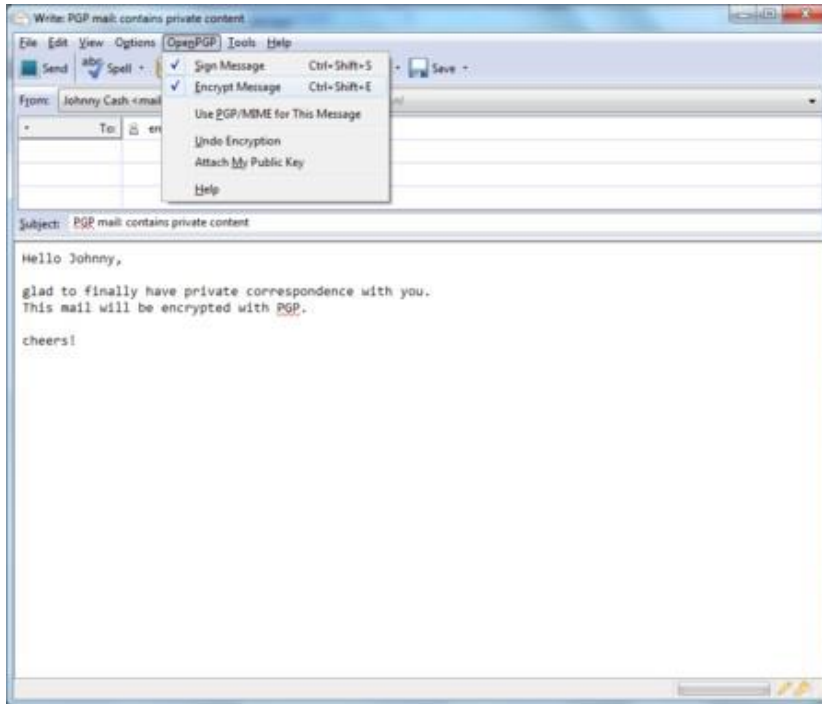



Figure 7.52:Daily GPG Usage

5. Click on the  button and your encrypted mail will be sent.

7.7.9 Automating encryption to certain recipients

You will often want to make sure all your messages to a certain colleague or friend are signed and encrypted. This is good practice, because you may forget to enable the encryption manually. You can do this by editing the per-recipient rules. To do this we access the OpenPGP per-recipient rule editor.

Select OpenPGP > Preferences from the Thunderbird menu bar.

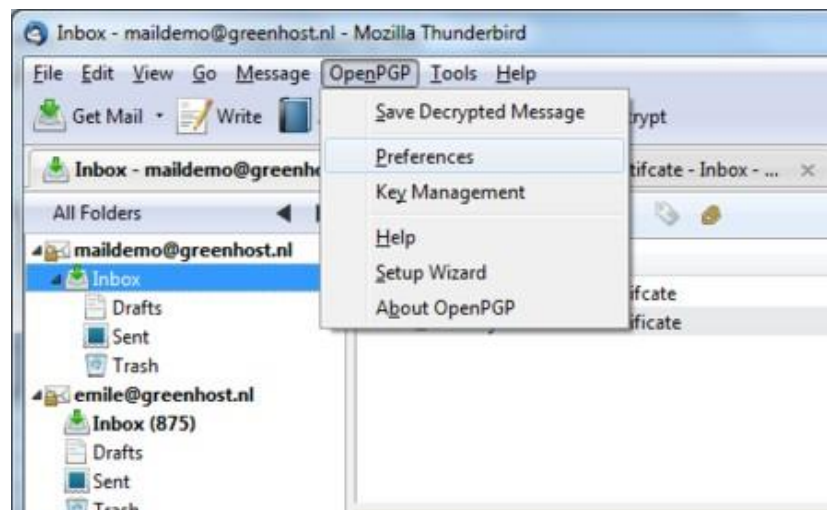


Figure 7.53:Daily GPG Usage

The preferences window will appear like below. We need to click on ‘Display Expert Settings’.

New menu tabs will appear in the window. Go to the tab ‘Key Selection’ and then click on the button labeled ‘Edit Rules . . . ’

We are now shown the per-recipient rules editor (see below). This editor can be used to specify the way how messages to certain recipients are sent. We will now add a rule saying we want to encrypt and sign all mail messages to maildemo@greenhost.nl

First click on the ‘Add’ button.

Now the window to add a new rule will be shown.

The first thing we should enter is the email address of the recipient. In the example below we have entered maildemo@greenhost.nl

Now we will set the encryption defaults by using the drop-downs below. For Signing select ‘Always’. For Encryption also select ‘Always’.

Finally we have to select the *public key* of the recipient, with which to encrypt our messages. Do not forget this important step, otherwise the e-mail will not be encrypted. Click on the button labeled ‘Select Key(s). . . ’. The key selection window will show up. The most obvious key will be selected by default. In the example below, we only have one public key available. We can select keys by clicking on the small box next to the address. Then we click ‘OK’ and close all relevant windows and we are finished.

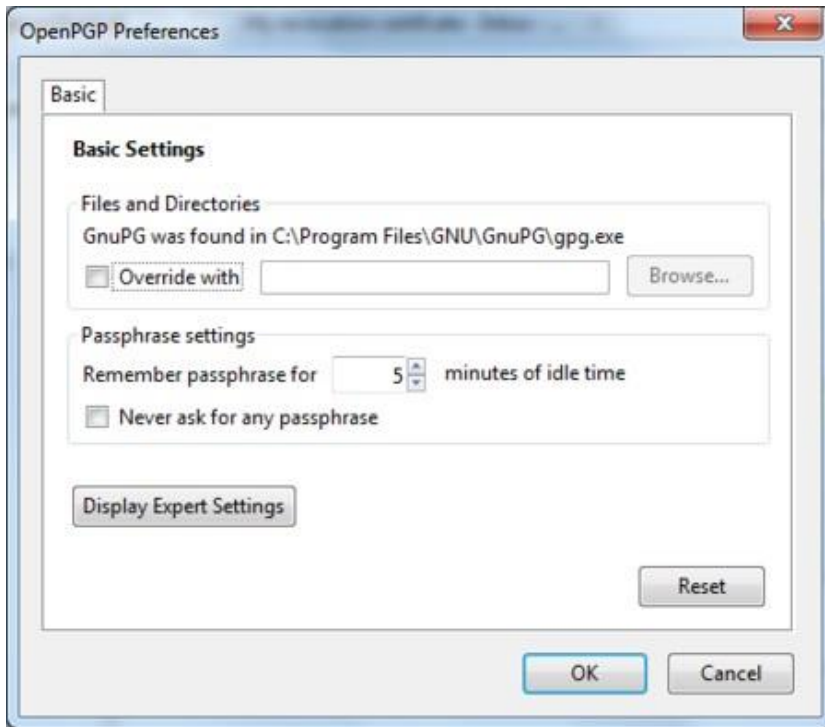


Figure 7.54:Daily GPG Usage

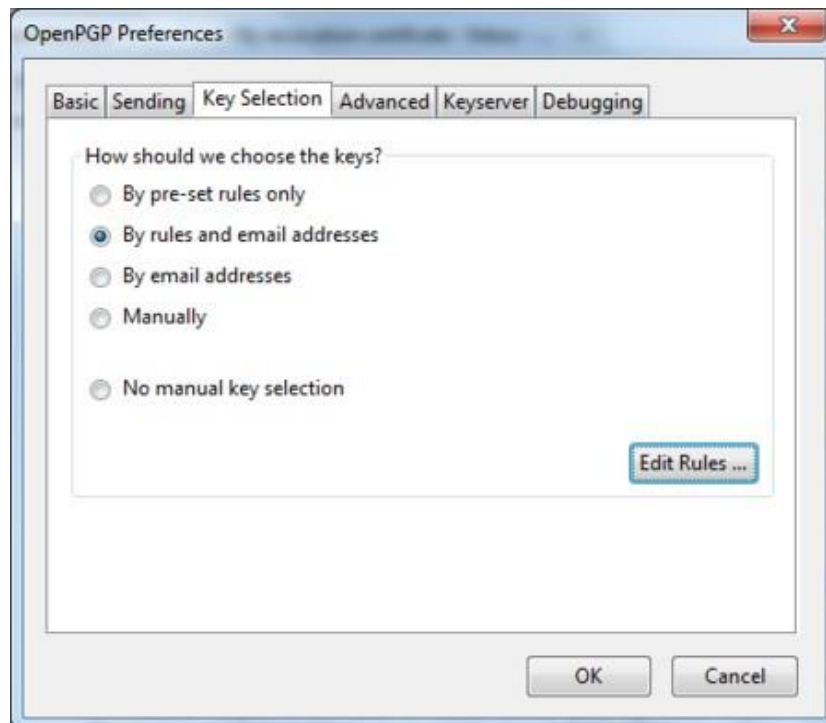


Figure 7.55:Daily GPG Usage

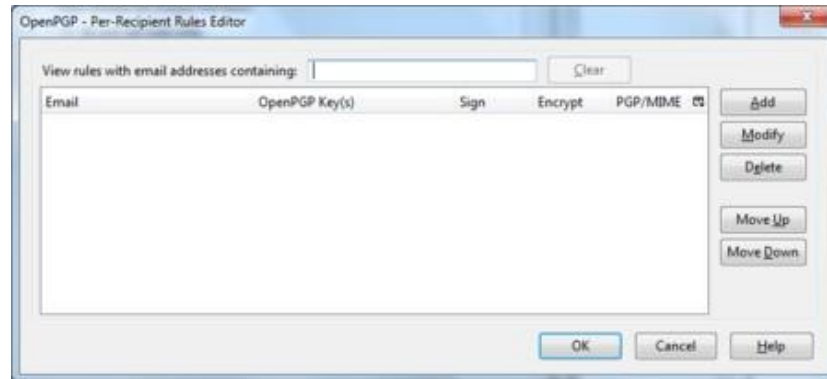


Figure 7.56:Daily GPG Usage

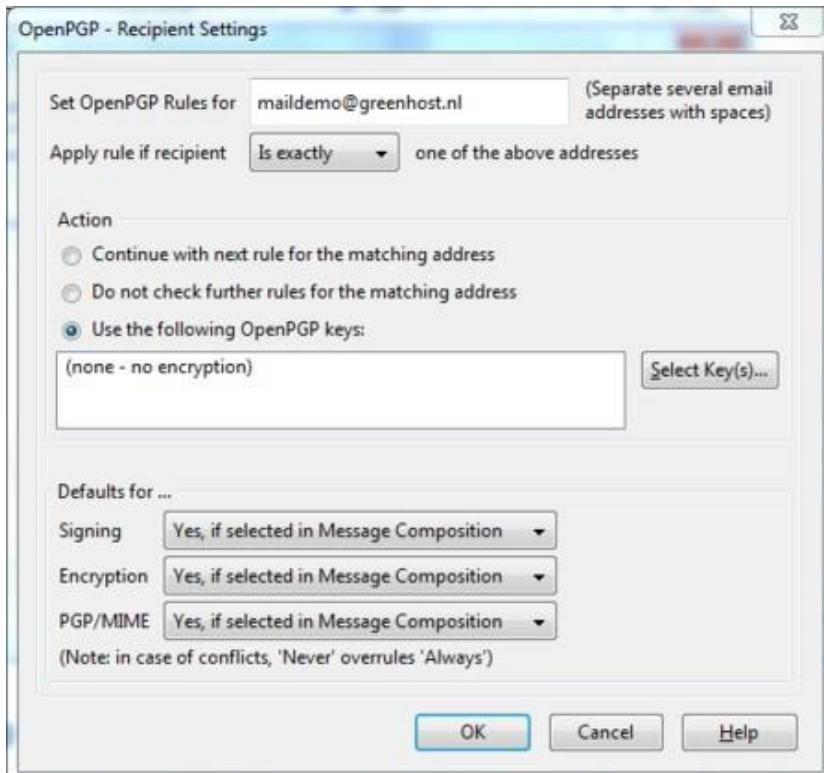


Figure 7.57:Daily GPG Usage

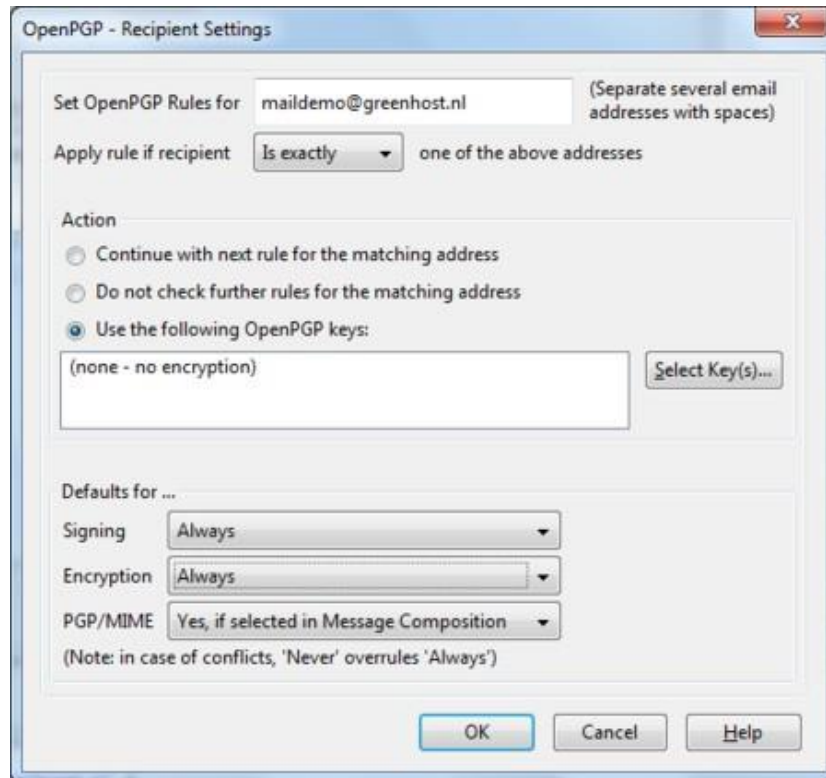


Figure 7.58:Daily GPG Usage



Figure 7.59:Daily GPG Usage

7.7.10 Verifying incoming e-mails

Decrypting email messages sent to you will be fully automatic and transparent. But it is obviously important to see whether or not a message to you has in fact been encrypted or signed. This information is available by looking at the special bar above the message body.

A valid signature will be recognized by a green bar above the mail message like the example image below.



Figure 7.60:Daily GPG Usage

The last example message was signed but not encrypted. If the message had been encrypted, it would show like this:



Figure 7.61:Daily GPG Usage

When a message which has been encrypted, but not signed, it could have been a forgery by someone. The status bar will become gray like in the image below and tells you that while the message was sent securely (encrypted), the sender could have been someone else than the person behind the email address you will see in the 'From' header. The signature is necessary to verify the real sender of the message. Of course it is perfectly possible that you have published your public key on the Internet and you allow people to send you emails anonymously. But is it also possible that someone is trying to impersonate one of your friends.



Figure 7.62:Daily GPG Usage

Similarly if you receive a signed email from somebody you know, and you have this person's public key, but still the status bar becomes yellow and displays a warning message, it is likely that someone is attempting to send you forged emails!

Sometimes secret keys get stolen or lost. The owner of the key will inform his friends and send them a so-called revocation certificate (more explanation of this in the next paragraph). Revocation means that we no longer trust the old key. The thief may afterwards still try his luck and send you a falsely signed mail message. The status bar will now look like this:

Strangely enough Thunderbird in this situation will still display a green status bar! It is important to look at the contents of the status bar in order to understand the encryption aspects of a message. GPG allows for strong security and privacy, but only if you are familiar with its use and concepts. Pay attention to warnings in the status bar.

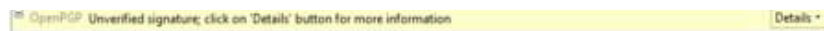


Figure 7.63:Daily GPG Usage

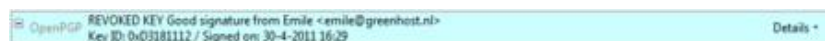


Figure 7.64:Daily GPG Usage

7.7.11 Revoking your GPG key-pair

Your secret key has been stolen by somebody. Your harddisk crashed and you have lost all your data. If your key is lost, you can no longer decrypt messages. If your key has been stolen, somebody else can decrypt your communication. You need to make a new set of keys. The process of creating keys, using the OpenPGP wizard in Thunderbird, has been described in this manual. But first you want to tell the world that your old public key is now worthless, or even dangerous to use.

7.7.12 What to do when you have lost your secret key, or forgot your passphrase

During the creation of your key-pair, the OpenPGP wizard offered you the possibility to create a so-called revocation certificate. This is a special file you send to others in the event you have to disable your key. If you have a copy of this file, sending the revocation key is simply sending the file as an attachment to all your friends. You can no longer send signed mails (obviously, because you have lost your secret key). That doesn't matter. Send it as a normal mail. The revocation certificate file could only have been created by the owner of the secret key and proves he or she wants to revoke it. That's why it should normally be kept hidden from others.

If you do not have the revocation certificate, there exists no other option than for you to contact your friends personally and convince them your key is lost and that they should no longer trust it.

7.7.13 What to do when your secret key has been stolen, or compromised

If you have reason to believe your secret key has been compromised, or worse your secret key and passphrase, it is very important to contact others that they should stop sending you encrypted messages. With your secret key, other persons will be able to break the encryption of your e-mail messages if they also have your passphrase. This is also true for those messages you have sent in the past. Cracking the passphrase is not trivial, but it may be possible if the party has lots of resources, like a state or a big organization for example, or if your passphrase is too weak. In any case you should assume the worst and assume your passphrase may have been compromised. Send a revocation certificate file to all your friends or contact them personally and inform them of the situation.

Even after you have revoked your old key pair, the stolen key may still be used to

decrypt your previous correspondence. You should consider other ways to protect that old correspondence, for instance by re-encrypting it with a new key. The latter operation will not be discussed in this manual. If you are uncertain you should seek assistance from experts or look up more information on the web.

7.7.14 Receiving a revocation certificate

If one of your friends sends you a revocation certificate, s/he asks you to distrust his public key from now on. You should always accept such a request and 'import' the certificate to disable their key. The process of accepting a revocation certificate is exactly the same as accepting a public key, as has already been described in the chapter. Thunderbird will ask you if you want to import the 'OpenPGP key file'. Once you have done so, a confirmation pop-up should be displayed like below.

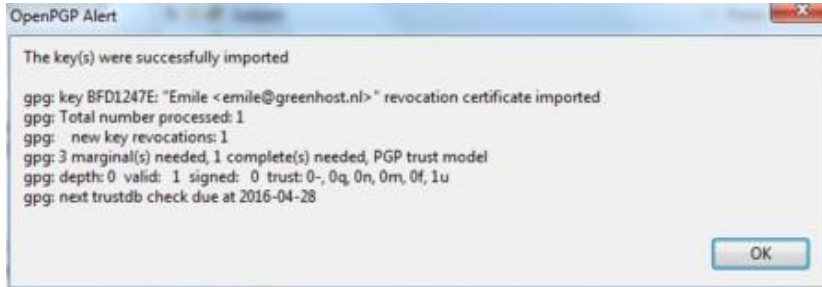


Figure 7.65: Daily GPG Usage

7.7.15 *Preparing for the worst: backup your keys*

Your keys are usually stored on your hard disk as normal files. They may get lost if your computer gets damaged. It is strongly advised to keep a backup of your keys in a safe place, like a vault. Making a backup of your secret key has another security advantage as well. Whenever you fear your laptop or computer is in immediate danger of being confiscated, you can safely delete your key-pair. Your email will be rendered unreadable immediately. At a later stage, you can retrieve your keys from the vault and re-import them in Thunderbird.

To make a backup of your key-pair, first head to the key manager by using the Thunderbird menu and click on OpenPGP > Key Management.

You need to have selected the 'Display All Keys by Default' option to get a list of all your keys. Lookup your own email address in the list and right click on the address. A selection window will appear with some options. Select the option 'Export Keys to File'. Now we will save the key-pair to a file. Thunderbird asks us if we want to include the secret key as well.

We do want to include the secret key, therefore we select 'Export Secret Keys'.

Finally Thunderbird asks us for the location of the key file. You can store the file anywhere you like, network disk, USB-stick. Just remember to hide it away from other

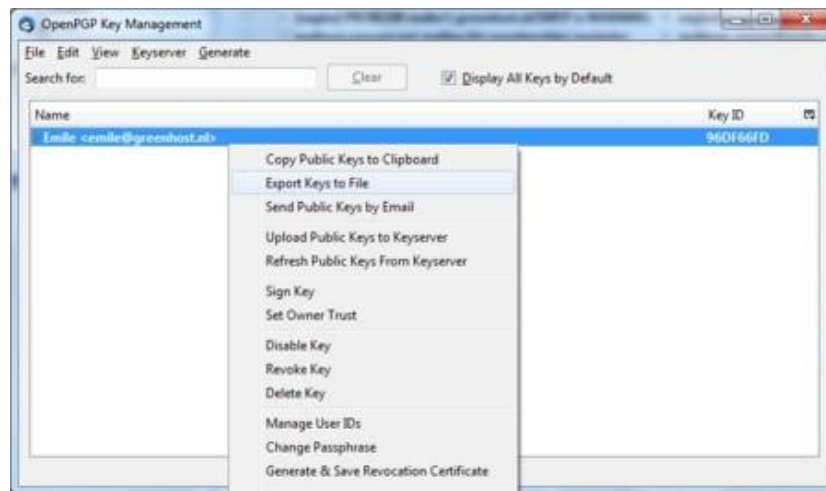


Figure 7.66:Daily GPG Usage



Figure 7.67:Daily GPG Usage

people.

7.7.16 Further reading

More documentation on using GPG with Thunderbird can be found on the website of the Enigmail plugin. The Enigmail handbook is the guide you will want to use.

<http://enigmail.mozdev.org/documentation/handbook.php.html>

7.8 Webmail and PGP

The only safe way of encrypting email inside of the browser window is to encrypt it outside and then copy & paste the encrypted text into the browser window.

For example, write the text in a text editor like gedit, vim or kate and save it as .txt file (in this example “message.txt”. Then type

```
gpg -ase -r <recipients email/gpg id> -r <your gpg id> message.txt
```

A new file called “message.asc” will be created. It contains the encrypted message and can thus be either attached to an email or its content safely copy & pasted into the browser window.

To decrypt a message from the browser window, simply type `gpg` into the command line and hit Enter. Then copy & paste the message to be decrypted into the commandline window and after being asked for your passphrase hit Ctrl+D (this enters a end-of-file character and prompts `gpg` to output the cleartext message).

If using the commandline seems too cumbersome to you, you might consider installing a helper application like `gpgApplet`, `kgpg` or whatever application ships with your operating system.

8 Safer Browsing

8.1 Why Firefox?

Firefox is open source software, developed by the non-profit organisation, the Mozilla Foundation. As such, it is independent from the interests of any one specific company although a [large portion of its funding comes from Google](#) for its placement as the default search engine within the Firefox browser. It is also highly extensible through the add-ons and plugins, which allows users greater control over how the browser acts as compared to Internet Explorer or Chrome (and it’s open-source’d version, Chromium). It should however be noted that this extensibility through add-ons is a double-edged sword and as such add-ons also have great power to subvert the browsers normal activities as well as enhance them.

If you are uncomfortable with Google as the default search engine, this can be changed through the ‘Manage Search Engines. . .’ option from the pull-down menu of the search box. Some more pro-privacy search engines that are worth considering are [Startpage](#) and [DuckDuckGo](#).

8.2 Accessing Firefox on Ubuntu

Firefox is already installed on Ubuntu by default. To open it, click on the Unity side bar where you see the Firefox icon:



Figure 8.1:Firefox on Ubuntu Firefox

starts and a welcome window opens:



Figure 8.2:Firefox on Ubuntu

8.3 Installing on Mac OS X

1.To download Firefox, visit <https://www.mozilla.org/firefox> and click on the big green button labeled “Firefox Free Download”. The download should start automatically, if it does not, click the link to download it manually.

2.When prompted, click **OK**.

Once the download is complete a window similar to this appears: 3.Click and

drag the **Firefox** icon on top of the **Applications** icon.

4.When the installation is finished, close the two small Firefox windows.

5.Eject the Firefox disk image. If this does not work by normal means, select the disk image icon and then, in the Finder menu, select File > Eject Firefox.

6.Now, open the **Applications** directory and drag the **Firefox** icon to the dock:

7.Click the **Firefox** icon in the Dock to start Firefox. The Import Wizard dialog box appears:

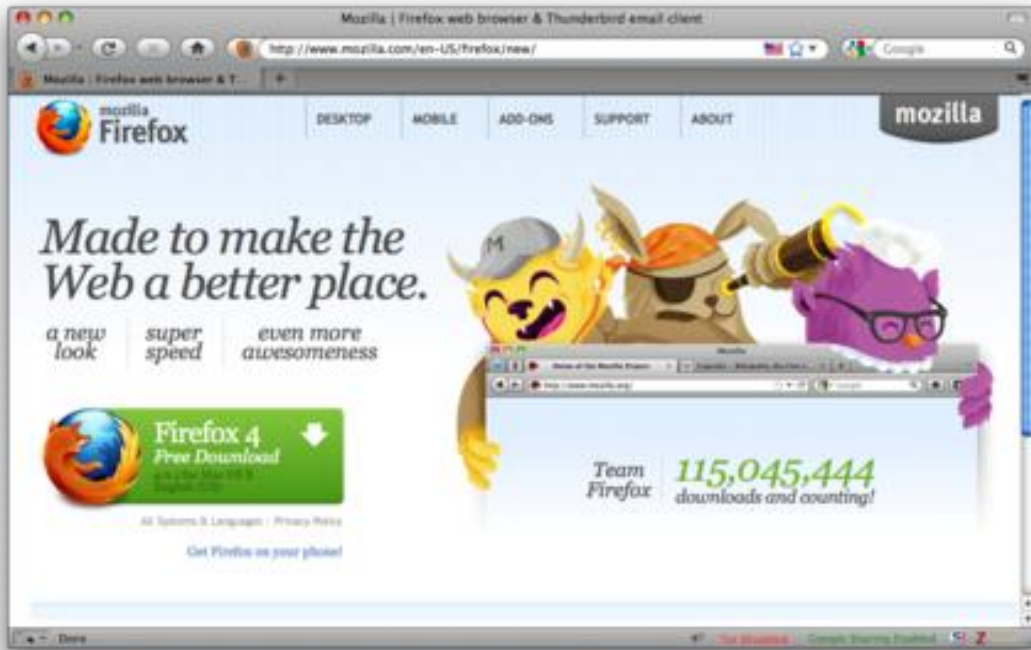


Figure 8.3:Mac OS X Firefox Install



Figure 8.4:Mac OS X Firefox Install



Figure 8.5:Mac OS X Firefox Install



Figure 8.6:Mac OS X Firefox Install



Figure 8.7:Mac OS X Firefox Install

8.To import your bookmarks, passwords and other data from Safari, click **Continue**.
If you don't want to import anything, just select **Cancel**.

Congratulations, you are now ready to use Firefox!

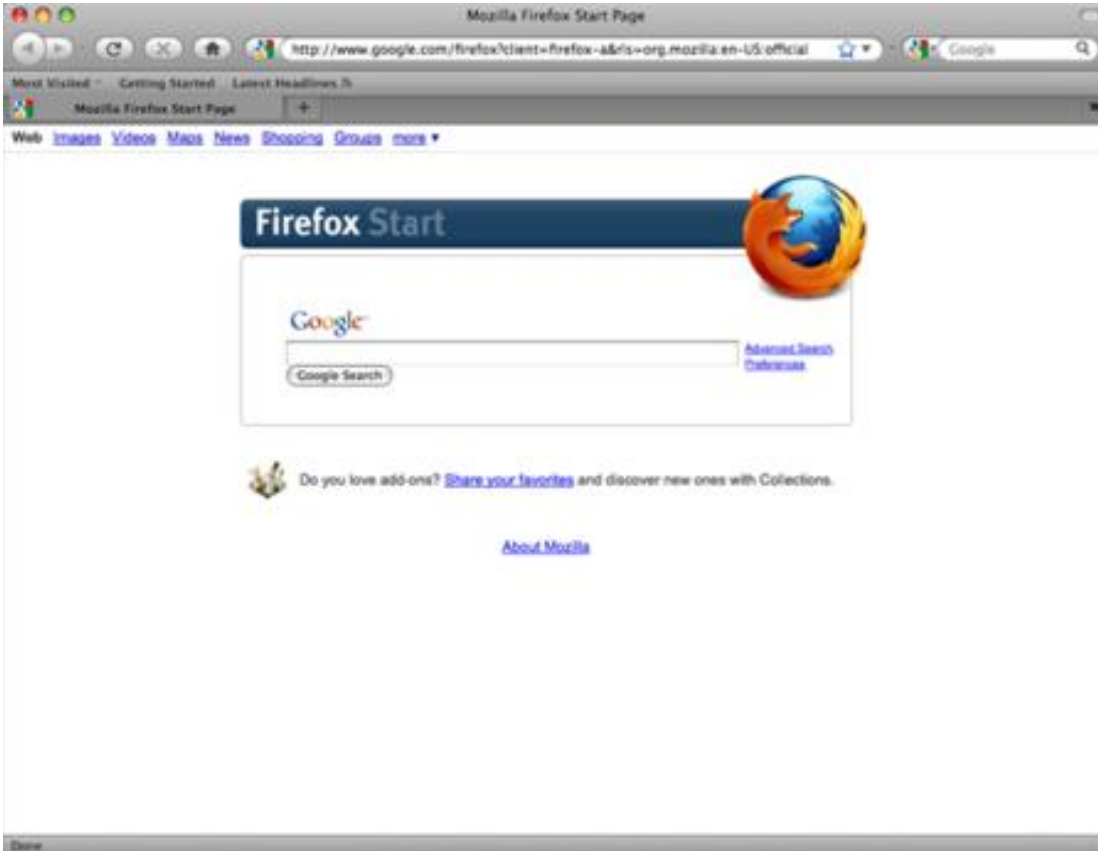


Figure 8.8:Mac OS X Firefox Install

8.4 Installing Firefox on Windows

1. To download Firefox, visit <https://www.mozilla.com/firefox/>.
2. Click the download button and the installation file will begin to download to your computer.
3. Once the download is complete, double-click the installation file to start the Firefox installation wizard.
 - If you are running Windows Vista, you may get a User Account Control prompt. In this case, allow the setup to run by clicking **Continue**.



Figure 8.9: Windows Firefox Install

- If you are running Windows 7, you will be asked whether to allow Firefox to make changes to your computer. Click on **Yes**.

A welcome screen appears.

4. Click **Next** to continue. You will be asked if you would like the standard installation, or whether you would like to customize it. Choose the standard installation and click **Next**.

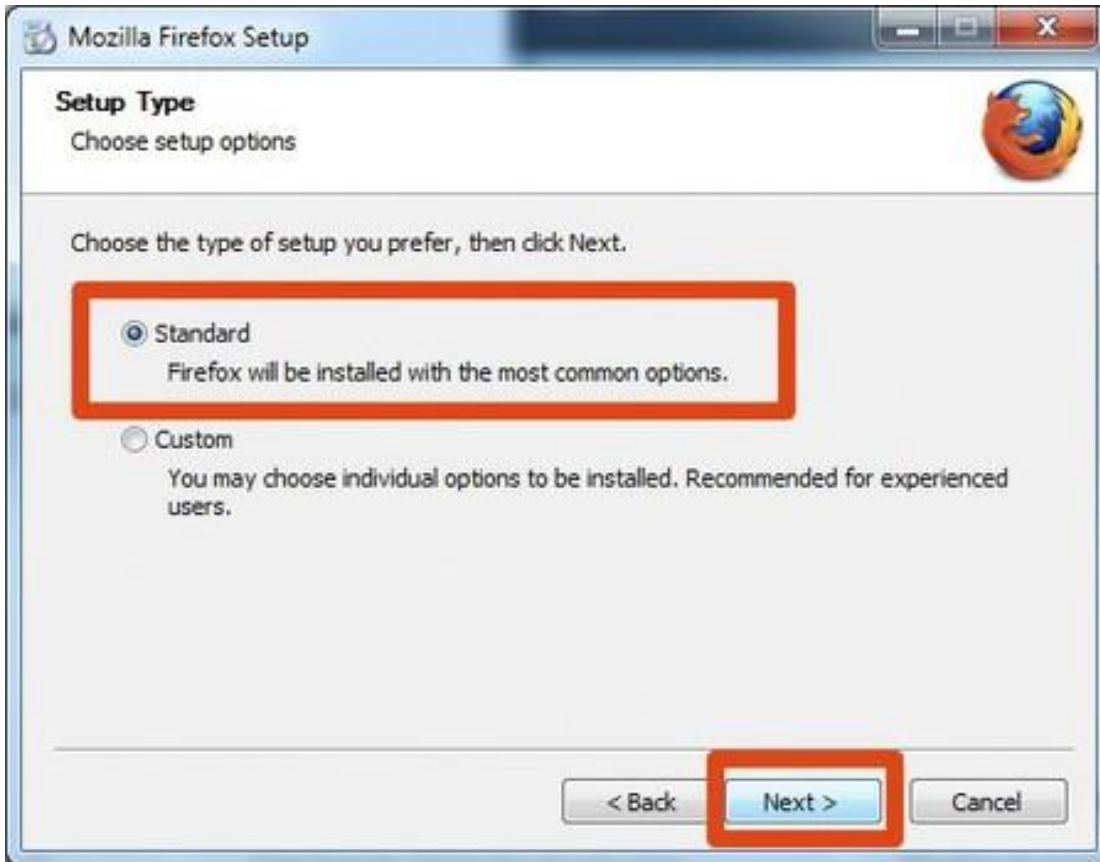


Figure 8.10: Windows Firefox Install

5. You will be asked if you want Firefox to be your default browser. This is recommended.
6. Click **Install**.
7. To import your bookmarks and other data from other browsers (for example Internet Explorer), click **Continue**. If you don't want to import anything, just select **Cancel**.

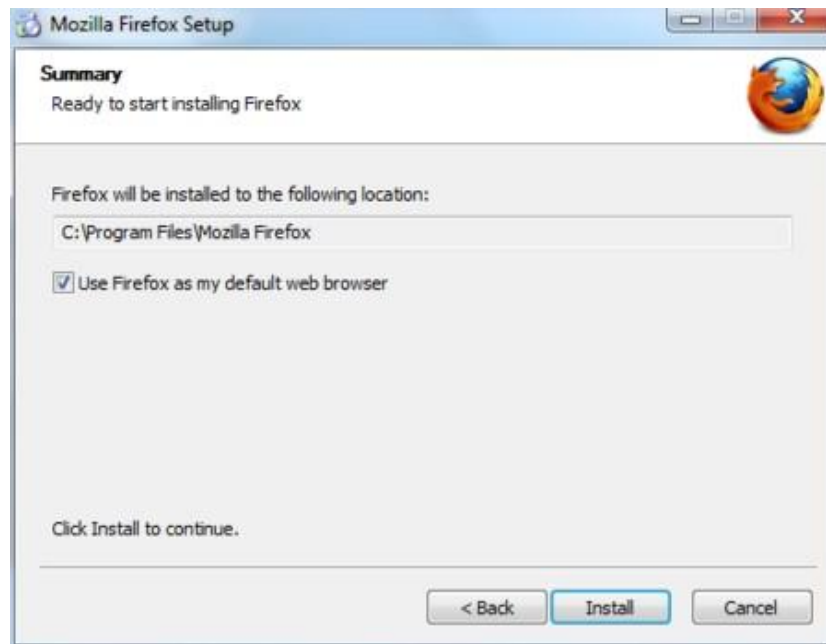


Figure 8.11:Windows Firefox Install

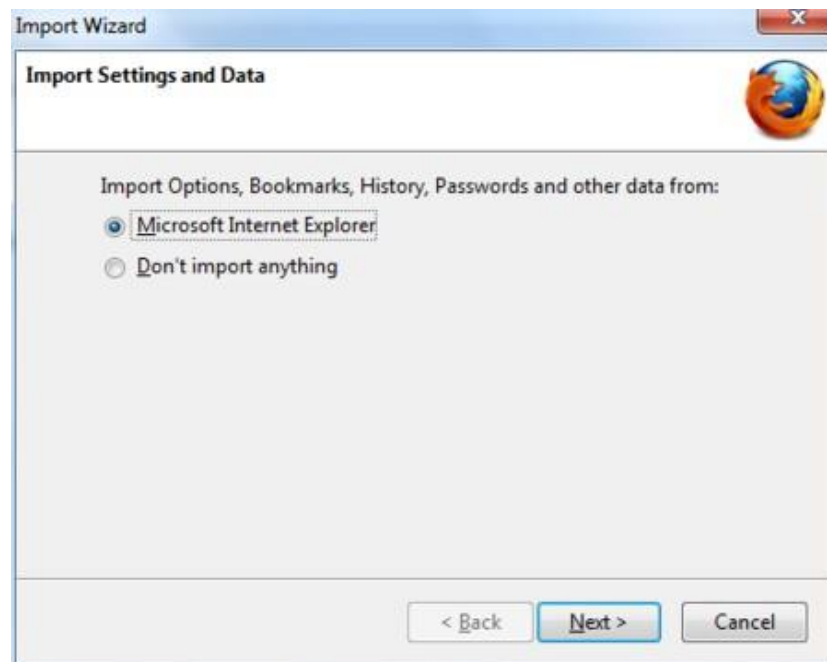


Figure 8.12:Windows Firefox Install

8. Once Firefox has been installed, click **Finish** to close the setup wizard.

If the **Launch Firefox now** check box is checked, Firefox will start after you click **Finish**. Otherwise you can launch Firefox through the start menu.

Windows Vista Users

If at any time throughout the installation process you are prompted with a User Account Control (UAC) window, press Continue, Allow, or Accept.

8.4.1 Troubleshooting

If you have problems starting Firefox, see <https://support.mozilla.com/kb/Firefox+will+not+start>

8.5 Extending Firefox

When you first download and install Firefox, it can handle basic browser tasks immediately. You can also add extra capabilities or change the way Firefox behaves by installing add-ons, small additions that extend Firefox's power.

Firefox extensions can pimp your browser, but they can also collect and transmit information about you. Before you install any add-on, keep in mind to choose add-ons from trusted sources. Otherwise, an add-on might share information about you without your knowing, keep a record on the sites you have visited, or even harm your computer.

There are several kinds of add-ons:

- *Extensions* add functionality to Firefox
- *Themes* change the appearance of Firefox.
- *Plugins* help Firefox handle things it normally can't process (i.e. Flash movies, Java applications).

For the topics covered in this book we are only going to need extensions. We will look at some add-ons that are particularly relevant for dealing with Internet security. The variety of available extensions is enormous. You can add dictionaries for different languages, track the weather in other countries, get suggestions for Web sites that are similar to the one you are currently viewing, and much more. Firefox keeps a list of current extensions on its site (<https://addons.mozilla.org/firefox/>), or you can browse them by category at <https://addons.mozilla.org/firefox/browse>.

Caution: We recommend that you never install an add-on for Firefox unless it is available from the Firefox add-on pages. You should also never install Firefox unless you get the installation files from a trusted source. It is important to note that using Firefox on someone else's computer or in an Internet caf increases your potential vulnerability. Know that you can take Firefox on a CD or USB-stick (check our chapter on that issue). While no tool can protect you completely against all threats to your online privacy and security, the Firefox extensions described in this chapter can significantly reduce your exposure to the most common ones, and increase your chances of remaining anonymous.

8.5.1 HTTPS Everywhere

HTTP is considered unsafe, because communication is transmitted in plain text. Many sites on the Web offer some support for encryption over HTTPS, but make it difficult to use. For instance, they may connect you to HTTP by default, even when HTTPS is available, or they may fill encrypted pages with links that go back to the unencrypted site. The HTTPS Everywhere extension fixes these problems by rewriting all requests to these sites to HTTPS. Although the extension is called "HTTPS Everywhere", it only activates HTTPS on a particular list of sites and can only use HTTPS on sites that have chosen to support it. It cannot make your connection to a site secure if that site does not offer HTTPS as an option.

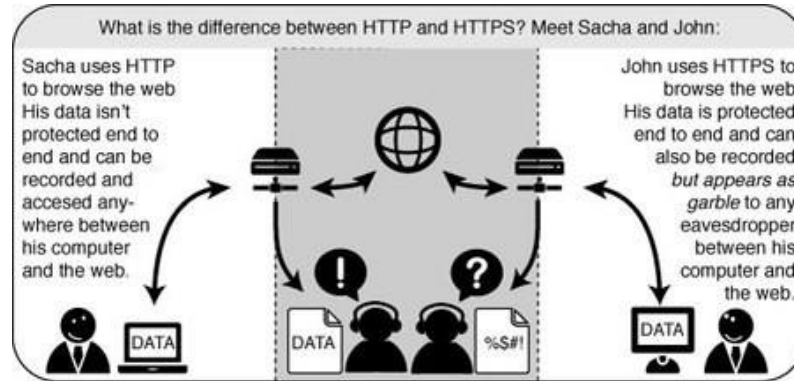


Figure 8.13:HTTPS Schema

Please note that some of those sites still include a lot of content, such as images or icons, from third party domains that is not available over HTTPS. As always, if the browser's lock icon is broken or carries an exclamation mark, you may remain vulnerable to some adversaries that use active attacks or traffic analysis. However, the effort required to monitor your browsing should still be usefully increased.

Some Web sites (such as Gmail) provide HTTPS support automatically, but using HTTPS Everywhere will also protect you from TLS/SSL-stripping attacks, in which an attacker hides the HTTPS version of the site from your computer if you initially try to access the HTTP version.

Additional information can be found at: <https://www.eff.org/https-everywhere>.

8.5.2 Installation

First, download the HTTPS Everywhere extension from the official Web site: <https://www.eff.org/https-everywhere>

Select the newest release. In the example below, version 2.2 of HTTPS Everywhere was used. (A newer version may be available now.)

Click on "Allow". You will then have to restart Firefox by clicking on the "Restart Now" button. HTTPS Everywhere is now installed.



HTTPS Everywhere

HTTPS Everywhere is a Firefox and Chrome extension that encrypts your communications with many major websites, making your browsing more secure.
Encrypt the web: Install HTTPS Everywhere today.

[HTTPS Everywhere](#)
[FAQ](#)
[Creating HTTPS Everywhere Rulesets](#)
[Hack On The Code](#)
[How to Deploy HTTPS Correctly](#)



Install in Firefox
Version 2.2 Stable



Install in Chrome
Alpha Version

Figure 8.14:HTTPS Everywhere



Figure 8.15:HTTPS Everywhere

8.5.3 Configuration

To access the HTTPS Everywhere settings panel in Firefox 4 (Linux), click on the Tools menu at the top of your screen and then select Add-ons. (Note that in different versions of Firefox and different operating systems, the Add-ons Manager may be located in different places in the interface.)



Figure 8.16:HTTPS Everywhere Click on the

Preferences button.

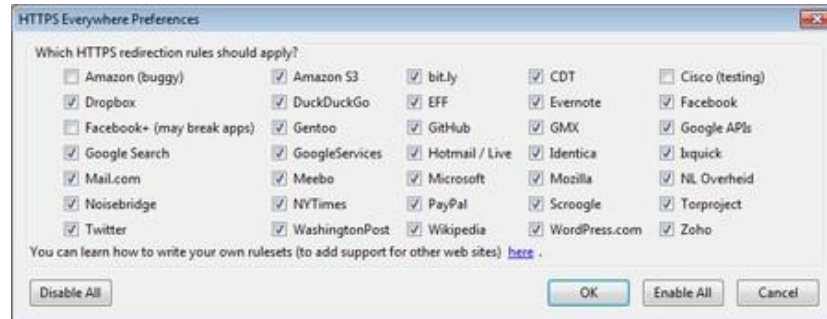


Figure 8.17:HTTPS Everywhere

A list of all supported Web sites where HTTPS redirection rules should be applied will be displayed. If you have problems with a specific redirection rule, you can uncheck it here. In that case, HTTPS Everywhere will no longer modify your connections to that specific site.

8.5.4 Usage

Once enabled and configured, HTTPS Everywhere is very easy and transparent to use. Type an insecure HTTP URL (for example, <http://www.google.com>).

Press Enter. You will be automatically redirected to the secure HTTPS encrypted Web site (in this example: <https://encrypted.google.com>). No other action is needed.

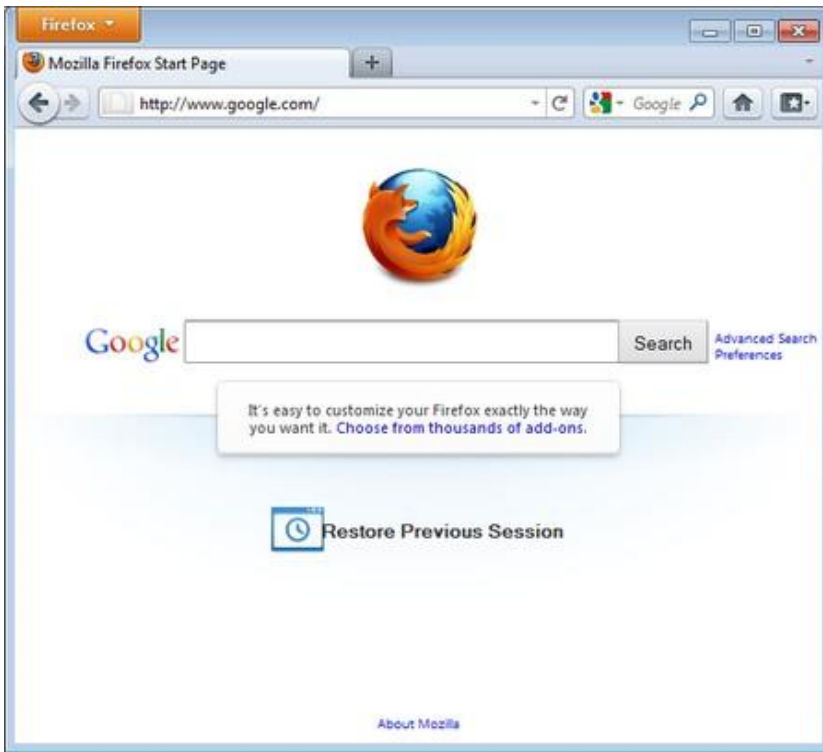


Figure 8.18:HTTPS Everywhere



Figure 8.19:HTTPS Everywhere

8.5.5 *If networks block HTTPS*

Your network operator may decide to block the secure versions of Web sites in order to increase its ability to spy on what you do. In such cases, HTTPS Everywhere could prevent you from using these sites because it forces your browser to use only the secure version of these sites, never the insecure version. (For example, we heard about an airport WiFi network where all HTTP connections were permitted, but not HTTPS connections. Perhaps the WiFi operators were interested in watching what users did. At that airport, users with HTTPS Everywhere were not able to use certain Web sites unless they temporarily disabled HTTPS Everywhere.)

In this scenario, you might choose to use HTTPS Everywhere together with a circumvention technology such as Tor or a VPN in order to bypass the network's blocking of secure access to Web sites.

8.5.6 *Adding support for additional sites in HTTPS Everywhere*

You can add your own rules to the HTTPS Everywhere add-on for your favorite Web sites. You can find out how to do that at: <https://www.eff.org/https-everywhere/rulesets>. The benefit of adding rules is that they teach HTTPS Everywhere how to ensure that your access to these sites is secure. But remember: HTTPS Everywhere does not allow you to access sites securely unless the site operators have already chosen to make their sites available through HTTPS. If a site does not support HTTPS, there is no benefit to adding a ruleset for it.

If you are managing a Web site and have made an HTTPS version of the site available, a good practice would be to submit your Web site to the official HTTPS Everywhere release.

8.5.7 Enforcing secure HTTPS server connections

Even if you instruct your browser to use the HTTPS protocol when communicating with a web server, it is still possible that the server (due to unsecure configuration on its own side) enforces a unsecure SSL cipher protocol for the connection. The only way to prevent this is by telling the browser to not accept such unsecure SSL protocols (like those based on RC4 encryption).

To disable RC4 encryption for HTTPS connections you have to switch those off in Firefox. In an empty address bar type “about:config”, press return and close the warning dialog displayed next (you can disable this dialog if you want for the next time you configure Firefox). In the search field enter “rc4” and look at the list displayed as a search result:

Any entry with a “true” in the last column (“Value” field) is activated and should be de-activated. Simply right-click on the entry and “Toggle” the value field to false. Proceed for all entries until all of them have a value of “false”.

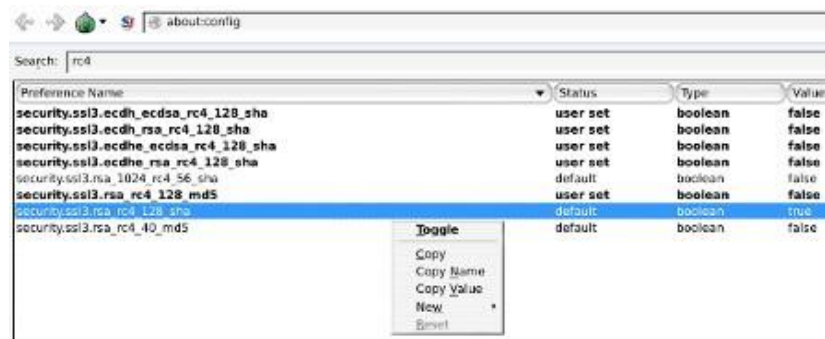


Figure 8.20:Disable RC4

8.5.8 Adblock Plus

Adblock Plus (<http://www.adblockplus.org>) is mainly known for blocking advertisements on websites. But it also can be used to block other content that may try to track you. To keep current with the latest threats, Adblock Plus relies on blacklists maintained by volunteers.

Extra Geek info: How does Adblock Plus block addresses?

The hard work here is actually done by Gecko, the engine on top of which Firefox, Thunderbird and other applications are built. It allows something called “content policies”. A content policy is simply a JavaScript (or C++) object that gets called whenever the browser needs to load something. It can then look at the address that should be loaded and some other data and decide whether it should be allowed. There is a number of built-in content policies (when you define which sites shouldn’t be allowed to load images in Firefox or SeaMonkey, you are actually configuring one of these built-in content policies) and any extension can register one. So all that Adblock Plus has to do is to register its content policy, other than that there is only application logic to decide which addresses to block and user interface code to allow configuration of filters.

8.5.9 Getting started with Adblock Plus

Once you have Firefox installed:

1. Download the latest version of Adblock Plus from the Add-On database of Firefox
2. Confirm that you want Adblock Plus by clicking “Install Now”.
3. After Adblock Plus has been installed, Firefox will ask to restart.

8.5.10 Choosing a filter subscription

Adblock Plus by itself doesn't do anything. It can see each element that a Web site attempts to load, but it doesn't know which ones should be blocked. This is what Adblock's filters are for. After restarting Firefox, you will be asked to choose a filter subscription (free).

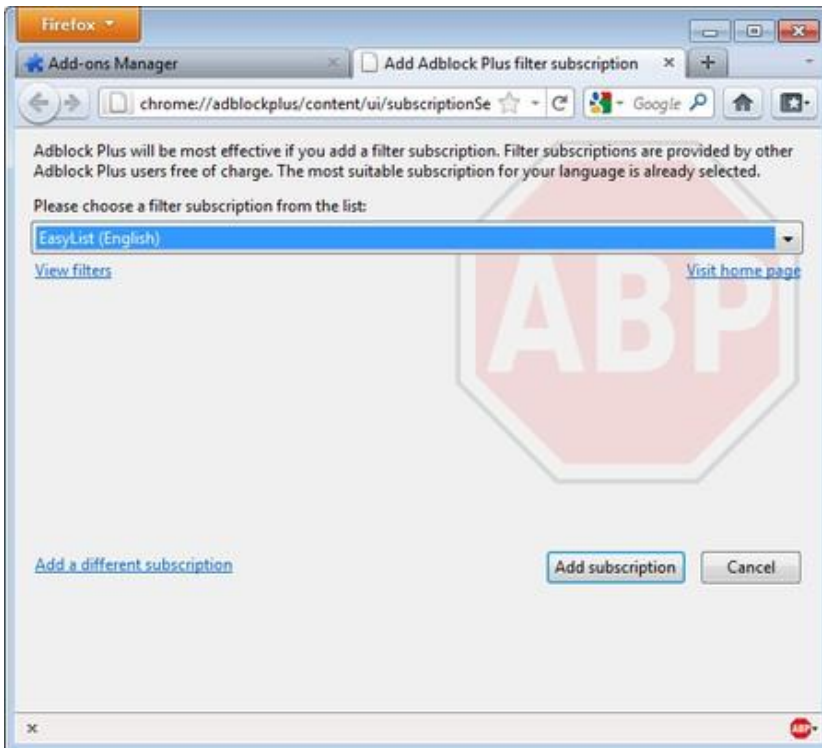


Figure 8.21:Ad Block Plus

Which filter subscription should you choose? Adblock Plus offers a few in its dropdown menu and you may wish to learn about the strengths of each. A good filter to start protecting your privacy is EasyList (also available at <http://easylist.adblockplus.org/en>).

As tempting as it may seem, don't add as many subscriptions as you can get, since some may overlap, resulting in unexpected outcomes. EasyList (mainly targeted at English- language sites) works well with other EasyList extensions (such as region-specific lists like RuAdList or thematic

lists like EasyPrivacy). But it collides with Fanboy’s List (another list with main focus on English-language sites).

You can always change your filter subscriptions at any time within preferences. Once you’ve made your changes, click OK.

8.5.11 Creating personalized filters

AdBlock Plus also lets you create your own filters, if you are so inclined. To add a filter, start with Adblock Plus preferences and click on “Add Filter” at the bottom left corner of the window. Personalized filters may not replace the benefits of well-maintained blacklists like EasyList, but they’re very useful for blocking specific content that isn’t covered in the public lists. For example, if you wanted to prevent interaction with Facebook from other Web sites, you could add the following filter:

```
||facebook.*$domain=~facebook.com!~127.0.0.1
```

The first part (||facebook.*) will initially block everything coming from Facebook’s domain. The second part (\$domain=~facebook.com!~127.0.0.1) is an exception that tells the filter to allow Facebook requests only when you are in Facebook or if the Facebook requests come from 127.0.0.1 (your own computer) in order to keep certain features of Facebook working.

A guide on how to create your own Adblock Plus filters can be found at <http://adblockplus.org/en/filters>.

8.5.12 Enabling and disabling AdBlock Plus for specific elements or Web sites

You can see the elements identified by AdBlock Plus by clicking on the ABP icon AdBlock Plus icon in your browser (usually next to the search bar) and selecting “Open blockable items”. A window at the bottom of your browser will let you enable or disable each element on a case-by-case basis. Alternatively, you can disable AdBlock Plus for a specific domain or page by clicking on the ABP icon and ticking the option “Disable on [domain name]” or “Disable on this page only”.

8.5.13 Other extensions that can improve your security

Below is a short list of extensions that are not covered in this book but are helpful to further protect you.

- **Flagfox** - puts a flag in the location bar telling you where the server you are visiting is most probably located. <https://addons.mozilla.org/en-US/firefox/addon/flagfox/>
- **BetterPrivacy** - manages “cookies” used to track you while visiting websites. Cookies are small bits of information stored in your browser. Some of them are used to track the sites you are visiting by advertisers. <https://addons.mozilla.org/en-US/firefox/addon/betterprivacy/>
- **GoogleSharing** - If you are worried that google knows your search history, this extension will help prevent that. <https://addons.mozilla.org/en-us/firefox/addon/googlesharing/>
- **NoScript** - Although not friendly for beginners, this addon will block scripts and third party

plugin content (eg, Adobe Flash) unless specifically allowed by the user, it also provides general protection against simple cross site scripting vectors. <http://noscript.net/>

- **User Agent Switcher** - Your browser supplies large amounts of identifying information to any remote server through the 'User-Agent' header, including Operating System and specific version information. This addon allows you to supply either a fake or generic User-Agent to the server. <http://chrispederick.com/work/user-agent-switcher/>

8.6 Proxy Settings

A proxy server allows you to reach a Web site or other Internet location even when direct access is blocked in your country or by your ISP. There are many different kinds of proxies, including:

- Web proxies, which only require that you know the address to the proxy Web site, which may have a URL similar to <http://proxy.com/cgi-bin/nph-proxy.cgi>
- HTTP proxies, which require that you modify your Browser settings. HTTP proxies only work for Web content. You may get the information about a HTTP proxy in the format proxy.example.com:3128 or 192.168.0.1:8080.
- SOCKS proxies, which also require that you modify your Browser settings. SOCKS proxies work for many different Internet applications, including e-mail and instant messaging tools. The SOCKS proxy information looks just like HTTP proxy information.

You can use a Web proxy directly without any configuration by typing in the URL. The HTTP and SOCKS proxies, however, have to be configured in your Web browser.

8.6.1 Default Firefox proxy configuration

In Firefox you can change the settings for using a proxy. You'll need to open the Options or Preferences window of Firefox. You can find this in the menu, by clicking on the top of the Window and selecting Edit > Preferences on Linux or Tools > Options on Windows.

Go to the Network section and open the Advanced tab.

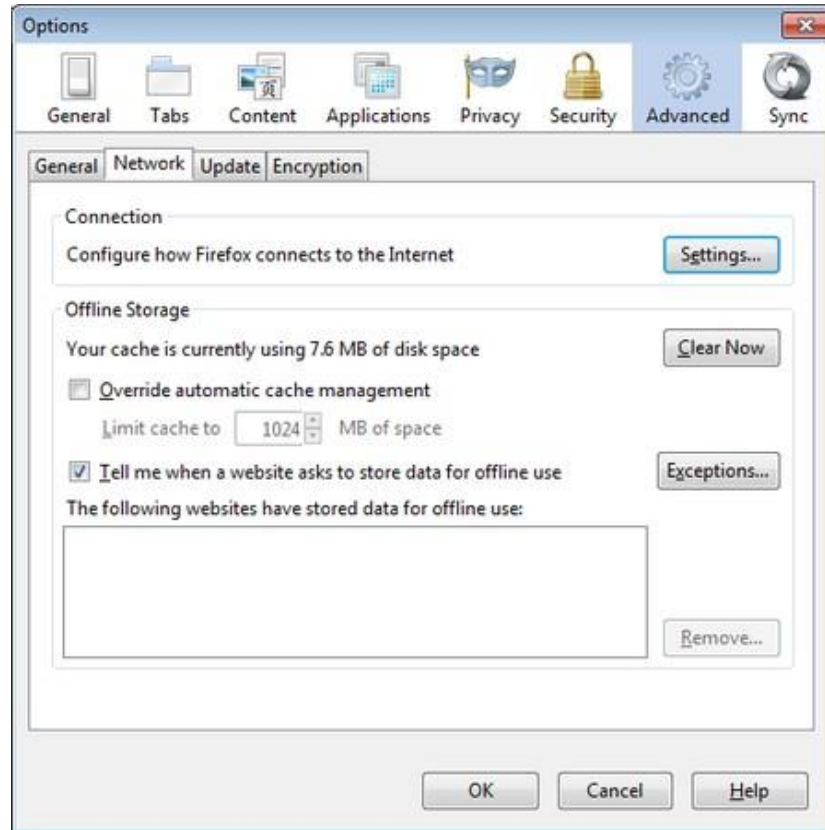


Figure 8.22:Firefox Proxy Settings

Select Settings, click on “Manual proxy configuration” and enter the information of the proxy server you want to use. Please remember that HTTP proxies and SOCKS proxies work differently and have to be entered in the corresponding fields. If there is a colon (:) in your proxy information, that is the separator between the proxy address and the port number. Your screen should look like this:

After you click OK, your configuration will be saved and your Web browser will automatically connect through that proxy on all future connections. If you get an error message such as, “The proxy server is refusing connections” or “Unable to find the proxy server”, there is a problem with your proxy configuration. In that case, repeat the steps above and select “No proxy” in the last screen to deactivate the proxy.

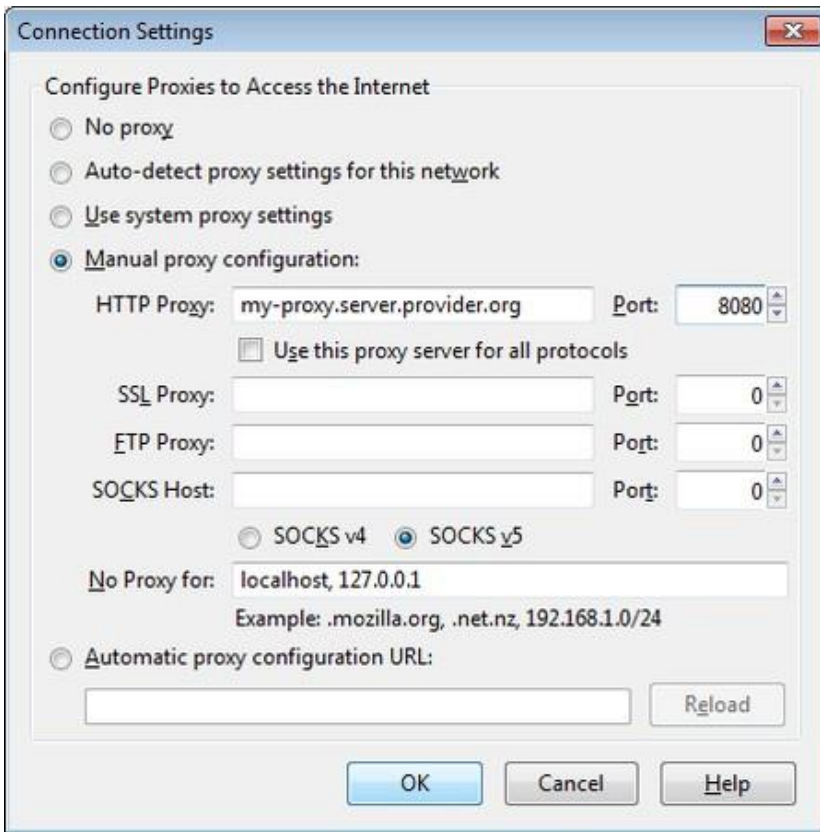


Figure 8.23:Firefox Proxy Settings

8.7 Using Tor?

Tor is a system intended to enable online anonymity, composed of client software and a network of servers which can hide information about users' location and other factors which might identify them. Imagine a message being wrapped in several layers of protection: every server is only able to take off one layer, thereby immediately deleting the sender information of the previous server.

If Alice wants to view Bob's website, instead of directly talking to Bob like this:

Alice -> Bob

This is fine, and Alice and Bob can use end to end cryptography to ensure privacy, integrity and authenticity of their communications, however if Alice does not want Bob to know that she is viewing his website or does not want Eve (a hypothetical evesdropper, on Alice or Bobs side of the connection) to know she and Bob are communicating, extra steps must be taken.

Alice instead makes an encrypted connection to an EntryNode on the Tor network, here they establish a TLS connection and the EntryNode allows Alice to make a further connection through it. Once again a TLS connection is established this time through the already encrypted EntryNode connection, and once again the process is repeated through the RelayNode to the ExitNode. Now, Alice is encrypting the data 3 times, first for the ExitNode, then for the RelayNode and finally for the EntryNode. Creating a network path like this:

Alice -> EntryNode -> RelayNode -> ExitNode -> Bob

When the EntryNode receives Alice's data it is still encrypted for the RelayNode and ExitNode, it knows where the data is coming from but not the final destination or the message content, it then passes the data on to the RelayNode. The RelayNode receives the data but it is still encrypted for the ExitNode, it knows it is coming from the EntryNode and going to the ExitNode but does not know the final destination or the original sender, it then passes the data to the ExitNode which removes the final layer of encryption, the ExitNode knows what the message is, that it came from RelayNode but does not know where it entered the network, or who sent it.

This layered approach is where Tor derives its name from (The Onion Router), each 'layer' knows only of the layer on either side of it, meaning that none in the chain except Alice knows the full path the data is taking, however Alice, Bob and the ExitNode are all able to see the message content, thus end-to-end cryptography is still required to ensure privacy, integrity and authenticity of the communications across the Tor network.

Use of this system makes it more difficult to trace internet traffic to the user, including visits to Web sites, online posts, instant messages, and other communication forms. It is intended to protect users' personal freedom, privacy, and ability to conduct confidential business, by keeping their internet activities from being monitored. The software is open-source and the network is free of charge to use.

8.7 Using Tor?

Like all current low latency anonymity networks, Tor cannot and does not attempt to protect against monitoring of traffic at the boundaries of the Tor network, i.e., the traffic entering and exiting the network. While Tor does provide protection against traffic analysis, it cannot prevent traffic confirmation (also called end-to-end correlation)

Caution: As Tor does not, and by design cannot, encrypt the traffic between an exit node and the target server, any exit node is in a position to capture any traffic passing through it which does not use end-to-end encryption such as TLS. (If your postman is corrupt he might still open the envelope and read the content). While this may or may not inherently violate the anonymity of the source, if users mistake Tor's anonymity for end-to-end encryption they may be subject to additional risk of data interception by third parties. So: the location of the user remains hidden; however, in some cases content is vulnerable for analysis through which identifying or revealing information about the user may be gained.

8.7.1 Using Tor Browser Bundle

The Tor Browser Bundle lets you use Tor on Windows, OSX and/or Linux without requiring you to configure a Web browser. Even better, it's a portable application that can be run from a USB flash drive, allowing you to carry it to any PC without installing it on each computer's hard drive.

8.7.2 Downloading Tor Browser Bundle

You can download the Tor Browser Bundle from the torproject.org Web site (<https://www.torproject.org>).

If the torproject.org Web site is filtered from where you are, type "tor mirrors" in your favorite Web search engine: The results probably include some alternative addresses to download the Tor Browser Bundle.

Please follow the instructions on the Tor Project Website on how to install the Tor Browser.

Caution: When you download Tor Bundle (plain or split versions), you should check the signatures of the files. This step ensures that the files have not been tampered with. To learn

more about signature files and how to check them, read <https://www.torproject.org/docs/verifying-signatures>

8.7.3 Running a Relay or Bridge

Tor is a network of volunteers who run relays and bridges. If you would like to grow the Tor network by contributing bandwidth and spare CPU cycles, consider running a relay. Plus, running a relay may improve your anonymity since an attacker can't distinguished between traffic that originated from you or from the relay. See the [Tor FAQ](#) for more details.

However, if you do run a relay, your IP address will be listed on the Internet as a Tor relay. Tor clients depend on this list, provided by Tor Directory Servers, so that they can build circuits. If you wish to contribute to Tor, but do not want to run a public relay,

consider running a bridge. Since Tor relays are public, some ISP block access to the Tor network by blocking *all the relays*. Tor Bridges are unlisted are therefore, more difficult to find.

Tor's goal is to protect anonymity on the Internet, but sometimes Tor is used for illegal purposes. As a relay operator, consult the [Legal FAQ](#), written by the Electronic Frontier Foundation (EFF). The EFF is a U.S. based non-profit organization whose mission is to "protect your digital right." Other countries should seek the advice of similar organizations. However, legal risks can be minimized by running a non-exit relay or bridge.

If you'd like to configure your computer to run a relay or a bridge, visit the [Tor website](#) for thorough instructions.

8.8 Extending Google Chrome

Chrome is Google's browser. Here are some useful tips and extensions:

8.8.1 Disabling Instant Search

Chrome can search as you type. The advantage of this is that you get search suggestions and can use Google's predictions - but the disadvantage is that every character you type is sent to Google's servers, where it may be logged.

To disable, open Chrome's settings by clicking the menu button at the right of the address bar and clicking Settings. Or, simply type `chrome://settings/` in your address bar.

Ensure that the **Enable Instant for faster searching (omnibox input may be logged)** checkbox is unchecked.

8.8.2 Adblock for Chrome

Just like Firefox, Adblock removes ads. Install from [this Chrome Webstore page](#).

8.8.3 HTTPS Everywhere

Forces encrypted https connections wherever possible. Installation link can be found on the [EFF HTTPS Everywhere homepage](#).

8.8.4 PrivacyFix

PrivacyFix (beta) gives you a dashboard view of your privacy settings on Facebook and Google, as well as Do-Not-Track headers and tracking cookies. It provides links to quickly change these

privacy settings without digging through many drilldown pages. Install from the [Chrome web store page](#)

9 Passwords

9.1 Keeping passwords safe

Passwords are like keys in the physical world. If you lose a password you will not be able to get in, and if others copy or steal it they can use it to enter. A good password should not be easy for others to guess and not easy to crack with computers, while still being easy for you to remember.

9.1.1 Password length and complexity

To protect your passwords from being guessed, length and complexity are important. Passwords like the name of your pet or a birth date are very unsafe, as is using single word that can be found in a dictionary. Do not use a password containing only numbers. Most importantly a secure password is long. Using combinations of lower case letters, capitals, numbers and special characters can improve the security, but length is still the most important factor.

For use with important accounts like the pass phrase which protects your PGP/GPG or TrueCrypt encrypted data, or the password for your main email account, use 20 characters or more, the longer the better. See [this XKCD cartoon](#) "correct horse battery staple" vis-À-vis "Tr0ub4dor&3" for an explanation.

9.1.2 Easy to remember and secure passwords

One way to create strong and easy to remember passwords is to use sentences.

A few examples:

- IloveDouglasAdamsbecausehe'sreallyawesome.
- Peoplelovemachinesin2029A.D.
- BarneyfromHowIMetYourMotherisAWESOME!

Sentences are easy to remember, even if they are 50 characters long and contain uppercase characters, lowercase characters, symbols and numbers.

9.1.3 Minimizing damage

It is important to minimize the damage if one of your passwords is ever compromised. Use different passwords for different websites or accounts, that way if one is compromised, the others are not. Change your passwords from time to time, especially for accounts you consider to be sensitive. By doing this you can block access to an attacker who may have learned your old password.

9.1.4 Using a password manager

Remembering a lot of different passwords can be difficult. One solution is to use a dedicated application to manage most of your passwords. The next section in this chapter will discuss *Keepass*,

a free and open source password manager with no known vulnerabilities, so long as you chose a sufficiently long and complex “master password” to secure it with.

For website passwords only, another option is the built-in password manager of the Firefox browser. Make sure to set a master password, otherwise this is very insecure!

9.1.5 Physical protection

When using a public computer such as at a library, an internet cafe, or any computer you do not own, there are several dangers. Using “over the shoulder” surveillance, someone, possibly with a camera, can watch your actions and may see the account you log in to and the password you type. A less obvious threat is software programs or hardware devices called “keystroke loggers” that record what you type. They can be hidden inside a computer or a keyboard and are not easily spotted. Do not use public computers to log in to your private accounts, such as email. If you do, change your passwords as soon as you get back to a computer you own and trust.

9.1.6 Other caveats

Some applications such as chat or mail programs may ask you to save or “remember” your username and password, so that you don’t have to type them every time the program is opened. Doing so may mean that your password can be retrieved by other programs running on the machine, or directly from your hard disk by someone with physical access to it.

If your login information is sent over an insecure connection or channel, it might fall into the wrong hands. See the chapters on secure browsing for more information.

9.2 Installing KeePass

We will cover installing KeePass on Ubuntu and Windows.

Mac OSX comes with an excellent built-in password manager called Keychain that is just as safe. Downsides are that it isn’t Open Source and doesn’t work on other systems. If you’d need to take your passwords from one Operating System to another it is better to stick with KeePass after all. How to use Keychain is covered in the next chapter.

9.2.1 Installing KeePassX on Ubuntu

To install on Ubuntu we will use the Ubuntu Software Center. Type KeePass in the search field at the top right and the application KeePassX should automatically appear in the listing.

Highlight the item (it may already be highlighted by default) and then press ‘Install’. You will be asked to Authorise the installation process:



Figure 9.1: KeePass Install

Enter your password and press ‘Authenticate’ the installation process will then begin. Ubuntu does not offer very good feedback to show the software is installed. If the green progress indicator on the left has gone and the progress bar on the right has gone then you can assumed the software is installed.

9.2.2 Installing KeePass on Windows

First visit the [KeePass download webpage](#) and choose the appropriate installer. For this chapter we are using the [current installer](#).

Download this to your computer then double click on the installer. You will first be asked to select a language, we will choose English:

Press ‘OK’ and you will be shown the following screen: Just press ‘Next >’ and go to the next screen:

In the screen shown above we must select ‘I accept the agreement’ otherwise we will not be able to install the software. Choose this option and then press ‘Next >’. In the next screen you will be asked to determine the installation location. You can leave this with the defaults unless you have good reason to change them.

Click on ‘Next >’ and continue.

The above image shows the KeePass components you can choose from. Just leave the defaults as they are and press ‘Next >’. You will come to a new screen:

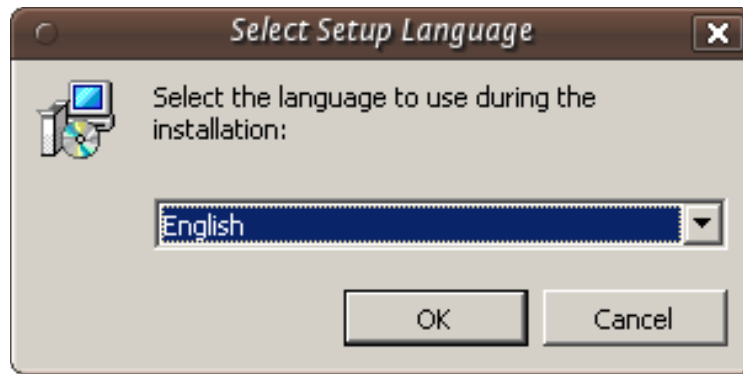


Figure 9.2:KeePass Install



Figure 9.3:KeePass Install



Figure 9.4: KeePass Install



Figure 9.5:Keepass Install



Figure 9.6: KeePass Install



Figure 9.7:Keepass Install

This doesn't do anything but give you a summary of your options. Press 'Install' and the installation process will begin.

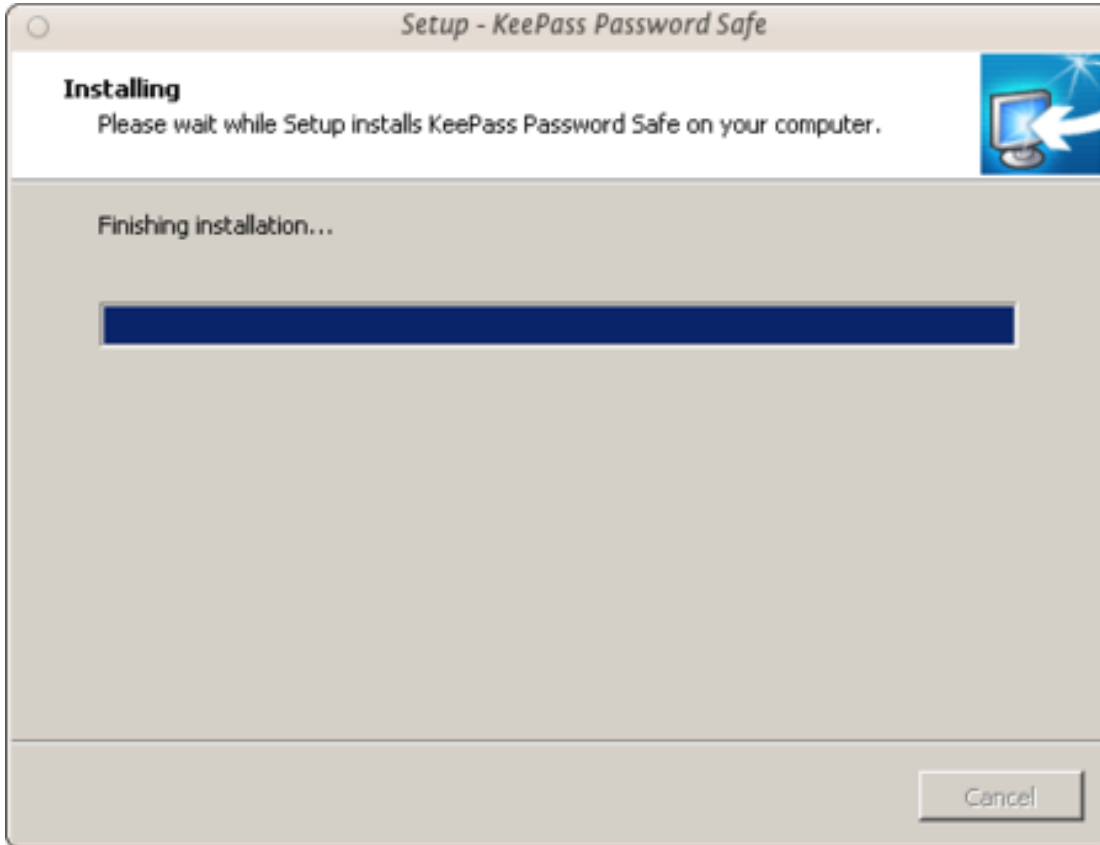


Figure 9.8: KeePass Install

9.2.3 Installing KeePass on Mac OS X

Although Keychain in Mac OS X does an excellent job of storing your passwords, you may want to run your own password database and manager. KeePass allows this added flexibility. First visit the KeePass download webpage <http://keepass.info/download.html> and choose the appropriate installer. Although the official installers are listed at the top of the page, there are unofficial/contributed installers further down. Scroll down to find [KeePass 2.x for Mac OS X]<http://keepass2.openix.be/>:

As this is an external link, your browser will be redirected to <http://keepass2.openix.be/>:

Note here that you must install the Mono framework first, so that KeePass can run in OS X. So click on each of the links [Mono 2.10.5](#) and [KeePass2.18](#) to download the DMG files to your computer. Double-click on each of the DMGs in your downloads folder to unpack the volumes to your desktop.

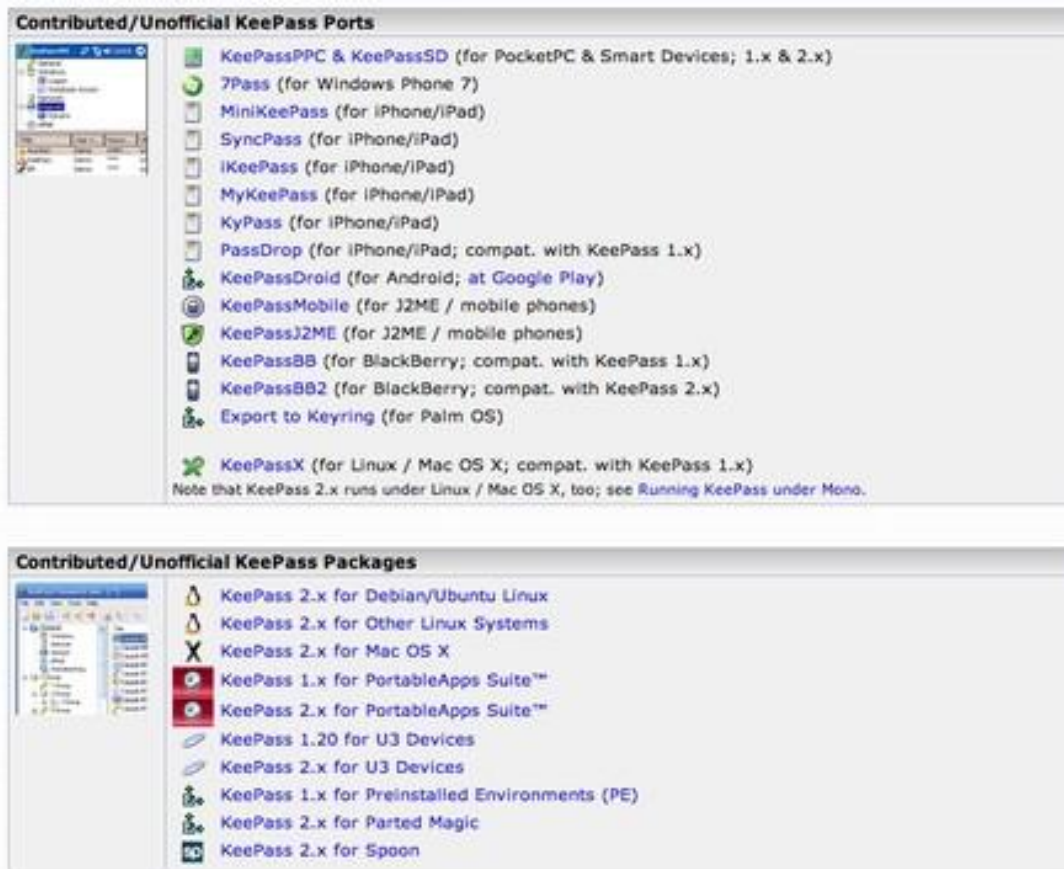


Figure 9.9:Keepass Install



Figure 9.10:Keepass Install

The Mono Package installer is in case called something similar to ‘MonoFramework- MRE-2.10.5_0.macos10.xamarin.x86.pkg’, so double-click on this document:

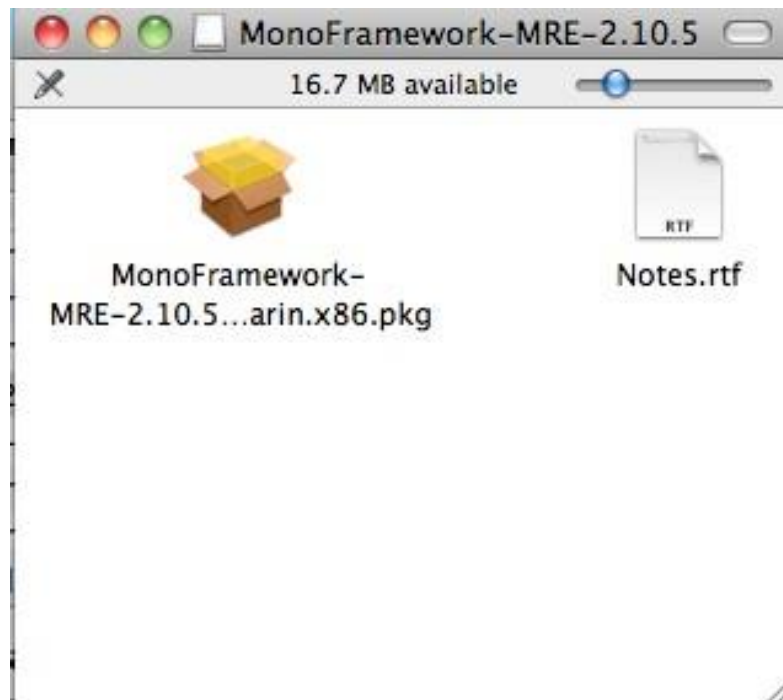


Figure 9.11:Keepass Install The

installer will open and run:

Follow each of the steps by clicking 'Continue', the next step being 'Read Me'. Inhere is important information such as all of the files that the package will install, including information on how to uninstall Mono:

Click 'Continue' to the next screen, the license. Clicking 'Continue' on the license screen pops up the agree/disagree dialogue box. If you agree with the license conditions, the installation will continue:

The following two steps in the installation ask you to choose an installation destination, and check there is enough space on the install disk. When the installation has completed, you will see this screen:

Now you can quit the installer. Next take a look at the KeePass disk image, double- click to open it, and drag the KeePass application into your Applications folder:

Now KeePass is ready to use for Mac OS X.

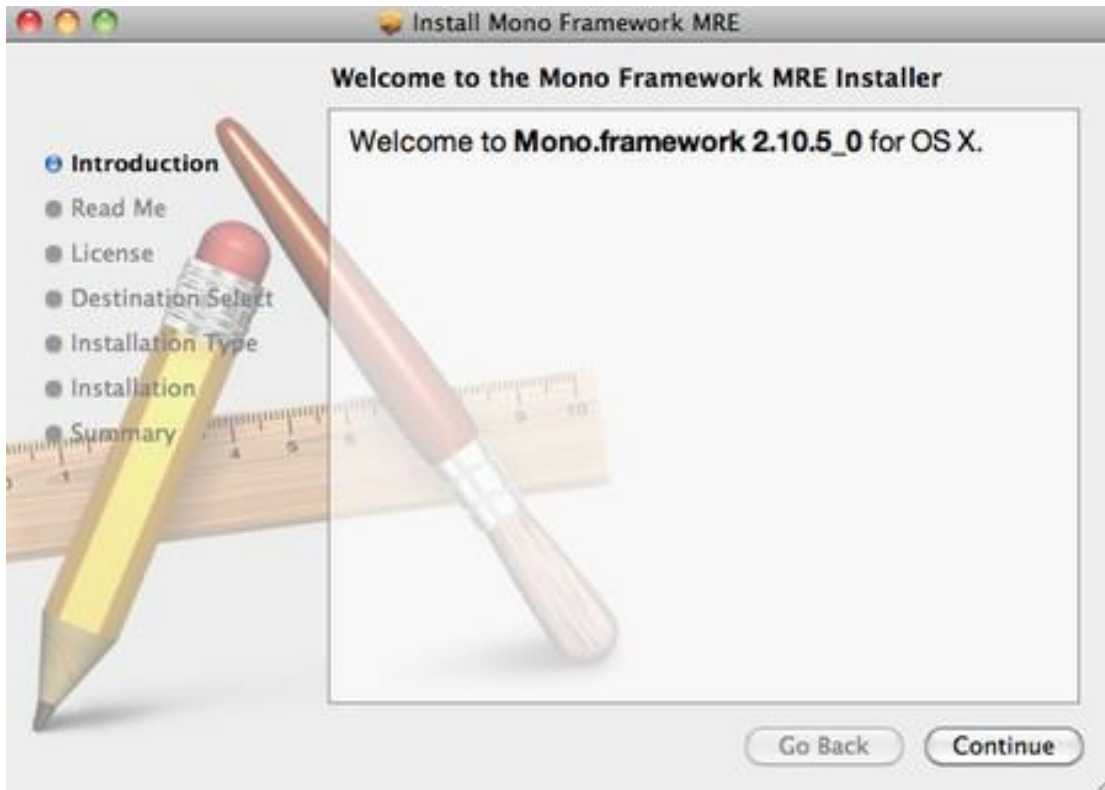


Figure 9.12:Keepass Install

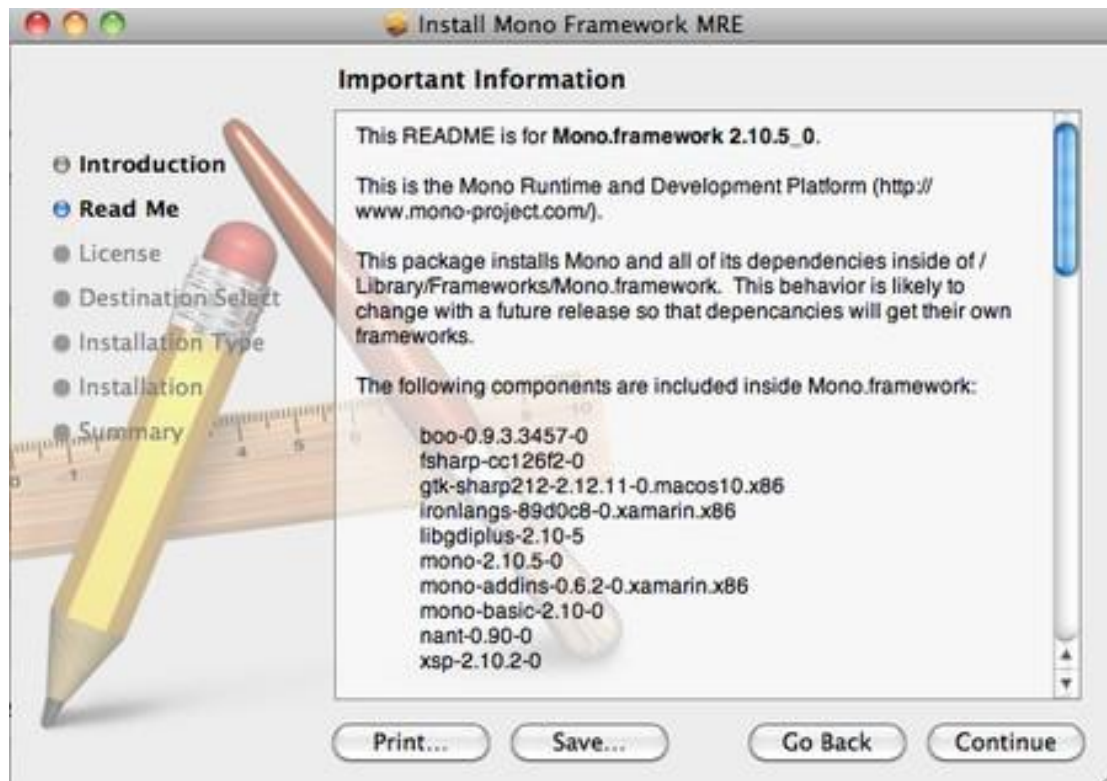


Figure 9.13:Keepass Install

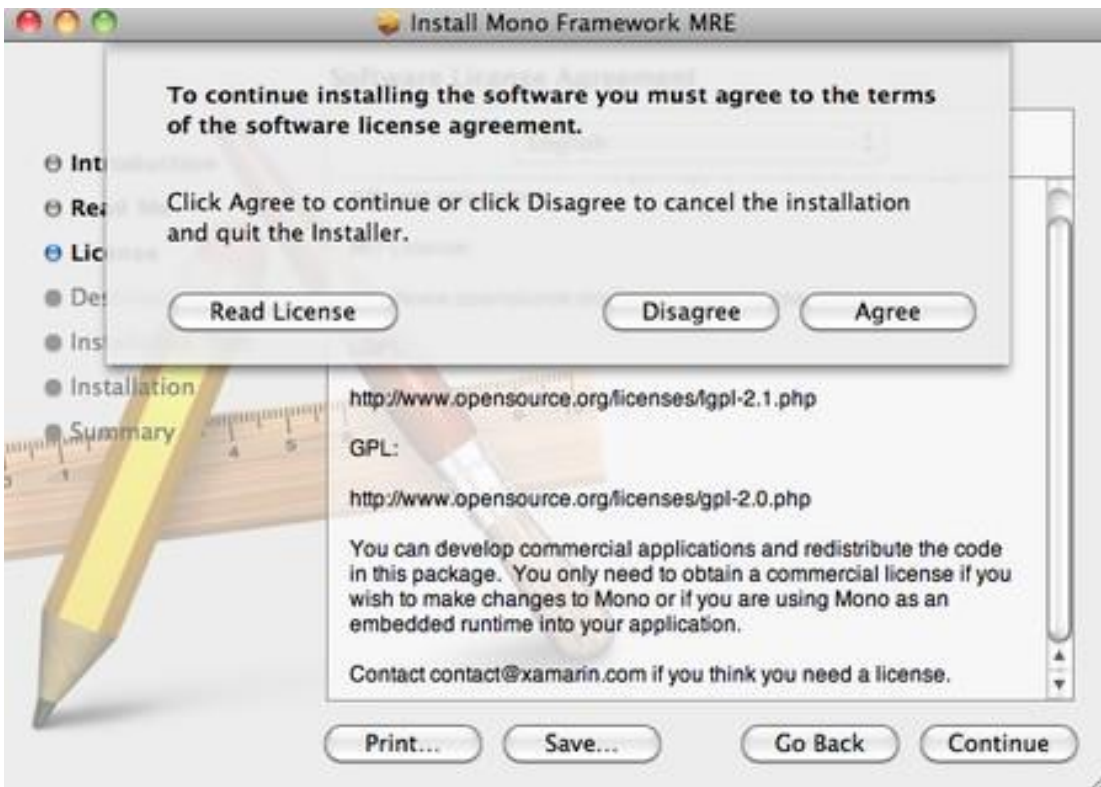


Figure 9.14:Keepass Install

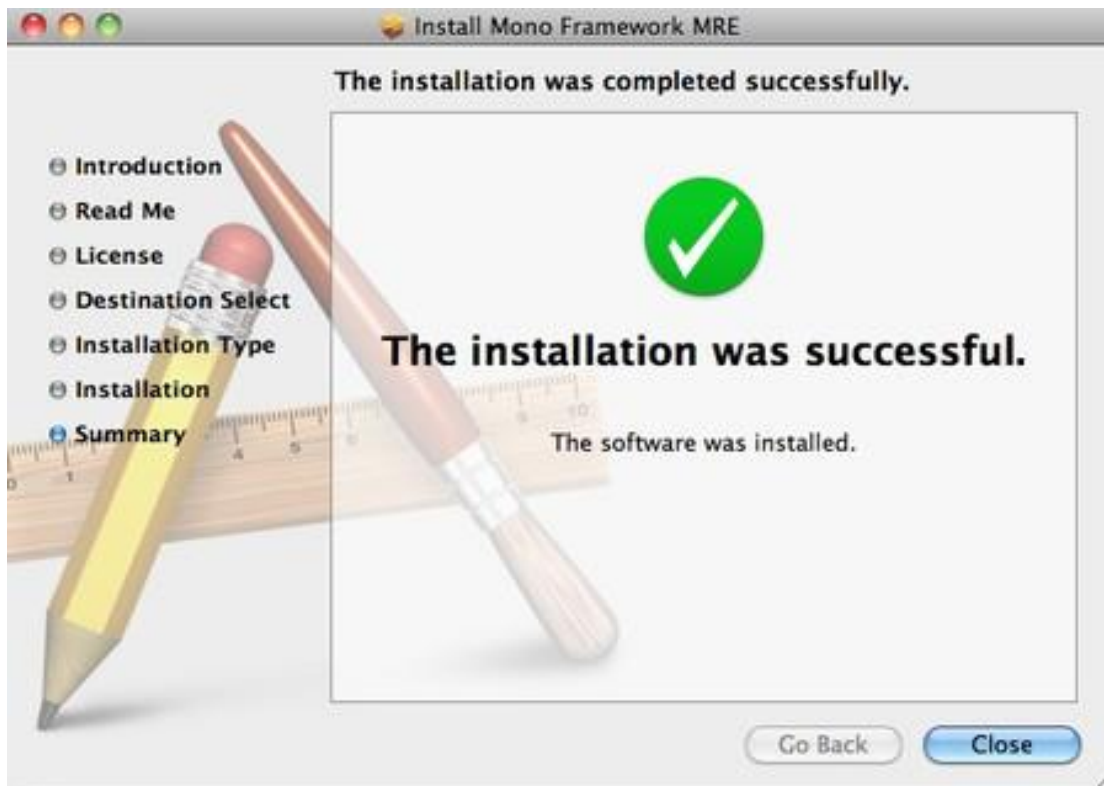


Figure 9.15:Keepass Install



Figure 9.16: KeePass Install

9.3 Encrypting Passwords with a Password Manager

To encrypt password we use KeePass on Windows and KeePassX Ubuntu, and Keychain on OSX. The basic principle is the same; you have a file on your computer which is encrypted with *one single very secure password*. This is sometimes referred to as a ‘Master Password’, ‘Admin-Password’, ‘Root-Password’ etc. but they are all *the ultimate key* to all your other keys and secure data. For this reason you can’t and shouldn’t think to light about creating this password.

If a password manager is part of your OS (like it is with OSX) it unlocks automatically for you after you login to your account and so opening secure information like passwords. For this, and other, reasons you should disable ‘Automatically Login’. When you start-up your computer you should always have to login and, even better, set your computer to automatically logout or lock the screen after a set amount of time.

9.3.1 Encrypting Passwords with KeePassX on Ubuntu

First open KeePassX from the Applications->Accessories -> KeePassX menu.

The first time you use KeePassX you need to set up a new database to store your passwords. Click on File->New Database

You will be asked to set a master key (password).

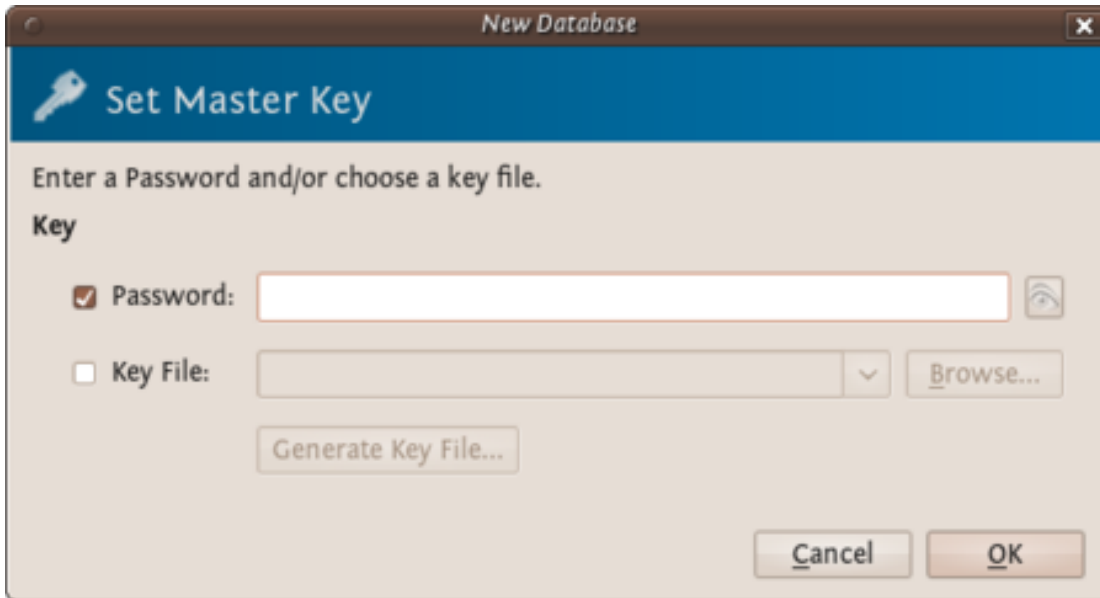


Figure 9.17: Password Manager

Choose a strong password for this field - refer to the chapter about passwords if you would like some tips on how to do this. Enter the password and press 'OK'. You then are asked to enter the password again. Do so and press 'OK'. If the passwords are the same you will see a new KeePassX 'database' ready for you to use.

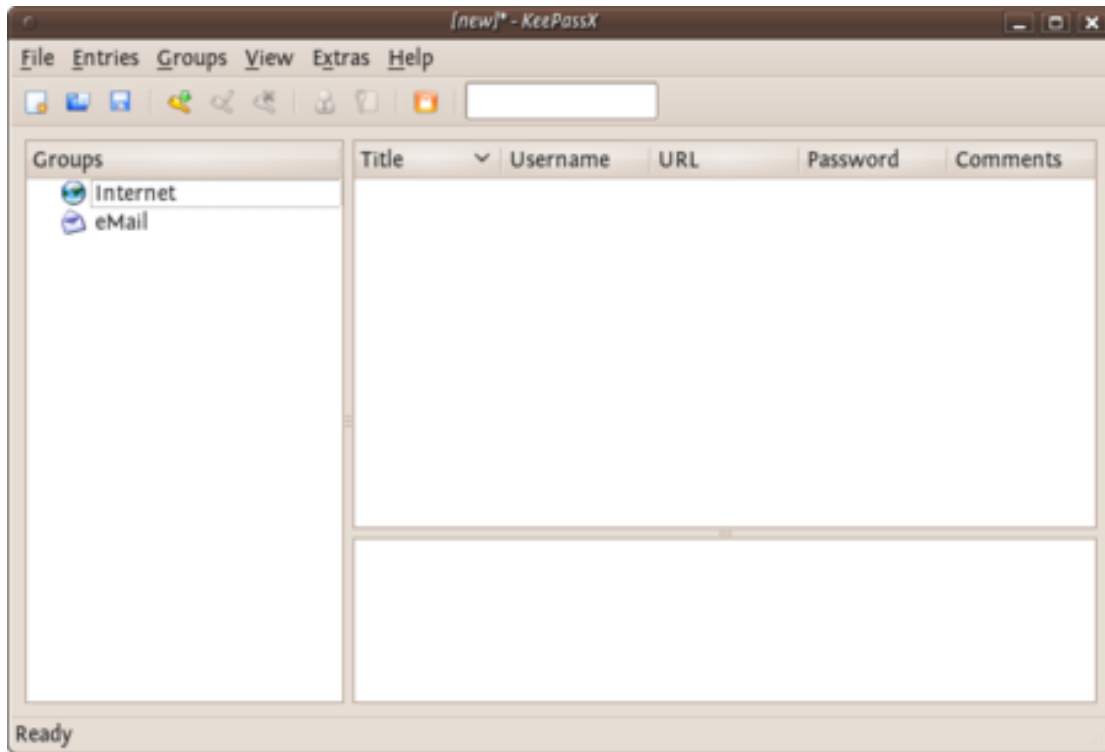


Figure 9.18:Password Manager

Now you have a place to store all your passwords and protect them by the ‘master’ password you just set. You will see two default categories ‘Internet’ and ‘Email’ - you can store passwords just under these two categories, you can delete categories, add sub- groups, or create new categories. For now we just want to stay with these two and add a password for our email to the email group. Right click on the email category and choose ‘Add New Entry. . .’:

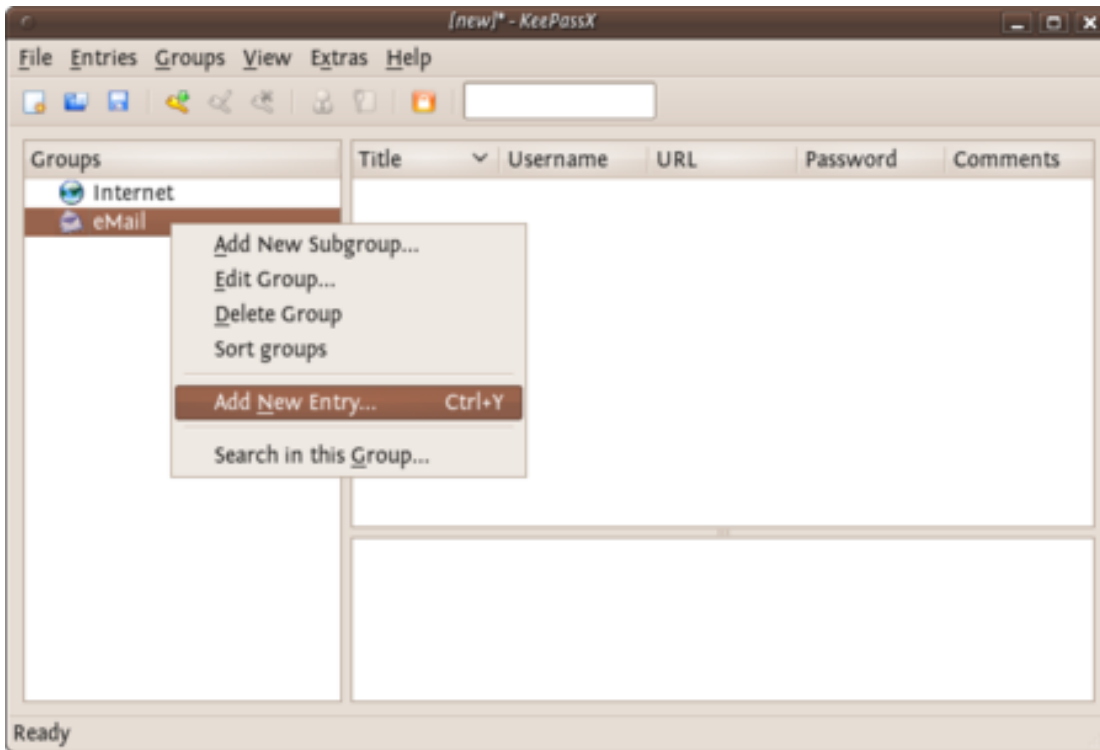


Figure 9.19: Password Manager

So now fill this form out with the details so you can correctly identify which email account the passwords are associated with. You need to fill out the fields 'Title' and the password fields. All else is optional.

KeePassX gives some indication if the passwords you are using are 'strong' or 'weak'. As may be obvious, you should try to use stronger passwords. For advice on this see the chapter on creating good passwords. Press 'OK' when you are done and you will see something like this:

To recover the passwords (see them) you must double click on the enter and you will see the same window you used for recording the information. If you click on the 'eye' icon to the right of the passwords they will be converted from stars (***) to the plain text so you can read it.

Now you can use KeePassX to store your passwords. However before getting too excited you must do one last thing. When you close KeePassX (choose File->Quit) it

[Untitled Entry]

New Entry

Group: eMail Icon:

Title:

Username:

URL:

Password:

Repeat: Gen.

Quality: 0 Bit

Comment:

Expires: Never

Attachment:

Tools Cancel OK

Figure 9.20:Password Manager

The image shows a 'New Entry' dialog box from a password manager application. The window title is 'my email'. The dialog has a blue header with a key icon and the text 'New Entry'. The main area contains several fields and controls:

- Group:** A dropdown menu set to 'eMail'.
- Icon:** A button with an envelope icon.
- Title:** A text field containing 'my email'.
- Username:** A text field containing 'adam'.
- URL:** An empty text field.
- Password:** A text field with masked characters '*****' and a visibility toggle icon (an eye).
- Repeat:** A text field with masked characters '*****' and a 'Gen.' button.
- Quality:** A progress bar showing approximately 25% completion, with the text '56 Bit' to its right.
- Comment:** A large empty text area.
- Expires:** A date/time field showing '1/1/00 12:00 AM', a dropdown arrow, a clock icon, and a checked checkbox labeled 'Never'.
- Attachment:** An empty text field with three icons: a folder, a document, and a delete icon.

At the bottom, there is a 'Tools' dropdown menu, a 'Cancel' button, and an 'OK' button.

Figure 9.21:Password Manager

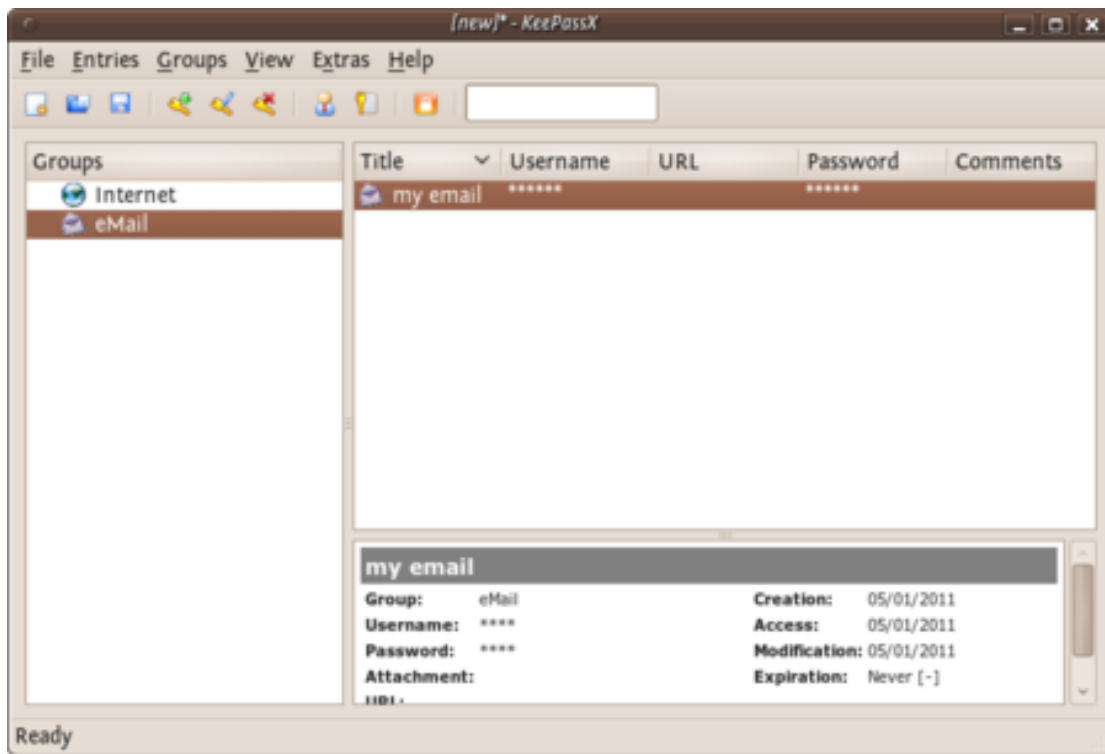


Figure 9.22:Password Manager

asks you if you would like to save the changes you have made.

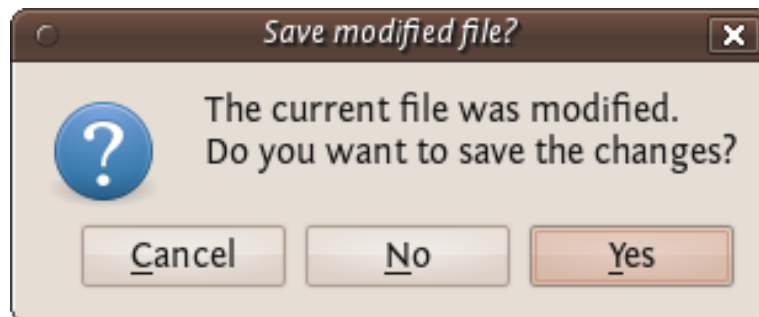


Figure 9.23:Password Manager

Press 'Yes'. If it is the first time you used KeePassX (or you have just created a new database) you must choose a place to store your passwords. Otherwise it will save the updated information in the file you have previously created.

When you want to access the passwords you must then open KeePassX and you will be asked for the master key. After typing this in you can add all your passwords to the database and see all your entries. It is not a good idea to open KeePassX and have it open permanently as then anyone could see your passwords if they can access your computer. Instead get into the practice of just opening it when you need it and then closing it again.

9.3.2 Encrypting Passwords with KeePass on Windows

After you installed KeePass on Windows you can find it in the application menu. Launch the application and the following window should appear.

You start by making a database, the file which will contain your key. From the menu select File > New. You have to choose the name and the location of the file in the dialog window below. In this example we call our database my_password_database.

The next screen will ask you for the master password. Enter the password and click on 'OK'. You will not need to select anything else.

The next window allows you to configure your new database. We do not need to edit anything. Just click on 'OK'.

Now the main window appears again and we see some default password categories on the left side. Lets add a new password in the category 'Internet'. First click on the word 'Internet', then click on the add entry icon under the menu bar.

A window will appear like below. Use the fields to give a description of this particular password, and of course, enter the password itself. When done, click on 'OK'.

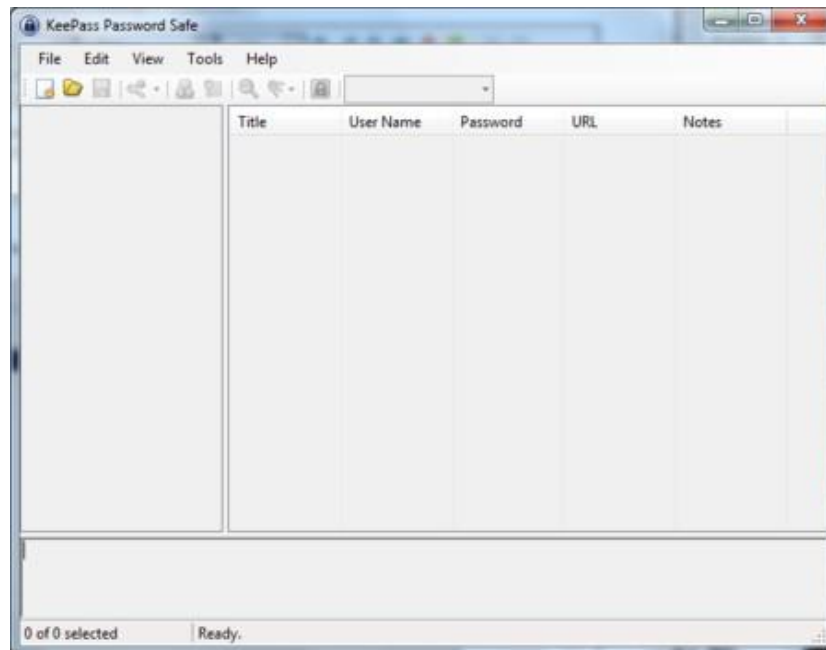


Figure 9.24:Password Manager

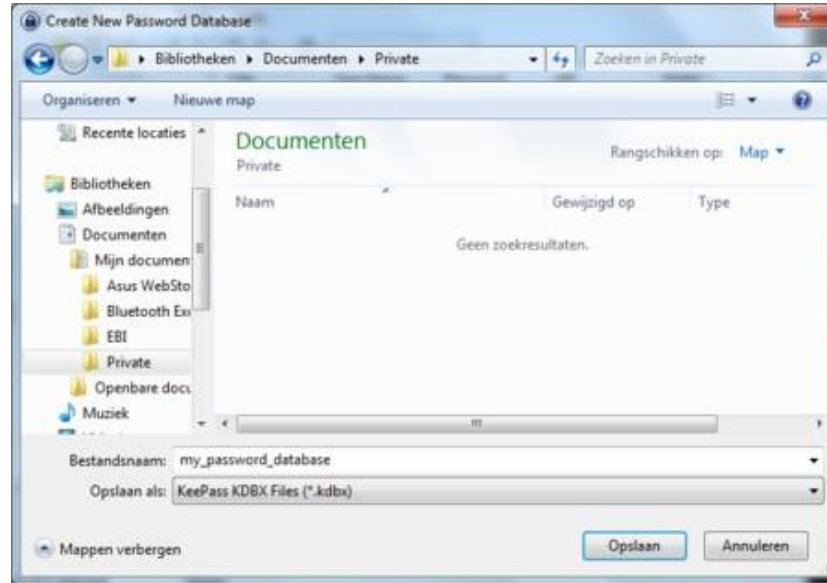


Figure 9.25: Password Manager



Figure 9.26: Password Manager

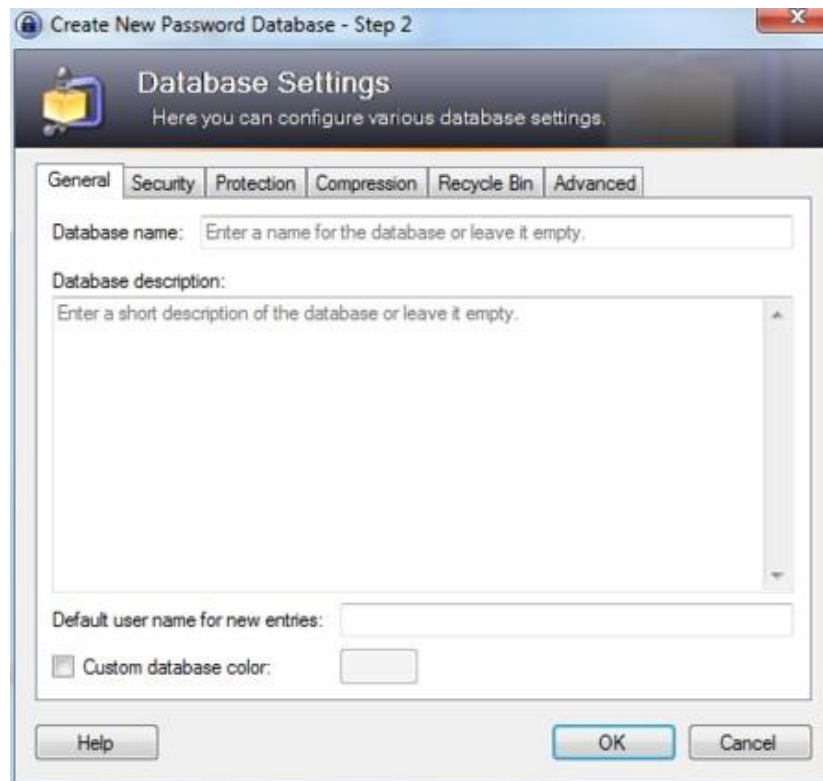


Figure 9.27: Password Manager

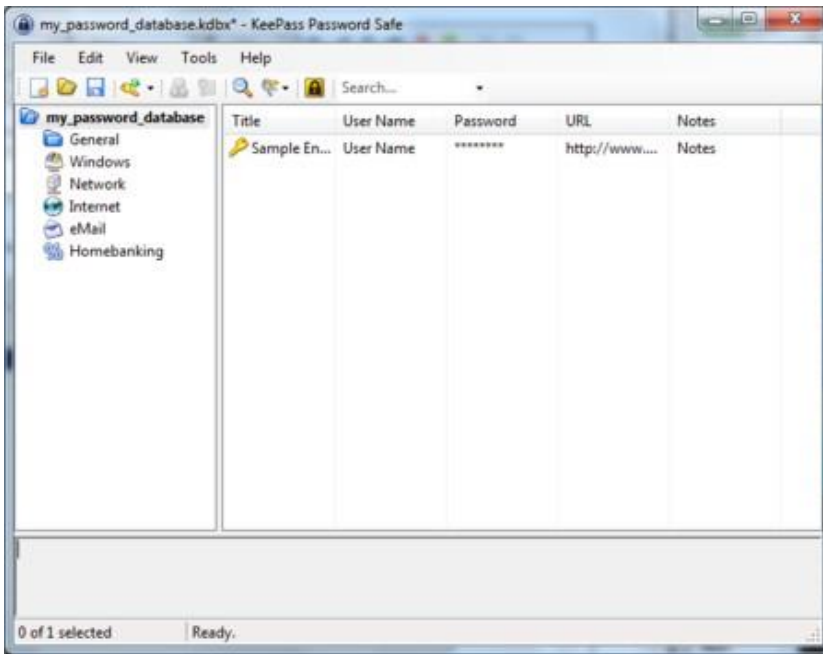


Figure 9.28:Password Manager

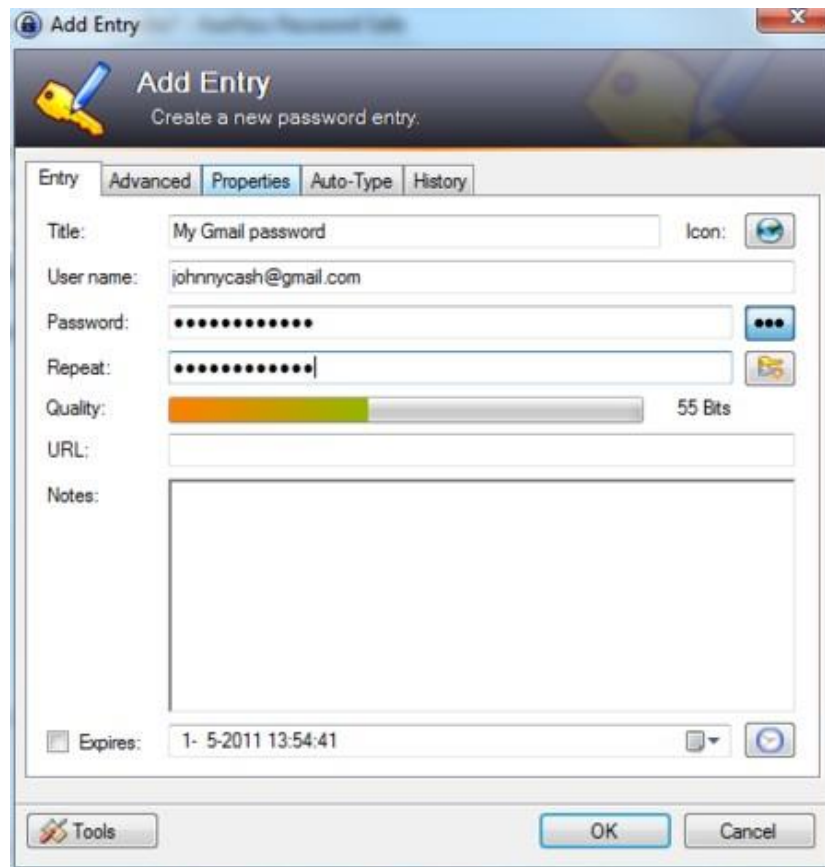


Figure 9.29:Password Manager

9.3.3 Encrypting Passwords with Keychain on Mac OSX

Mac OSX comes pre-installed with the built in password manager ‘Keychain’. Because of its tight integration with the OS most of the time you will hardly know it exists. But every now and then you will have a pop-up window in almost any application asking ‘do you want to store this password in your keychain?’. This happens when you add new email accounts to your mail client, login to a protected wireless network, enter your details in your chat client etc. etc. etc.

Basically what happens is that Mac OSX offers you to store all that login data and different passwords in an encrypted file which it unlocks as soon as you login to your account. You can then check your mail, logon to your WiFi and use your chat client without having to enter your login data all the time over and over again. This is a fully automated process, but if you want to see what is stored where and alter passwords, or lookup a password you will have to open the Keychain program.

You can find the Keychain program in the Utilities folder which lives in the Applications folder.



Figure 9.30: Password Manager

When you open it you will see that your 'Login' keychain is unlocked and see all the items contained in it on the right bottom side of the window.

(note: the window here is empty because it seemed to be deceiving the purpose of this manual to make a screenshot of my personal keychain items and share it here with you)

You can double click any of the items in the Keychain to view it's details and tick 'Show password:' to see the password associated with the item.

You will note that it will ask you for your master or login password to view the item.

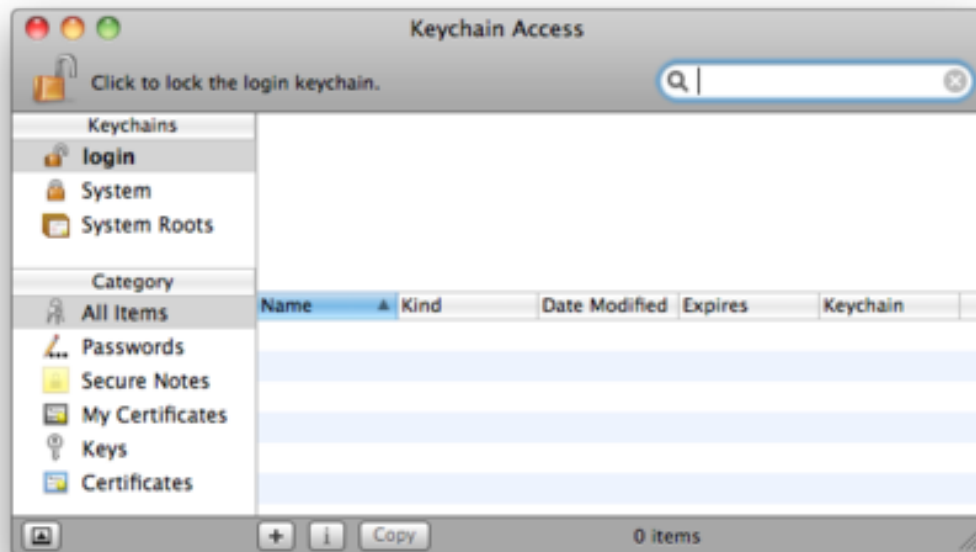


Figure 9.31:Password Manager

You can access modify any of the items and also use the Keychain to securely save any bits and pieces of text using the notes. To do this click on notes and than choose 'New secure Note item' from the file menu.



Figure 9.32:Password Manager



Figure 9.33:Password Manager

10 Using Vpn

10.1 Getting, setting-up and testing a VPN account

In all the VPN systems, there is one computer set up as a server (in an unrestricted location), to which one or more clients connect. The set up of the server is out of the scope of this manual and the set up of this system is in general covered by your VPN provider. This server is one of the two ends of the encrypted tunnel. It is important that the company running this server can be trusted and is located in an area you trust. So to run a VPN, an account is needed at such a trusted server.

Please keep in mind that an account can often only be used on one device at a time. If you want to use a VPN with both your mobile and laptop concurrently, it is very well possible you need two accounts.

10.1.1 An account from a commercial VPN provider

There are multiple VPN providers out there. Some will give you free trial time, others will begin charging right away at an approximate rate of e5 per month. Look for a VPN provider that

offers OpenVPN accounts - it is an Open Source, trusted solution available for Linux, OS X, and Windows, as well as Android and iOS.

When choosing a VPN provider you need to consider the following points:

- Information that is required from you to register an account - the less that is needed the better. A truly privacy concerned VPN provider would only ask you for email address (make a temporary one!), username and password. More isn't required unless the provider creates a user database which you probably don't want to be a part of.
- Payment method to be used to pay for your subscription. Cash-transfer is probably the most privacy-prone method, since it does not link your bank account and your VPN network ID. Paypal can also be an acceptable option assuming that you can register and use a temporary account for every payment. Payment via a bank transfer or by a credit card can severely undermine your anonymity on and beyond the VPN.
- Avoid VPN providers that require you to install their own proprietary client software. There is a perfect open source solution for any platform, and having to run a "special" client is a clear sign of a phony service.
- Avoid using PPTP based VPNs, as several security vulnerabilities exist in that protocol. In fact, if two providers are otherwise equal, choose the one *not* offering PPTP if feasible.
 - Look for a VPN provider that's using OpenVPN - an open source, multi-platform VPN solution.
 - Exit gateways in countries of your interest. Having a choice of several countries allows you to change your geo-political context and appears to come from a different part of the world. You need to be aware of legislation details and privacy laws in that particular country.
 - Anonymity policy regarding your traffic - a safe VPN provider will have a non-disclosure policy. Personal information, such as username and times of connection, should not be logged either.
 - Allowed protocols to use within VPN and protocols that are routed to the Internet. You probably want most of the protocols to be available
 - Price vs. quality of the service and its reliability.
 - Any known issues in regard to anonymity of the users the VPN provider might have had in the past. Look online, read forums and ask around. Don't be tempted by unknown, new, cheap or dodgy offers.

There are several VPN review oriented places online that can help you make the right choice:

- <http://www.bestvpnservice.com/vpn-providers.php>
- <http://vpncreative.com/complete-list-of-vpn-providers>
 - <http://en.cship.org/wiki/VPN>

Setting up your VPN client

"OpenVPN [...] is a full featured SSL VPN software solution that integrates OpenVPN server capabilities, enterprise management capabilities, simplified OpenVPN Connect UI, and OpenVPN Client software packages that accommodate GNU/Linux, OSX, Windows and environments. OpenVPN Access Server supports a wide range of configurations, including secure and granular remote access to internal network and/or private cloud network resources and

applications with fine-grained access control.” (<http://openvpn.net/index.php/access-server/overview.html>)

There is a number of different standards for setting up VPNs, including PPTP, L2TP/IPSec and **OpenVPN**. They vary in complexity, the level of security they provide, and which operating systems they are available for. Do not use PPTP as it has several security vulnerabilities. In this text we will concentrate on OpenVPN. It works on most versions of GNU/Linux, OSX, Windows. OpenVPN is TLS/SSL-based - it uses the same type of **encryption** that is used in HTTPS (Secure HTTP) and a myriad of other encrypted protocols. OpenVPN encryption is based on **RSA** key exchange algorithm. For this to work and in order to communicate, both the server and the client need to have public and private RSA keys.

Once you obtain access to your VPN account the server generates those keys and you simply need to download those from the website of your VPN provider or have them sent

10.1 Getting, setting-up and testing a VPN account

to your email address. Together with your keys you will receive a *root certificate (*.ca)* and a *main configuration file (*.conf or *.ovpn)*. In most cases only the following files will be needed to configure and run an OpenVPN client:

- **client.conf** (or client.ovpn) - configuration file that includes all necessary parameters and settings. NOTE: in some cases certificates and keys can come embedded inside the main configuration file. In such a case the below mentioned files are not necessary.
- **ca.crt** (unless in configuration file) - root authority certificate of your VPN server, used to sign and check other keys issued by the provider.
- **client.crt** (unless in configuration file) - your client certificate, allows you to communicate with VPN server.

Based on a particular configuration, your VPN provider might require a username and password to authenticate your connection. Often, for convenience, these can be saved into a separate file or added to the main configuration file. In other cases, key-based authentication is used, and the key is stored in a separate file:

- **client.key** (unless in configuration file) - client authentication key, used to authenticate to the VPN server and establish an encrypted data channel.

In most cases, unless otherwise necessary, you don't need to change anything in the configuration file and (surely!) **do not edit key or certificate files!** All VPN providers have thorough instructions regarding the setup. Read and follow those guidelines to make sure your VPN client is configured correctly.

NOTE: Usually it's only allowed to use one key per one connection, so you probably shouldn't be using the same keys on different devices at the same time. Get a new set of keys for each device you plan to use with a VPN, or attempt to set up a local VPN gateway (advanced, not covered here).

Download your OpenVPN configuration and key files copy them to a safe place and proceed to the following chapter.

10.1.2 Setting up OpenVPN client

In the following chapters some examples are given for setting up OpenVPN client software. On any flavor of GNU/Linux use your favorite package manager and install **open-vpn** or **openvpn-client** package.

If you want to use OpenVPN on Windows or OSX, have look at:

- <http://openvpn.se> (Windows interface)
- <http://code.google.com/p/tunnelblick> (OSX interface)

10.1.3 Caveats & Gotchas

Although a VPN will obfuscate your IP address, due to the nature of most VPNs your TCP/IP stack meta-data and other identifying information will be sent across the wire as-is.

This may seem trivial, but consider, a standard IP header is 20 bytes in size, some of this is covered by required obvious information, (4 bytes for source IP, 4 bytes for destination IP), etc but some of this header may be other arbitrary options, the TCP header is at least 20 bytes also, with the potential for another 20 bytes of options. The specific configuration of these options varies between operating systems, and even versions of operating system, as such a single TCP SYN packet is often enough to identify a users operating system, version and other potentially revealing information, like the systems uptime. There are [readily available tools](#) which you can use to fingerprint this information, as a test, try connecting to a server running this tool with your normal internet connection, then connecting again over your VPN. You will most likely find that the fingerprints are an identical match both with and without the VPN, and that if your friend were to connect their fingerprint would be different.

As such, it is important to remember some facts: * No one will go to jail for you, if your VPN provider is served a legal request for information about you, they will provide it. Just because they claim they don't log, does not mean they do not have logs. * VPNs provide privacy, they do not provide anonymity, regardless of the advertising and marketing materials provided.

10.2 VPN on Ubuntu

If you use Ubuntu as your operating system, you can connect to a VPN by using the built-in *NetworkManager*. This application is able to set up networks with OpenVPN. PPTP should not be used for security reasons. Unfortunately at the time of writing a L2TP interface is not available in Ubuntu. (It can be done manually, but it goes beyond the scope of this document).

The following example will explain how to connect with an OpenVPN-server. Under all situations we assume you already have a VPN account as described earlier in this section.

10.2.1 Preparing Network Manager for VPN networks

For Ubuntu there is an excellent network utility: Network Manager. This is the same utility you use to set up your Wireless (or wired) network and is normally in the upper right corner of your screen (next to the clock). This tool is also capable of managing your VPNs, but before it can do so, it's necessary to install some extensions.

Installing OpenVPN extension for Network Manager

To install the plugins for Network Manager we will use the Ubuntu Software Center.

1. Open the Ubuntu Software Center by typing software in the Unity search bar



Figure 10.1:VPN on Ubuntu

2. The Ubuntu Software Center enables you to search, install and remove software on your computer. Click on the search box at the top right of the window.
3. In the search box, type in “network-manager-openvpn-gnome” (which is the extension that will enable OpenVPN). It’s necessary to type the full names because the packages are classified as “technical” and don’t pop-up earlier. These packages include all the files you need to establish a VPN connection successfully.
4. Ubuntu may ask you for additional permissions to install the program. If that is the case, type in your password and click Authenticate. Once the package is installed, you can close the Software Center window.
5. To check if the extensions are correctly installed, click on the NetworkManager (the icon at the left of your system clock) and select VPN Connections > Configure VPN.
6. Click Add under the VPN tab.
7. If you see a pop-up asking for the type of VPN and the tunnel technology (Open-VPN) option is available, this means that you have installed the VPN extension in Ubuntu correctly. If you have your VPN login information ready, you can continue right away, else you first have to get a VPN account from a VPN-provider. If this is the case, click cancel to close the Network Manager.



Figure 10.2:VPN on Ubuntu

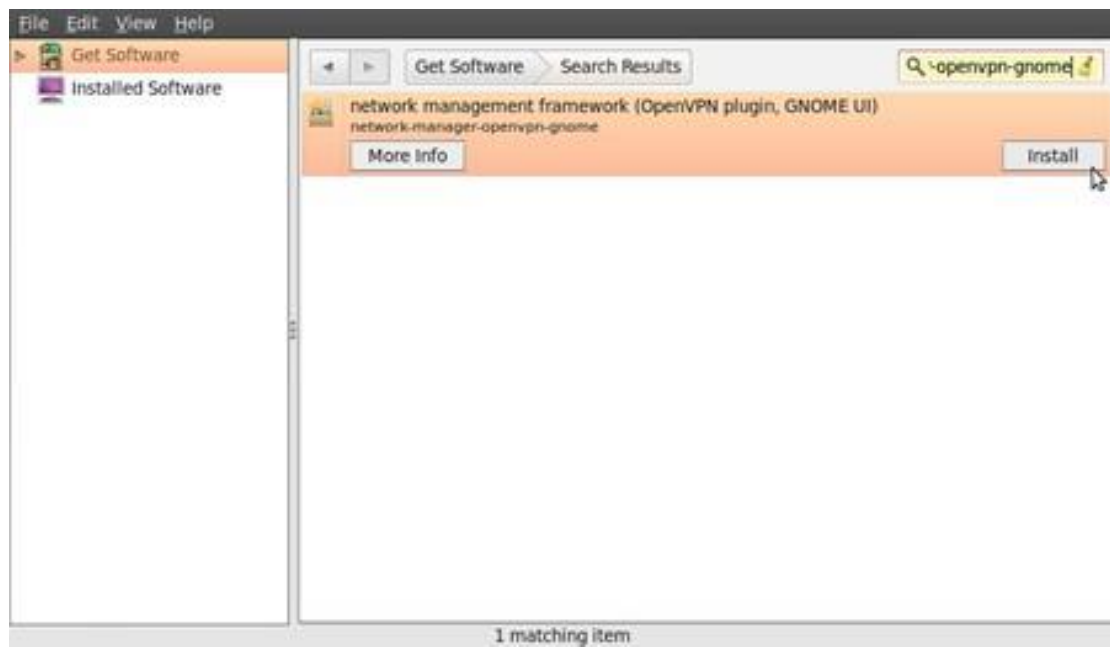


Figure 10.3:VPN on Ubuntu



Figure 10.4:VPN on Ubuntu



Figure 10.5:VPN on Ubuntu

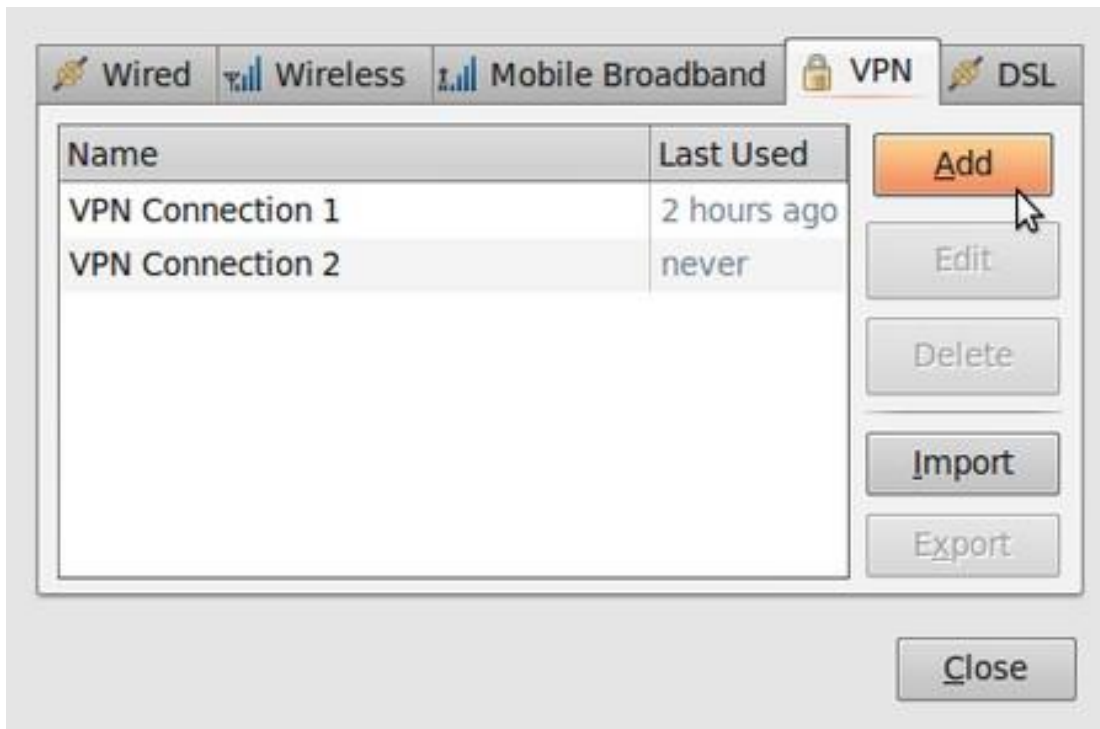


Figure 10.6:VPN on Ubuntu



Figure 10.7:VPN on Ubuntu

10.2.2 Configuring an OpenVPN network

Let us assume you have received your configuration files and credentials from your VPN provider. This information should contain the following

- an *.ovpn file, ex. air.ovpn
- The file: ca.crt (this file is specific for every OpenVPN provider)
- The file: user.crt (this file is your personal certificate, used for encryption of data)
- The file: user.key (this file contains your private key. It should be protected in a good manner. Losing this file will make your connection insecure)

In most cases your provider will send these files to you in a zip file. Some openvpn providers use username and password authentication which will not be covered.

1. Unzip the file you have downloaded to a folder on your hard drive (for example “/home/[yourusername]/.vpn”). You should now have four files. The file “air.ovpn” is the configuration file that you need to import into NetworkManager.

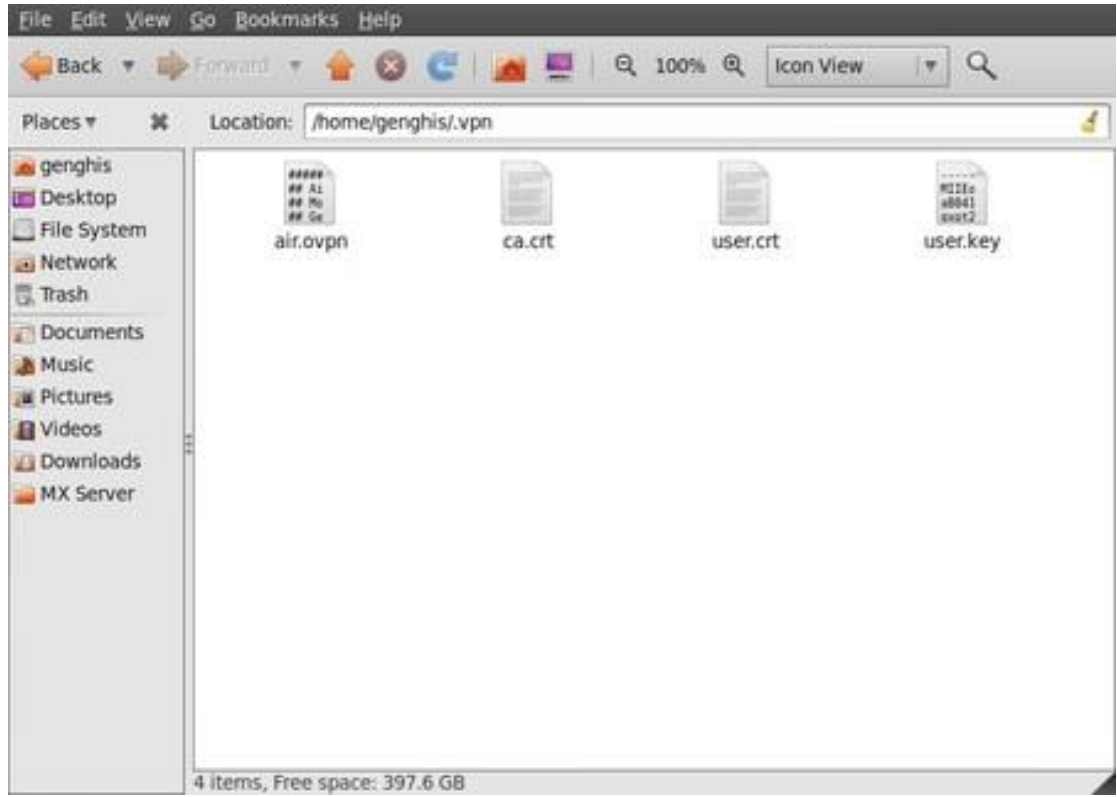


Figure 10.8:VPN on Ubuntu



Figure 10.9:VPN on Ubuntu

- 2.To import the configuration file, open NetworkManager and go to VPN Connections > Configure VPN.
- 3.Under the VPN tab, click Import.



Figure 10.10:VPN on Ubuntu

4. Locate the file `air.ovpn` that you have just unzipped. Click `Open`.
5. A new window will open. Leave everything as it is and click `Apply`.
6. Congratulations! Your VPN connection is ready to be used and should appear on the list of connections under the VPN tab. You can now close `NetworkManager`.

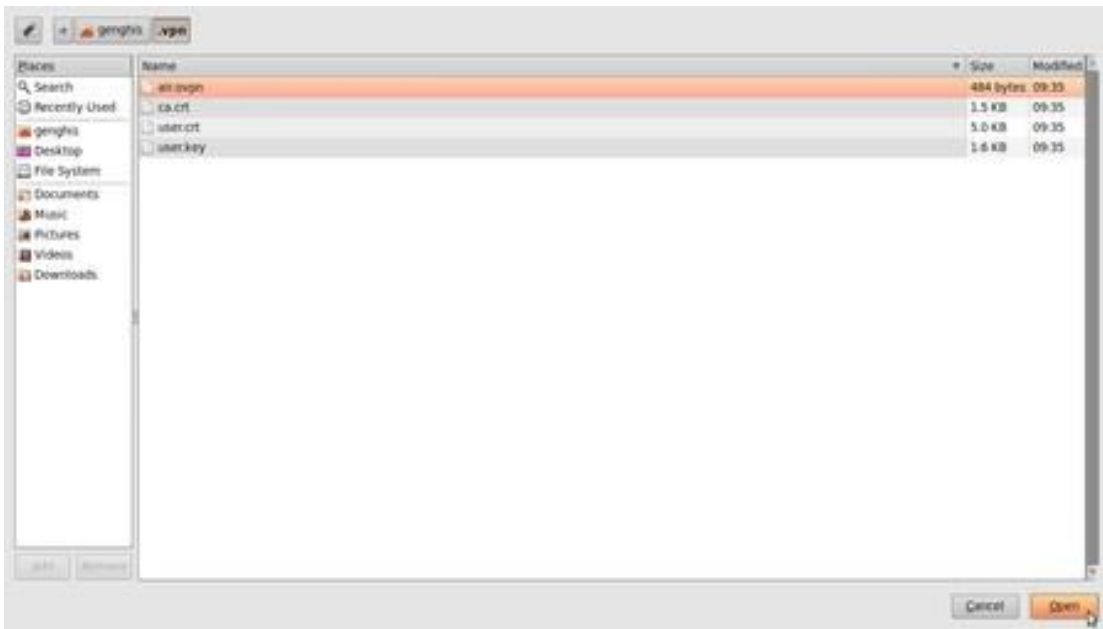


Figure 10.11:VPN on Ubuntu

Connection name:

Connect automatically

VPN **IPv4 Settings**

General

Gateway:

Authentication

Type:

User Certificate:

CA Certificate:

Private Key:

Private Key Password:

Show passwords

Available to all users

Figure 10.12:VPN on Ubuntu

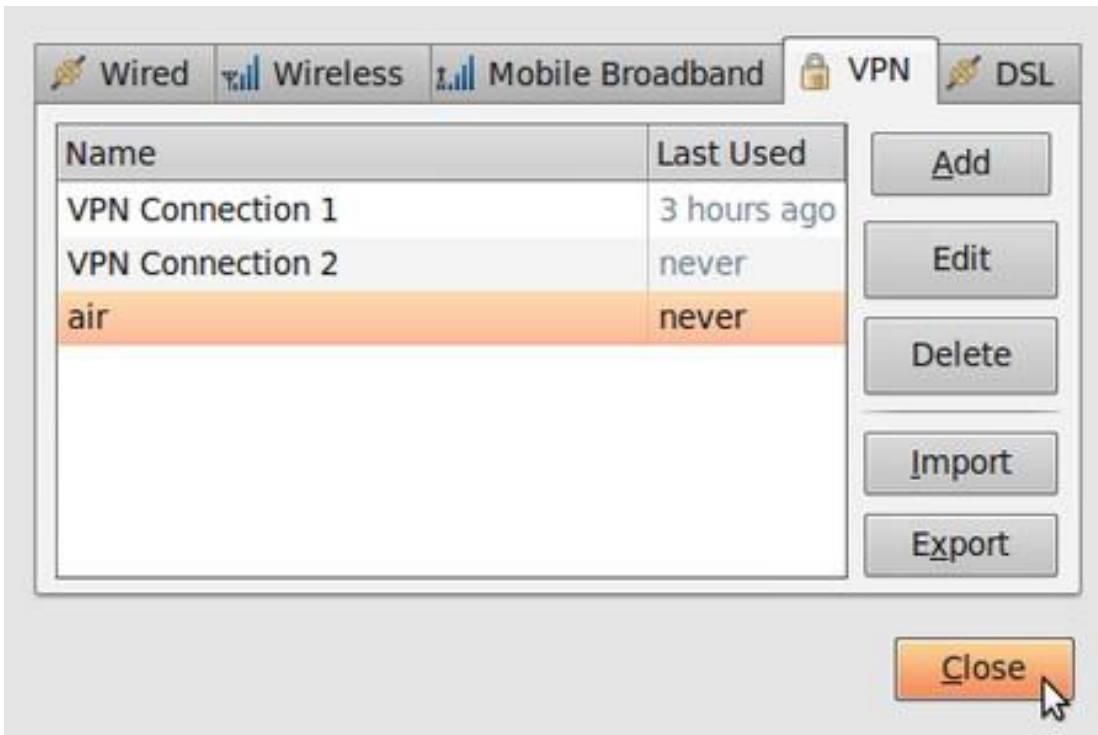


Figure 10.13:VPN on Ubuntu

10.2.3 Using your new VPN connection

Now that you configured NetworkManager to connect to a VPN service using the Open-VPN client, you can use your new VPN connection to circumvent Internet censorship. To get started, follow these steps:

1. In the NetworkManager menu, select your new connection from VPN Connections.



Figure 10.14:VPN on Ubuntu

2. Wait for the VPN connection to be established. When connected, a small padlock should appear right next to your NetworkManager icon, indicating that you are now using a secure connection. Move your cursor over the icon to confirm that the VPN connection is active.
3. Test your connection, using the method described in the “Make sure it works” section of this chapter.
4. To disconnect from your VPN, select VPN Connections > Disconnect VPN in the NetworkManager menu. You are now using your normal connection again.

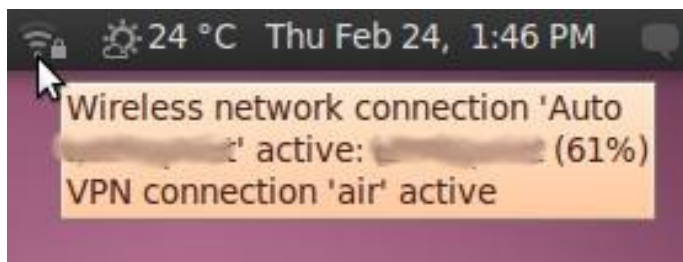


Figure 10.15:VPN on Ubuntu



Figure 10.16:VPN on Ubuntu

10.3 VPN on MacOSX

Setting up a VPN on MacOSX is very easy once you have your account details ready, Let's assume have your credentials from your VPN provider for L2TP/IPSec connection ready. This information should contain the following:

- Username, ex. bill2
- Password, ex. verysecretpassword
- VPN server, ex. tunnel.greenhost.nl
- A Pre-Shared-Key or Machine-certificate

10.3.1 Setup

1. Before getting started, please be sure you've read the paragraph "testing before and after account set up", this way you will be able to validate if your connection is actually working after set up.
2. A VPN is configured in the network settings, that are accessible via "System Preferences.." in the Apple menu.

3. Next, open the Network preferences.
4. OSX uses this nifty system to lock windows. To add a VPN it is necessary to unlock the screen: you can do this by clicking on the lock on the left bottom of the screen.
5. Enter our user credentials
6. Now we can add a new network. Do this by clicking on the “+” sign
7. In the pop-up you need to specify the type of connection. In this case choose an VPN interface with L2TP over IPSec. This is the most common system. Also don't forget to give the connection a nice name.
8. Next comes the connection data. Please fill in the provided server name and user name (called ‘Account Name’). If this is done, click on the “Authentication Settings. . .” button

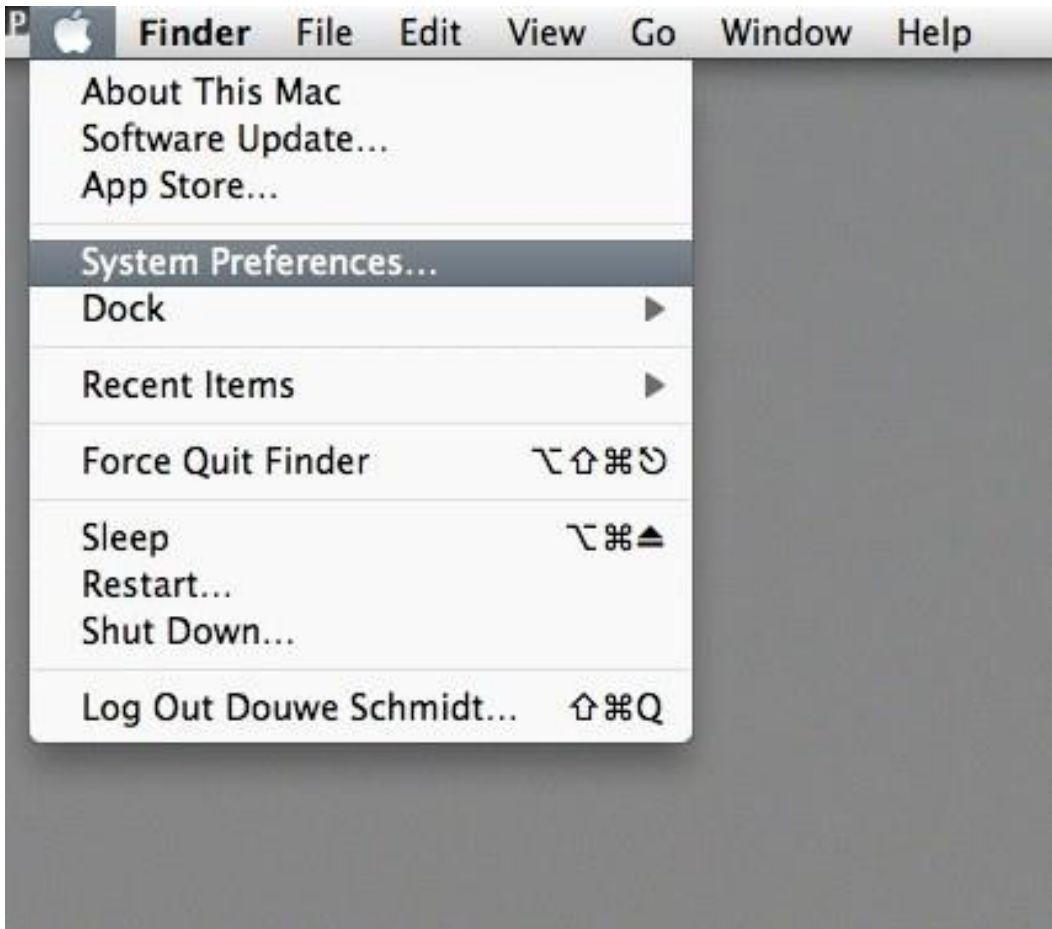


Figure 10.17:VPN on Mac OS X



Figure 10.18:VPN on Mac OS X



Figure 10.19:VPN on Mac OS X



Figure 10.20:VPN on Mac OS X

9. In the new pop-up you can specify connection specific information. This is the way the user is authenticated and how the machine is authenticated. The user is very commonly authenticated by using a password, although other methods are possible. Machine authentication is often done by a Shared Secret (Pre-Shared-Key/PSK), but also quite often by using a certificate. In this case we use the Shared Secret method. When this is done click OK.
10. Now you return back to the network screen. The next step is very important, so click on “Advanced. . .”
11. In the new pop up you will see an option to route all traffic through the VPN connection. We want to enable this, so all our traffic is encrypted.
12. Well, all is done. Now hit the Connect button!
13. A pop-up appears. You need to confirm your changes, just hit “Apply”
14. After a few seconds, on the left side the connection should turn green. If so, you are connected!
15. Ok, now test your connection!



Figure 10.21:VPN on Mac OS X



Figure 10.22:VPN on Mac OS X



Figure 10.23:VPN on Mac OS X

User Authentication:

Password:

RSA SecurID

Certificate

Kerberos

CryptoCard

Machine Authentication:

Shared Secret:

Certificate

Group Name:

(Optional)

Figure 10.24:VPN on Mac OS X



Figure 10.25:VPN on Mac OS X

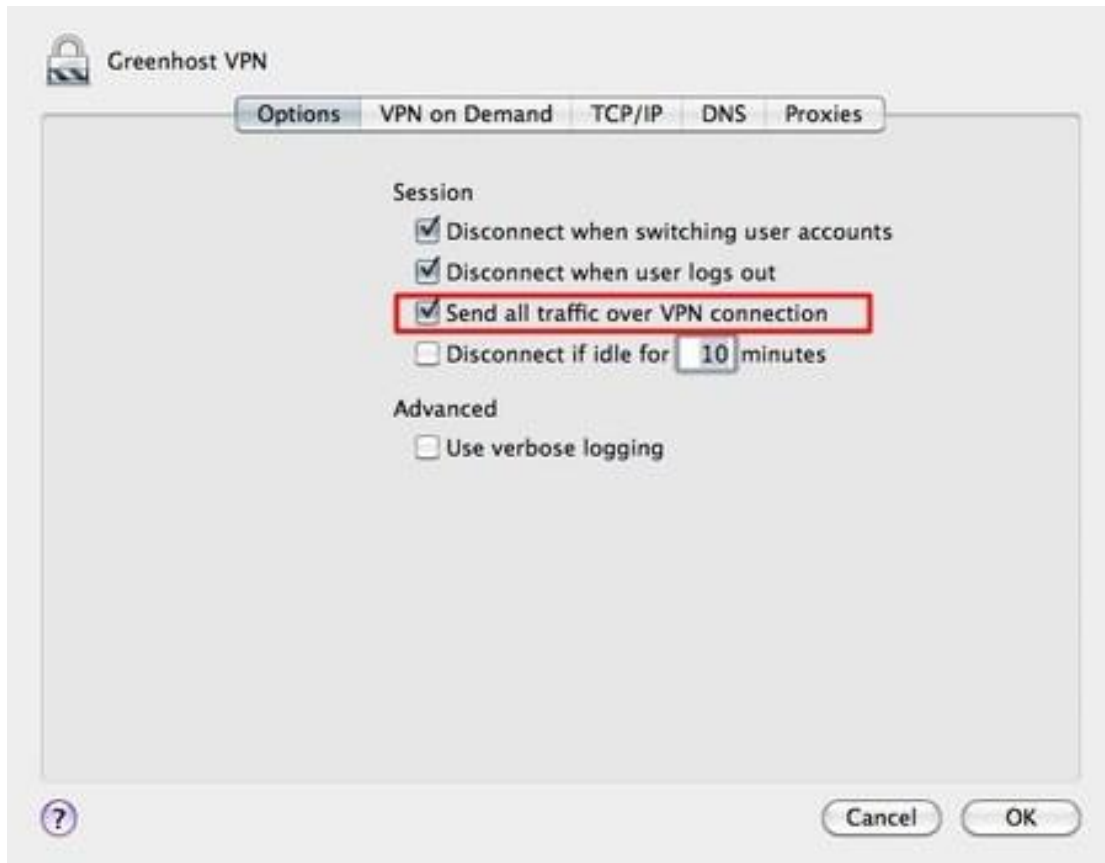


Figure 10.26:VPN on Mac OS X

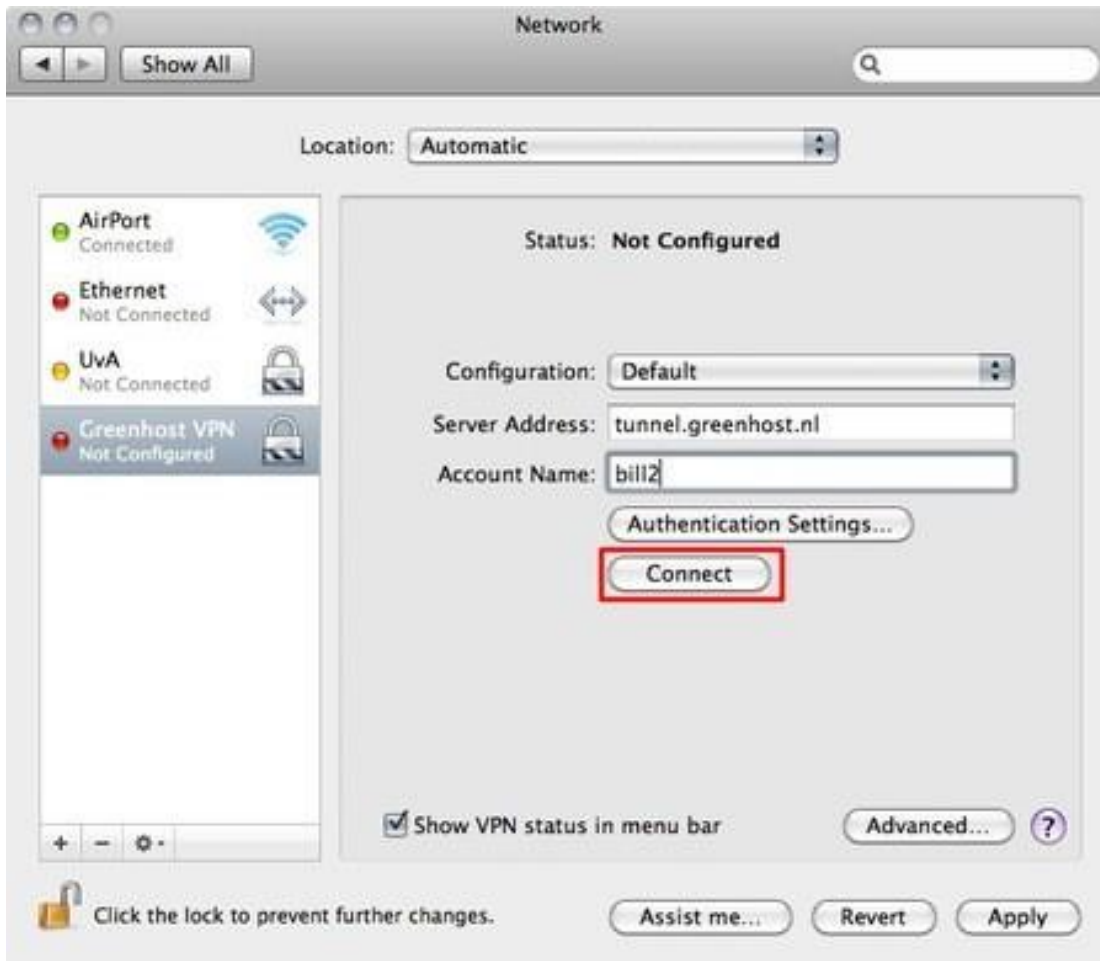


Figure 10.27:VPN on Mac OS X



Figure 10.28:VPN on Mac OS X



Figure 10.29:VPN on Mac OS X

10.4 VPN on Windows

Setting up a VPN on Windows is very easy once you have your account details ready. Let's assume have your credentials from your VPN provider for L2TP/IPSec connection ready. This information should contain the following:

- Username, ex. bill2
- Password, ex. verysecretpassword
- VPN server, ex. tunnel.greenhost.nl
- A Pre-Shared-Key or Machine-certificate

10.4.1 Setup

1. Before getting started, please be sure you've read the paragraph "testing before and after account set up", this way you will be able to validate if your connection is actually working after set up.
2. We need to go to the "Network and Sharing Center" of Windows to create a new VPN connection. We can access this center easily by clicking on the network icon next to the

systemclock en click on “open Network and Sharing Center”

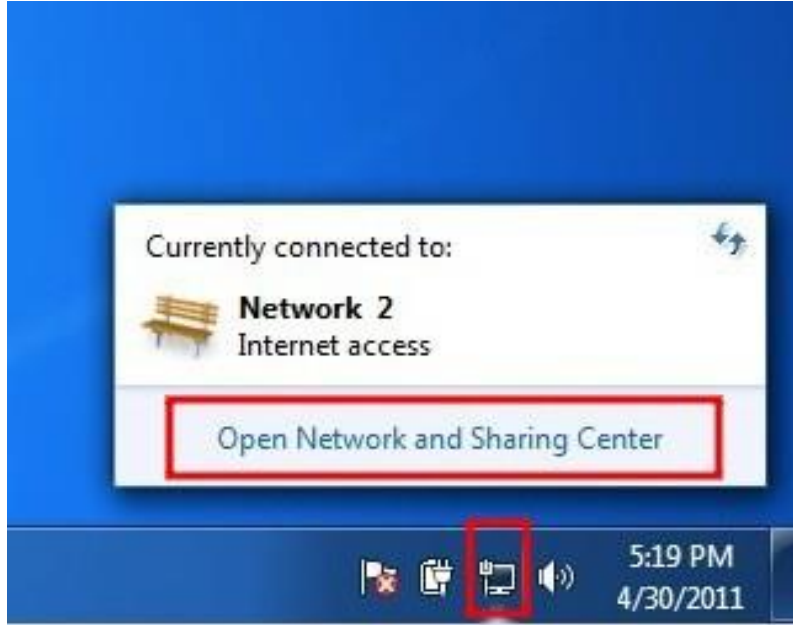


Figure 10.30:VPN on Windows

3.The “Network and Sharing Center” will popup. You will see some information about your current network. Click on “Connect to a network” to add a VPN connection.

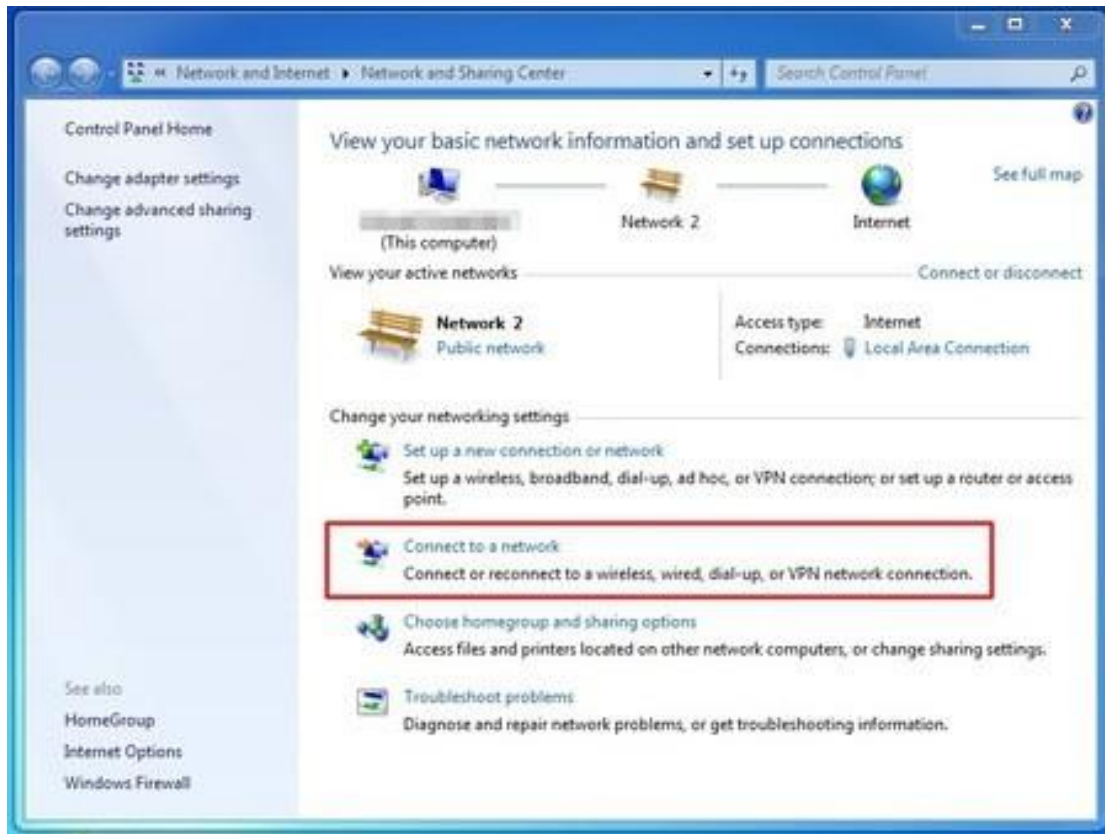


Figure 10.31:VPN on Windows

4. The wizard to setup a connection will popup. Choose the option to “connect to a workplace”, which is Microsoft’s way of naming a VPN connection.

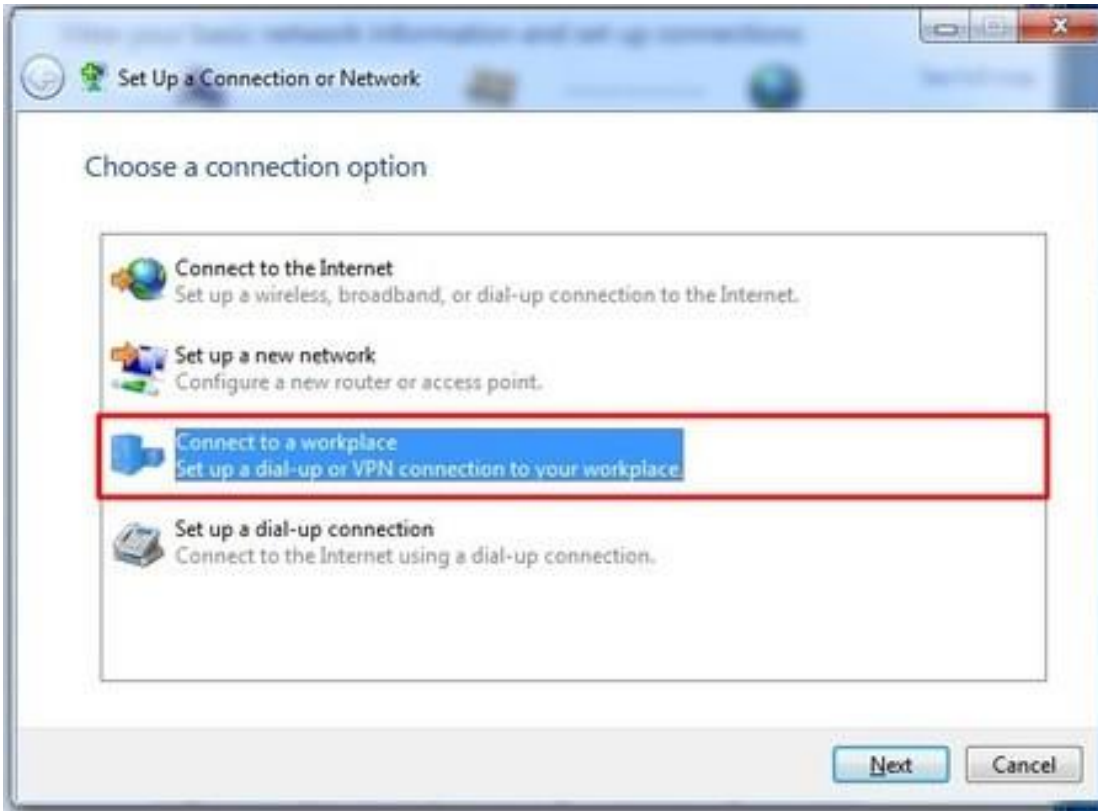


Figure 10.32:VPN on Windows

5. The next screen asks us if we want to use our Internet connection or an old-school phone line to connect to the VPN. Just choose the first option then.
6. The next screen asks for the connection details. Enter here the server of your VPN-provider (called “Internet address” in this dialog). On the bottom please check the box “Don’t connect now; just set it up”. Using this option the connection will be automatically saved and it’s easier to control extra settings. If this is all done, hit the “next” button
7. Next up are your username and password. Just give them like you received them from your VPN-provider. If the connection fails, Windows forgets them. So keep them with you, you maybe need them later. If this is done. Click “create”.



Figure 10.33:VPN on Windows

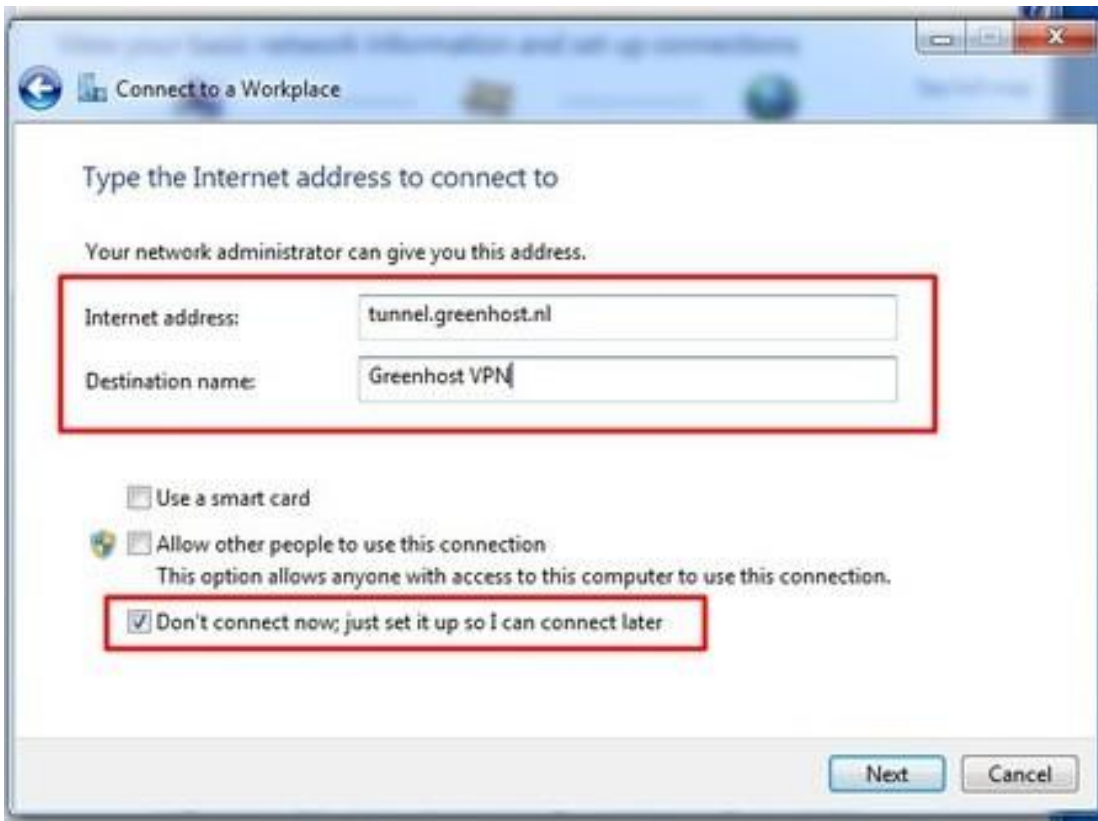


Figure 10.34:VPN on Windows



Figure 10.35:VPN on Windows

8. Your connection is now available, if you click the the network icon again, you will see a new option in the network menu, the name of your VPN connection, just click it to connect.



Figure 10.36:VPN on Windows

9.And click “connect”

10.A VPN connection dialog appears. This give us the opportunity to review our settings and to connect. You can try to connect, Windows will try to discover all



Figure 10.37:VPN on Windows

other settings automatically. Unfortunately, this does not always work, so if this is not working for you, hit the “properties” button.



Figure 10.38:VPN on Windows

11. The properties windows appear. The most important page is the “Security” page, click on the Security tab to open it.
12. In the security tab you can specify VPN type, normally L2TP/IPSec. Do not use PPTP as it has several security vulnerabilities. For L2TP/IPSec also have a look at the Advanced settings.

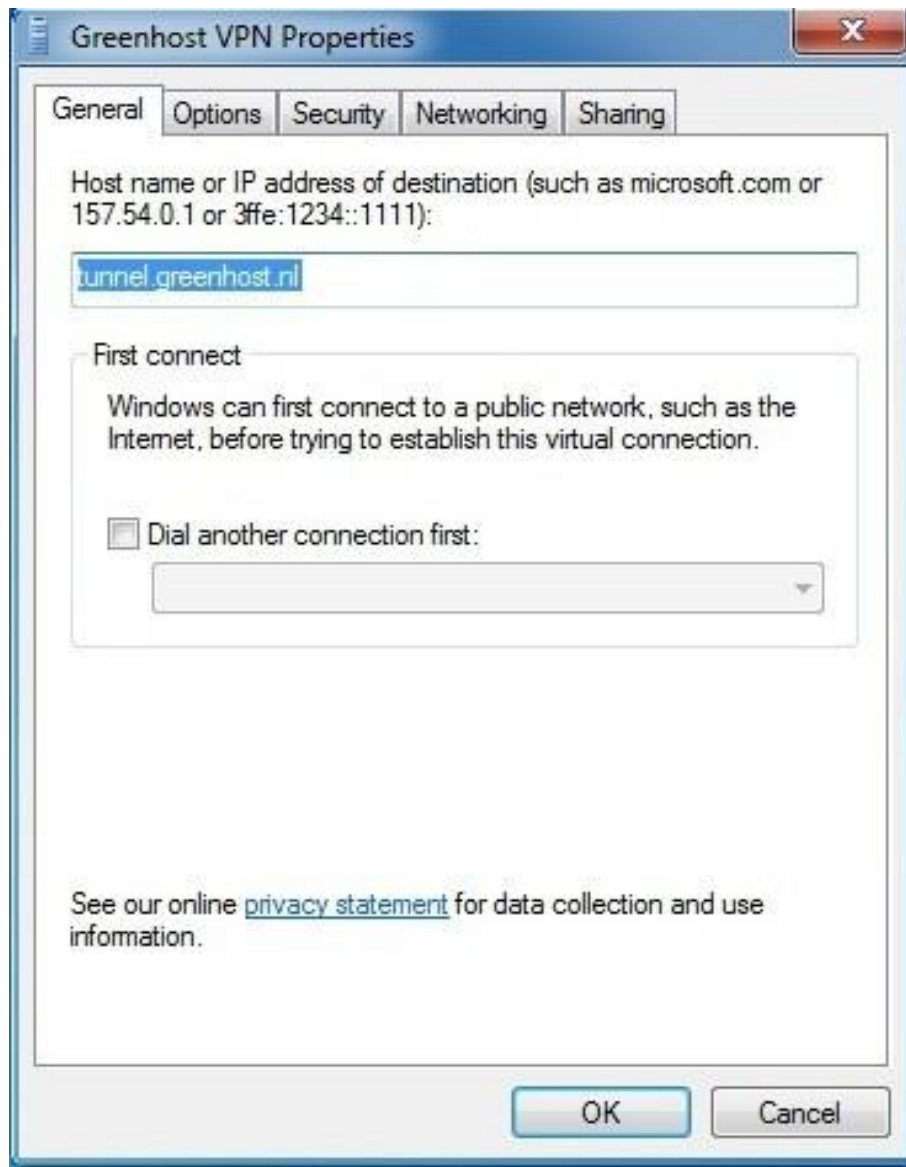


Figure 10.39:VPN on Windows



Figure 10.40:VPN on Windows

13. In the Advanced Settings window, you can specify if you are using a pre-shared key or a certificate. This depends on your VPN-provider. If you have received a pre-shared-key, Select this option and fill in this key. Hit ok afterwards. You will return to the previous window, click ok there also



Figure 10.41:VPN on Windows

14. Back in to connection window try to connect now. Please be sure your username and password are filled out.

15.A connection popup will appear

16.Online! Don't forget to check if your VPN is working properly.



Figure 10.42:VPN on Windows



Figure 10.43:VPN on Windows

10.5 Making Sure Your VPN Works

Once you're done setting up your VPN, one of the first things you should do is test whether your data is actually being transferred through your VPN network. The simplest way to test this is to check your public IP address, which is the IP address you're exposing to the internet.

There are numerous websites that will tell you what your IP address is, and where that IP address is located (also known as its geolocation). Many search engines will report your IP address if you search for “My IP,” but you can also use dedicated services like <http://www.myip.se> and <http://www.ipchicken.com>.

Check your IP address before connecting to your VPN. Once you connect to your VPN, your computer’s public IP address should change to match that of your VPN server, and your geolocation should change to wherever your VPN server is located.

Once your external IP is the same as the IP of your VPN server, you can rest assured your communication is encrypted.

11 Disk Encryption *** TrueCrypt Compromised***

Do not use TrueCrypt. It is no longer safe, has been compromised, and is no longer being updated. Please replace all instances of TrueCrypt with VeraCrypt. The instructions will work similarly and I have replaced all links to TrueCrypt with VeraCrypt.

11.1 Installing VeraCrypt

VeraCrypt can be installed on Windows, Linux, or Mac OSX. The installation files are available here: <https://veracrypt.codeplex.com/releases/view/616110>

The following three sections give complete details on how to install TrueCrypt for each of these Operating Systems, starting with Ubuntu and Debian.

11.1.1 Installing on Ubuntu/Debian

TrueCrypt is not available in the standard Ubuntu repositories. This means you cannot use the Ubuntu Software Center or *apt-get* (a command line method for installing software on Ubuntu) to install it. Instead you must first visit the TrueCrypt downloads page (<https://veracrypt.codeplex.com/releases/view/616110>).

You will see a drop-down menu under the heading Linux.

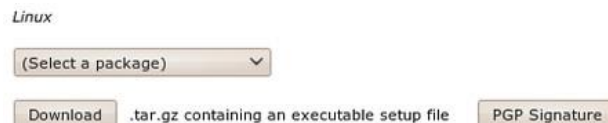


Figure 11.1: Dropdown menu on the download page

From the ‘(Select a package)’ drop down menu you can choose from four options:

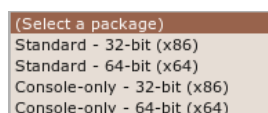


Figure 11.2: Choosing the correct package to download

This is a little technical - the console version is the one you choose if you are either very technical and don't like Graphical User Interfaces or you wish to run this on a machine that you have only a terminal (command line or 'shell') access to (like a remote server for example).

Assuming you are running this in your laptop its best to choose the easy 'standard' option - this will give you a nice user interface to use. From these two options you need to choose the one most suitable for the *architecture* of your machine. Don't know what this means? Well, it basically comes down to the type of hardware (processor) running

on your computer, the options are 32-bit or 64-bit. Unfortunately Ubuntu does not make it easy for you to find this information if you don't already know it. You need to open a 'terminal' from the Applications->Accessories menu and type the following, followed by the [enter] key

```
uname -a
```

The output will be something like Linux bigsy 2.6.32-30-generic #59-Ubuntu SMP Tue Mar 1 21:30:46 UTC 2011 x86_64 GNU/Linux. In this instance you can see the architecture is 64-bit (x86_64). In this example I would choose the 'Standard - 64-bit (x64)' option. If you see i686 somewhere in the output of the uname command then you would choose the other standard option to download.

Once selected press the 'download' button and save the file to somewhere on your computer.

So the installation process is still not over. The file you downloaded is a compressed file (to make downloading it faster) and you need to first de-compress the file before you install it. Fortunately Ubuntu makes this easy - simply browse to the file on your computer and right click on it and choose 'Extract Here'.

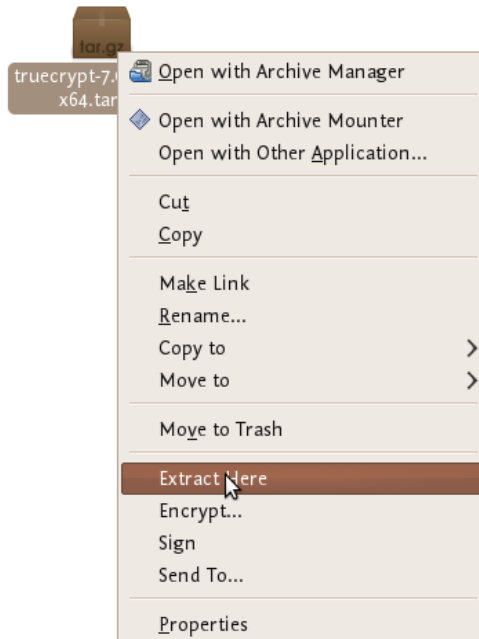


Figure 11.3:Right click and 'extract here'

You will see a new file appear next to the compressed file: Nearly done!
Now right click on the new file and choose 'open': If all is well you will see a window open like this:

Choose 'run' and you see the following:

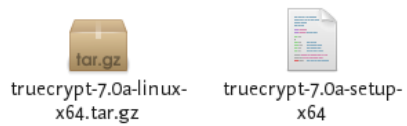


Figure 11.4: The extracted file

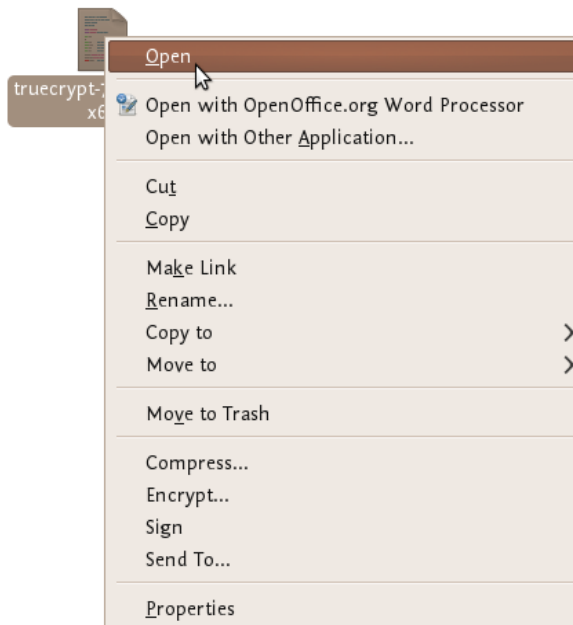


Figure 11.5: Right click and 'open'



Figure 11.6: Window opens to confirm you want to 'run' the file

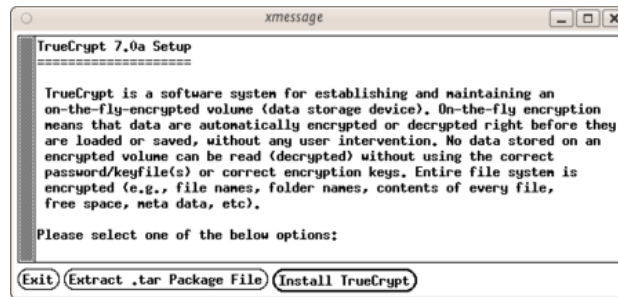


Figure 11.7: Truecrypt installer starts. . .

Now we are getting somewhere. . . press ‘Install TrueCrypt’. You will be displayed a user agreement. At the bottom press ‘I accept and agree to be bound by the license terms’ (sounds serious). You will then be shown another info screen telling you how to uninstall TrueCrypt. Press ‘OK’ then you will be asked for your password to install software on your computer. Enter your password and then you will finally see a screen like this:

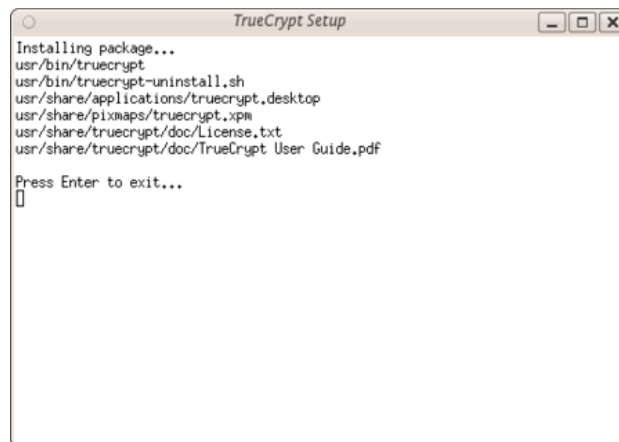


Figure 11.8: Truecrypt install finished. . . ‘press Enter to exit’

Believe it or now your are done. . . VeraCrypt is installed and you can access it from the Applications->accessories menu. . . close the setup window. Now proceed to the chapter on Using TrueCrypt.

11.1.2 Installing on OSX

1. To install TrueCrypt on OSX first visit the download page (<https://veracrypt.codeplex.com/releases/view/616110>) and press the download button under the OSX section.
2. Download this to your computer find the .dmg file and open it to access the installation package.

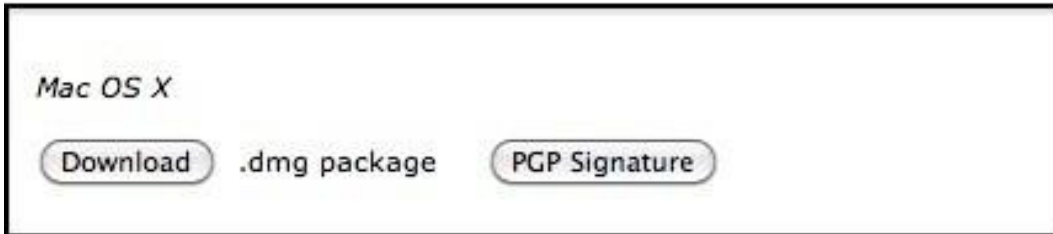


Figure 11.9:Download button



Figure 11.10:Open the .dmg file

3. Open the installation package, and click through the dialogues.
4. Choose the standard installation. You can choose to do a customized installation and deselect FUSE, but why would you? You need it!
5. After the installation finishes you can find the program in your 'Applications' folder.

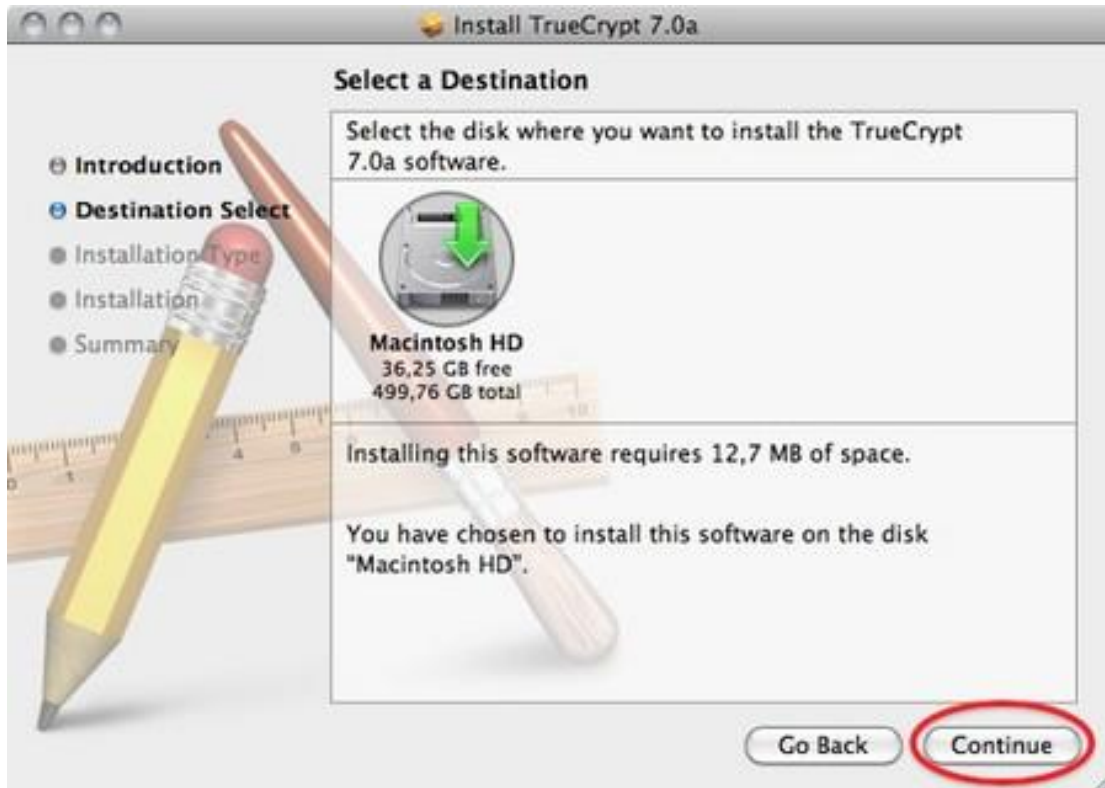


Figure 11.11: Click through the dialogues

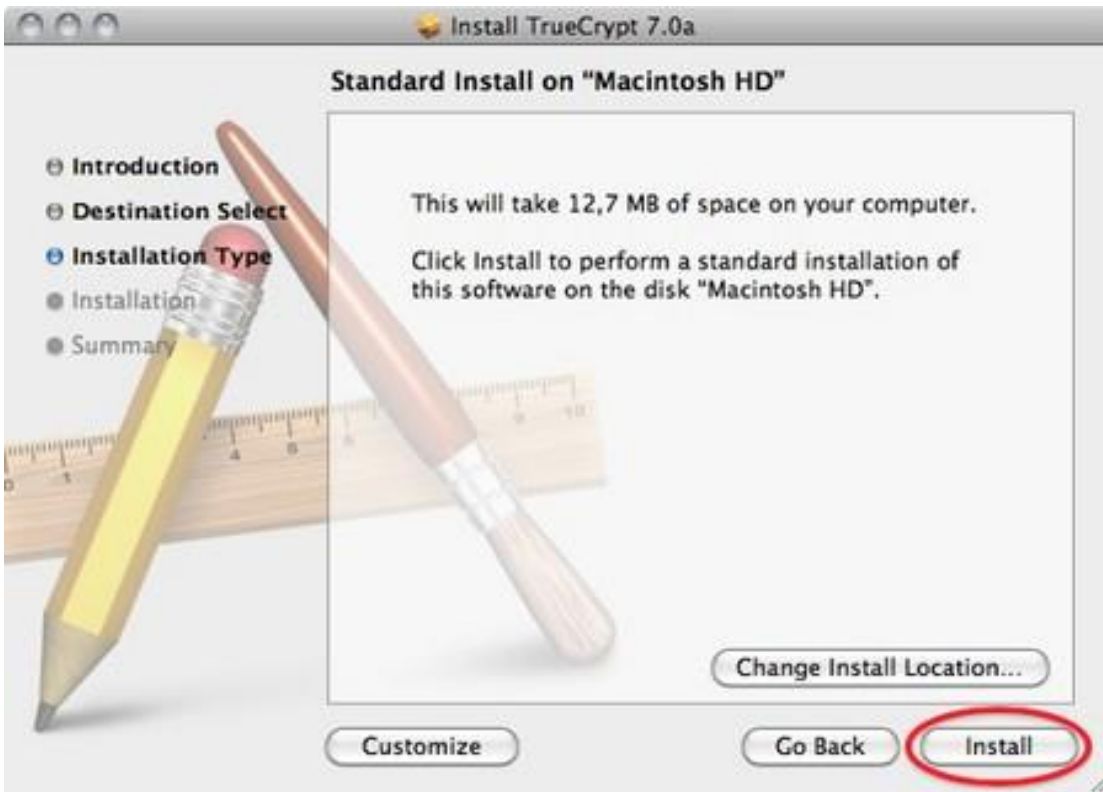


Figure 11.12: Choose standard installation



Figure 11.13: Truecrypt launcher now in Applications

11.1.3 Installing on Windows

To install TrueCrypt on Windows first visit the download page (<https://veracrypt.codeplex.com/releases/view/616110>) and press the download button under the Windows section.

Download this to your computer and then double click on the file. You will see a license agreement.

Click on 'I accept and agree to be bound by the license terms' and then click 'Accept'.

Leave the above screen with the defaults and press 'Next >' and you will be taken to the Setup Options window:

You can leave this with the defaults. If you want to set up TrueCrypt just for yourself then consider not selecting the 'Install for all users'. However if you are installing this on your own machine and no one else uses the computer then this is not necessary. You may also wish to consider installing TrueCrypt in a folder other than the default. In which case click 'Browse' and choose another location. When you are done click 'Install' and the process will proceed:

When the installation is complete you will get a verification popup that it was successful. Close this window and click 'Finish' and all is done. Now proceed to the chapter on Using TrueCrypt.

11.2 Using VeraCrypt

The following are step-by-step instructions on how to create, mount, and use a TrueCrypt volume.

11.2.1 Creating a VeraCrypt Container

1. Install VeraCrypt. Then launch VeraCrypt by

- double-clicking the file VeraCrypt.exe in Windows
- opening Applications->Accessories->VeraCrypt in Ubuntu
- on MacOSX open it by clicking Go > Applications. Find VeraCrypt in the Applications folder and double click on it.

2. When the main VeraCrypt window appears. Click Create Volume.

3. You should see the VeraCrypt Volume Creation Wizard window appear on screen.

Where do you want to create the VeraCrypt volume? You need to choose now. This can be in a file, which is also called a container, in a partition or drive. The following steps will take you through the first option creating a VeraCrypt volume within a file.

You can just click Next, as the option is selected by default,

4. Next you need to choose whether to create a standard or hidden VeraCrypt volume. We will walk you through the former option and create a standard VeraCrypt volume.

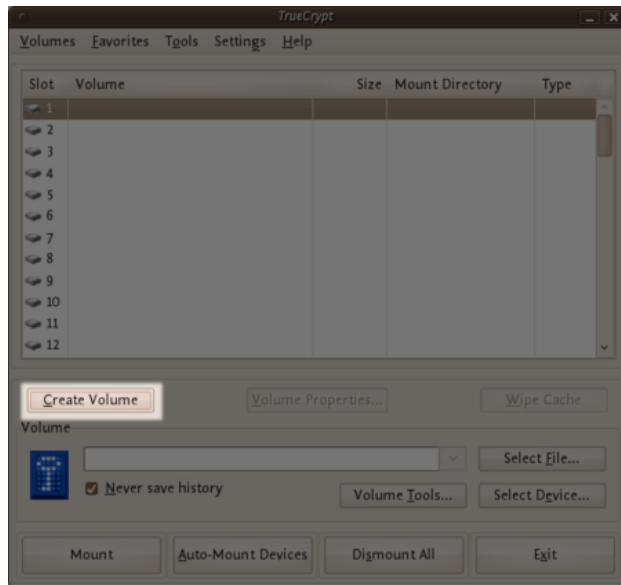


Figure 11.14:Using TrueCrypt



Figure 11.15:Using TrueCrypt



Figure 11.16:Using TrueCrypt

You can just click Next, as the option is selected by default.

5. Now you have to specify where to have the VeraCrypt volume (file container) created.

Note that a VeraCrypt container behaves like any normal file. It can be moved or deleted as any normal file.



Figure 11.17:Using TrueCrypt

Click Select File.

The standard file selector will now appear on screen (the VeraCrypt Volume Creation Wizard remains open in the background). You need to browse to the folder that the file should be created in and then type into the 'name' field the name for the file you wish to create.

We will create our VeraCrypt volume in the folder 'adam/true' and the filename of the volume (container) will be 'myencryptedfile'. You may, of course, choose any other filename and location you like (for example, on a USB stick). Note that the file 'myencryptedfile' does not exist yet - VeraCrypt will create it. Press 'Save' when you are ready. The file selector window should close.

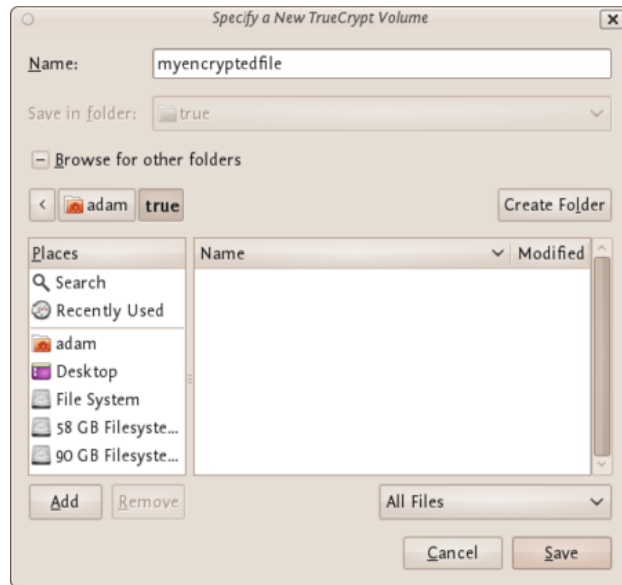


Figure 11.18:Using TrueCrypt

IMPORTANT: Note that VeraCrypt will not encrypt any existing files. If an existing file is selected in this step, it will be overwritten and replaced by the newly created volume (the contents of the existing file will be lost). You will be able to encrypt existing files later on by moving them to the VeraCrypt volume that we are creating now.

6. In the Volume Creation Wizard window (which was previously running in the background), click Next.

7. Here you can choose an encryption algorithm and a hash algorithm for the volume.



Figure 11.19:Using TrueCrypt

The VeraCrypt manual suggests that if you are not sure what to select here, you can use the default settings and click Next (for more information about each setting have a look at the VeraCrypt documentation website).

8. Now choose the size of your container. You should be fine with 1 megabyte but for this example we will enter '20' into the available field.



Figure 11.20: Using TrueCrypt

You may, of course, specify a different size. After you type the desired size in the input field, click Next.

9. This step is really important, choosing a password.

The information displayed in the Wizard window about what is considered a good password, should be read carefully.

Choose a strong password, type it in the first input field. Then re-type it in the input field below the first one.

When you are done click Next.

10. Now you must choose the format of your partition (this step may not be available for you under windows or OSX). If using Ubuntu you can choose a Linux file type or FAT (Windows) for simplicity leave it at the default.

Then press Next.

11. Next VeraCrypt tries to generate random information to help encrypt your container. For 30 seconds move your mouse as randomly as possible within the Volume Creation Wizard window. Move the mouse as much as possible for up to a minute. This significantly increases security by increasing the cryptographic strength of the encryption keys. (security). Move your mouse around until you are bored.



Figure 11.21:Using TrueCrypt

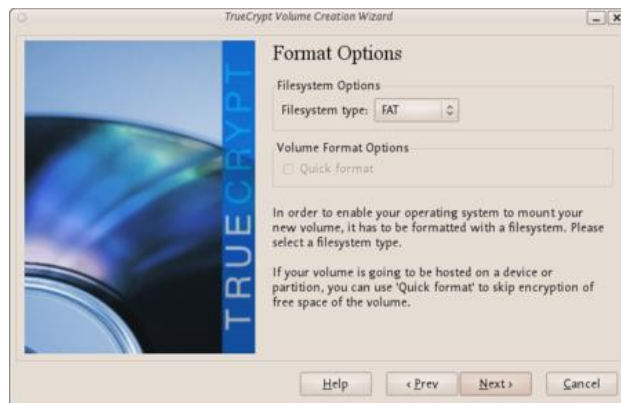


Figure 11.22:Using TrueCrypt



Figure 11.23:Using TrueCrypt

Then Click Format.

VeraCrypt will now create a file in the folder you selected with the name you chose. This file will be a VeraCrypt container, containing the encrypted VeraCrypt volume. This may take some time depending on the size of the volume. When it finishes this should appear:



Figure 11.24:Using TrueCrypt Click

OK to close the dialog box.

12. Well done! You've just successfully created a VeraCrypt volume (file container).

In the VeraCrypt Volume Creation Wizard window, click Exit.

11.2.2 Mounting the Encrypted Volume

1. Open up VeraCrypt again.

2. Make sure one of the 'Slots' is chosen (it doesn't matter which - you can leave at the default first item in the list). Click Select File.

The standard file selector window should appear.

3. In the file selector, browse to the container file (which we created earlier) and select it.

Click Open (in the file selector window). The file selector window should disappear.

4. In the main VeraCrypt window, click Mount.

Password prompt dialog window should appear. 5. Type

the password in the password input field.

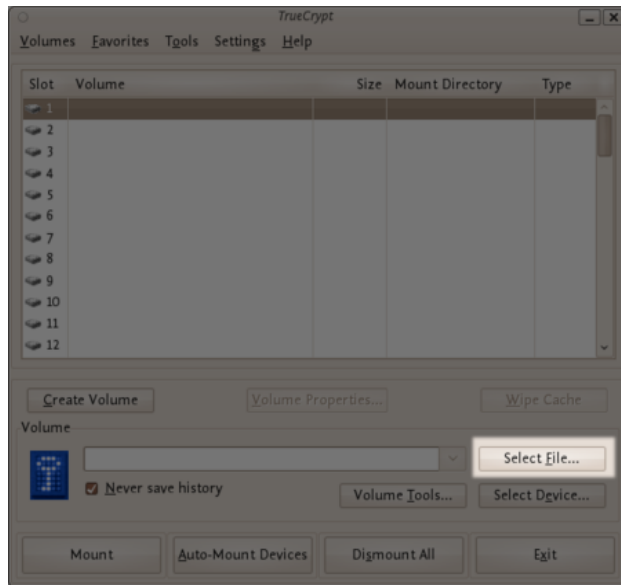


Figure 11.25:Using TrueCrypt

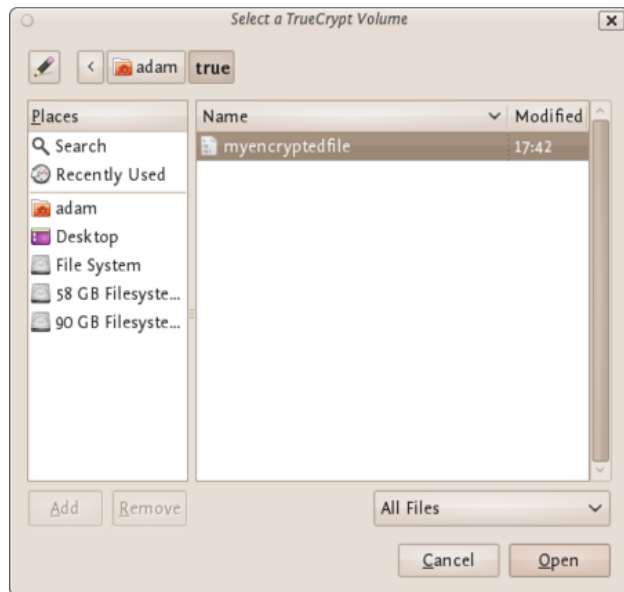


Figure 11.26:Using TrueCrypt



Figure 11.27:Using TrueCrypt

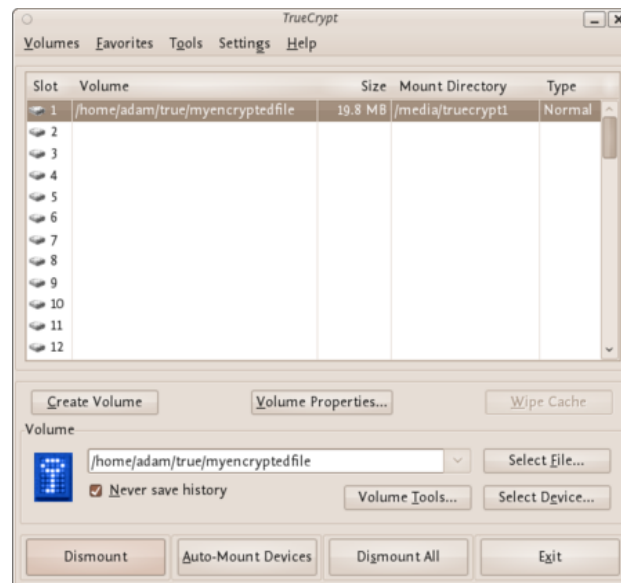


Figure 11.28:Using TrueCrypt

6. Click OK in the password prompt window.

VeraCrypt will now attempt to mount the volume. If the password is correct, the volume will be mounted.

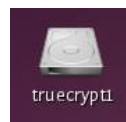


Figure 11.29:Using TrueCrypt

If the password is incorrect (for example, if you typed it incorrectly), VeraCrypt will notify you and you will need to repeat the previous step (type the password again and click OK).

7. We have just successfully mounted the container as a virtual disk 1. The container will appear on your Desktop or you will see it in your file browser.

11.2.3 What does this mean?

The disk that you have just created is completely encrypted and behaves like a real disk. Saving (moving, copying, etc) files to this disk will allow you to encrypt files on the fly. You'll be able to open a file which is stored on a VeraCrypt volume, which will automatically be decrypted to RAM while it is being read, and you won't need to enter your password each time. You'll only need to enter this when your mounting the volume.

11.2.4 Remember to dismount!

To do this right click on the drive and select unmount. This will automatically happen when you turn off your computer but will not happen if you just put the computer on sleep.

11.3 Setting up a hidden volume

A VeraCrypt hidden volume exists within the free space of a typical VeraCrypt volume. Given then the 'outer volume' is accessed it is (almost) impossible to determine if there is a hidden volume within it. This is because VeraCrypt *always* fills the empty space of an encrypted volume with random data. So a hidden volume looks the same as an empty VeraCrypt volume.

To create and use a hidden volume you need two passwords - one each for the outer and inner (hidden) volumes. When you mount (open) the volume you can use either password and that will determine which of the two is opened. If you want to open just the hidden volume you use one password, and if you want to access just the non-hidden encrypted volume you use the other password.

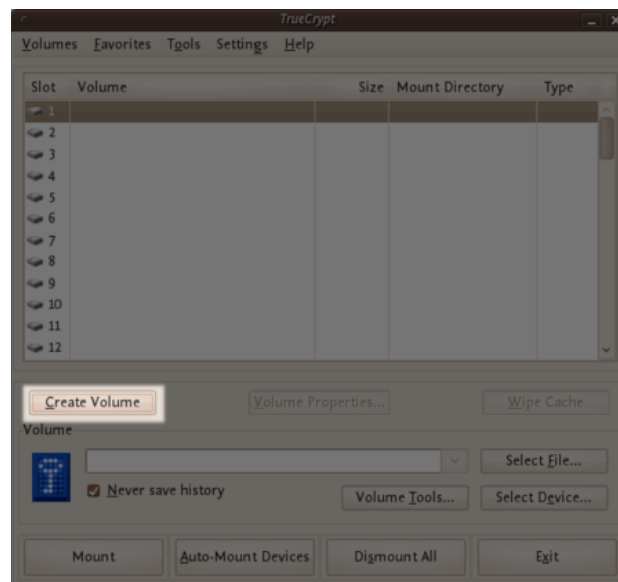


Figure 11.30:Hidden volumes

To create a hidden volume open VeraCrypt and press the ‘Create Volume’ button: The options for half of this process are almost the same as for setting up a standard VeraCrypt volume and then the process continues for setting up the hidden volume but lets go through the entire process step by step anyway. In the screen shown below you just want to stay with the default setting ‘Create an encrypted file container’:



Figure 11.31:Hidden volumes Press

‘Next >’ and continue to the next screen.

In the above screen you want to be sure that you choose the second option ‘Hidden TrueCrypt Volume’. Select this and click on ‘Next >’ you will then be asked to choose the location and name of the VeraCrypt *outer* volume.

Click ‘Select File. . .’ and browse to a location for a new VeraCrypt volume. We will

use the name ‘myencryptedfile’ in this example. Its the same name as we used in the last example so be aware that if you have just followed those instructions you must now create a new volume with a new name.

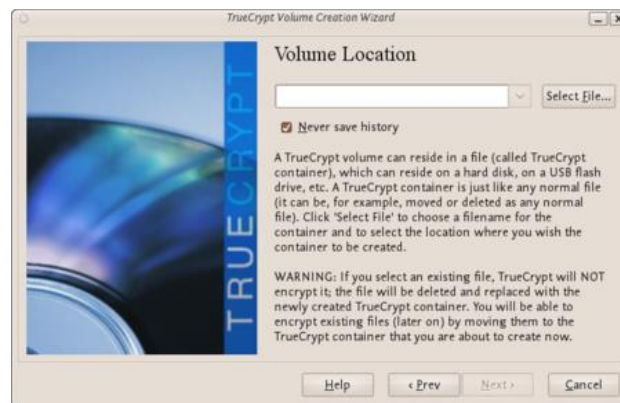


Figure 11.32:Hidden volumes

Browse to the directory where you want to put the outer volume and enter the name of the volume in the field named ‘Name’ as in the example above. When you are satisfied all is well click on ‘Save’. The file browser will close and you return to the Wizard. Click ‘Next >’. Here you are presented with some very technical choices. Don’t worry about them. Leave them at the defaults and click ‘Next >’. The next screen asks you to determine the size of the outer volume. Note that when you do this the maximum inner ‘hidden’ volume size is determined by TrueCrypt. This

maximum size will of course be smaller than the size you are setting on this screen. If you are not sure what the ratio of outer volume size to inner (hidden) volume size is then go through the process now as a 'dummy' run - you can always trash the encrypted volume and start again (no harm done).

So choose the size of the outer volume, I will choose 20MB as shown below:

You cannot set the outer volume size to be larger than the amount of free space you have available on your disk. VeraCrypt tells you the maximum possible size in bold letters so create a volume size smaller than that. Then click 'Next >' and you will be taken to a screen asking you to set a password for the outer (not the hidden, this comes later) volume.

Enter a password that is strong (see the chapter on creating good passwords) and press 'Next >'. Next VeraCrypt wants you to help it create the random data it will fill the volume up with. So wave your mouse around, browse the web, and do whatever you want for as long as you can. When you feel VeraCrypt should be happy then press 'Format'. You will see a progress bar zip by and then you will be presented with the next screen:

You can open the outer volume if you like but for this chapter we will skip that and go ahead to create the hidden volume. Press 'Next >' and VeraCrypt will work out how the maximum possible size of the hidden volume.

When you see the above screen just press 'Next >'. Now you must choose the encryp-

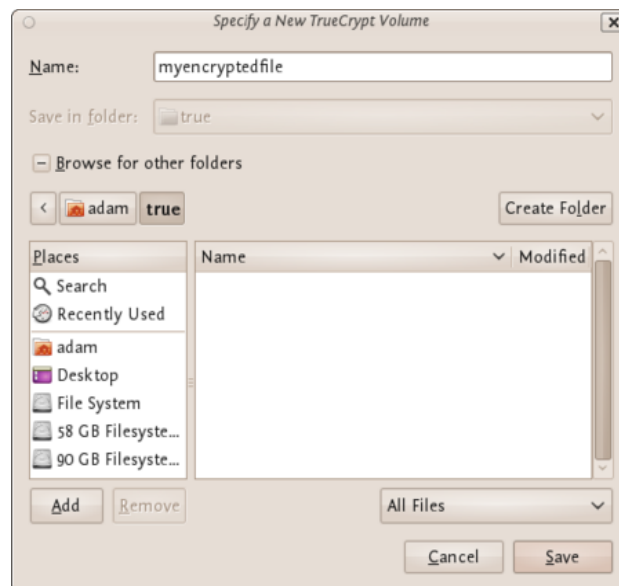


Figure 11.33:Hidden volumes



Figure 11.34:Hidden volumes

tion type for the hidden volume. Leave it at the defaults and press ‘Next >’.

Now you will be asked to choose the size of the hidden volume.

I have set (as you see above) the maximum size as 10MB. When you have set your maximum size press ‘Next >’ and you will be prompted to create a password for the hidden volume.

When creating the password for the hidden volume make sure you make it substantially different fro the password for the outer volume. If someone really does access your drive and finds out the password for the outer volume they might try variations on this password to see if there is also a hidden volume. So make sure the two passwords are not alike.

Enter your password in the two fields and press ‘Next >’.

Leave this window at the defaults and press ‘Next >’ and you will be presented with the same screen you have seen before to generate random data for VeraCrypt. When you are happy click ‘Format’ and you should see the following :



Figure 11.35:Hidden volumes

Click ‘OK’ and keep and exit VeraCrypt. You can now mount the volume as noted in the previous chapter.

11.4 *Securely destroying data*

Just hit the delete button and you are done! No it’s not that easy. To understand how to securely delete data, we have to understand how data is stored. In an analogy to the real world, an explanation of how data is stored follows:

Assume you have a small notebook with 10 pages and you want to write some data in this notebook. You just start writing on the first page up to the end of the notebook. Maybe you decide the information on page 5 must be destroyed. Probably you will just take out the page and burn it.

Unfortunately data on a harddisk doesn’t work this way. A harddisk contains not ten but thousands or maybe even millions of pages. Also it’s impossible to take out a “page” of a harddisk and destroy it. To explain how a harddisk work, we will continue with our 10-page notebook example. But now we will work a little bit different with it. We will work in a way similar to how a harddisk works.

This time we use the first page of our notebook as an index. Assume we write a piece about “WikiLeaks”, then on the first page we write a line “piece about WikiLeaks: see page 2”. The actual piece is then written on page 2.

For the next document, a piece about “Goldman Sachs” we add a line on page 1, “Goldman Sachs: see page 3”. We can continue this way till our notebook is full. Let’s assume the first page will look like this:

- WikiLeaks -> see page 2
- Goldman Sachs -> see page 3
- Monstanto scandal -> see page 4
- Holiday pictures -> see page 5
- KGB Investigation -> see page 6
- Al Jazeeraa contacts -> see page 7
- Iran nuclear program -> see page 8
- Sudan investigation -> see page 9
- Infiltration in EU-politics -> see page 10

Now, let’s decide you want to wipe the “Goldman Sachs” piece, what a harddisk will do, it will only remove the entry on the first page, but not the actual data, your index will be:

- WikiLeaks -> see page 2
- Monstanto scandal -> see page 4
- Holiday pictures -> see page 5
- KGB Investigation -> see page 6
- Al Jazeeraa contacts -> see page 7
- Iran nuclear program -> see page 8
- Sudan investigation -> see page 9
- Infiltration in EU-politics -> see page 10

What we did, we removed only the reference to the article, but if we open page 3, we will still be able to read the Goldman Sachs piece. This is exactly the way what a harddisk does when you “delete” a file. With specialized software it is still able to “recover” page 3.

To securely delete data, we should do the following:

1. Open the “Goldman Sachs” page (page 3)
2. Use an eraser to remove the article there, if done return to page 1
3. Delete the reference in the index on page 1

Well you will be surprised by the similarity between this example and the real world. You know when you removed the article on page 3 with an eraser, it is still possible to read the article slightly. The pencil leaves a track on the paper because of the pressure of the pencil on the paper and also you will be unable to erase all of the graphite. Small traces are left behind on the paper. If you really need this article, you can reconstruct (parts) of it, even if it’s erased.

With a harddisk this is very similar. Even if you erased every piece of data, it is sometimes possible with (very) specialized hardware to recover pieces of the data. If the data is very confidential and must be erased with the greatest care, you can use software to “overwrite” all pieces of data with random data. When this is done multiple times, this will make the data untraceable.

11.4.1 A note on Solid State Hard Drives

The instructions below explain how to use file deletion tools to securely delete files from your hard drives. These tools rely on the Operating System you are using being able to directly address every byte on the hard drive in order to tell the drive “set byte number X to 0”. Unfortunately, due to a number of advanced technologies used by Solid State Drives (SSDs) such as TRIM, it is not always possible to ensure with 100% certainty that every part of a file on an SSD has been erased using the tools below.

11.4.2 Securely delete data under Windows

For Windows there is a good open source tool called “File Shredder”. This tool can be downloaded from <http://www.fileshredder.org>

The installation is very straightforward, just download the application and install it by hitting the next button. After installation this application will automatically start. You can then start using it for shredding files. However the best part of the program is that you can use it from within windows itself by right clicking on a file.

1. Click right on the file you want to shred, and choose File Shredder -> Secure delete files
2. A pop-up asks if you really want to shred this file
3. After confirming, there your file goes. Depending on the size of the file this can take a while

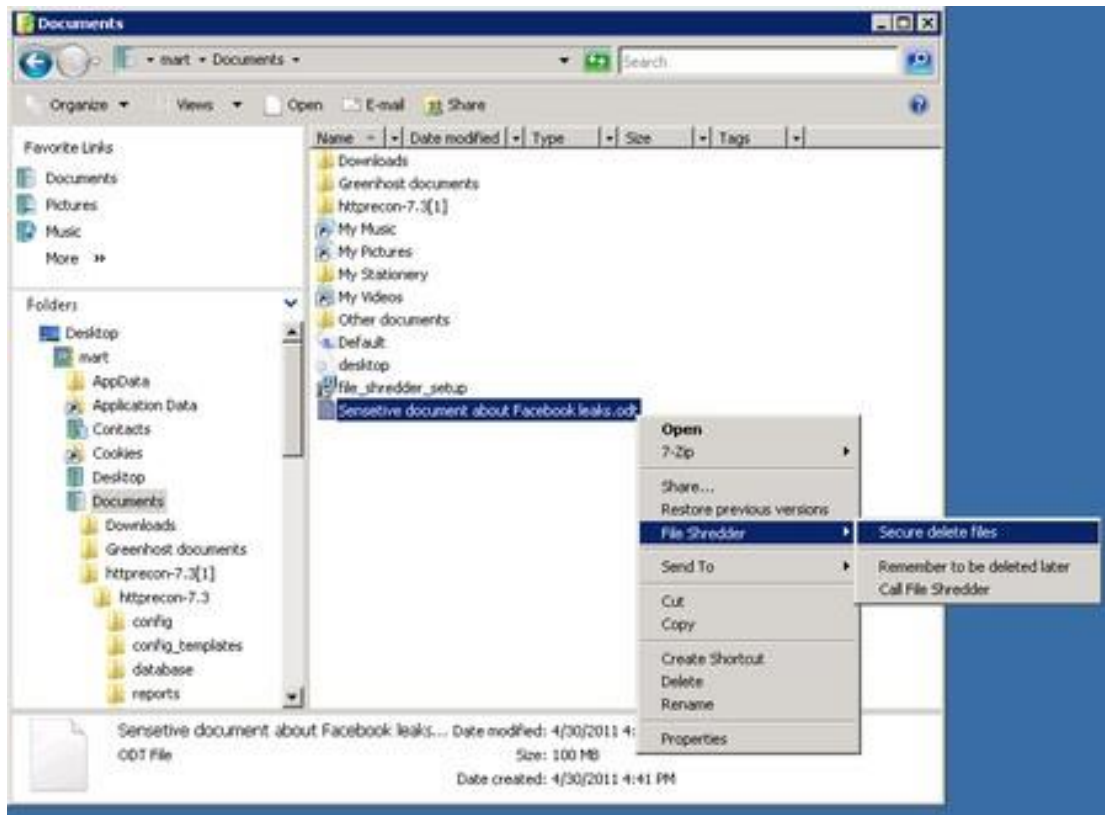


Figure 11.36:Destroying data

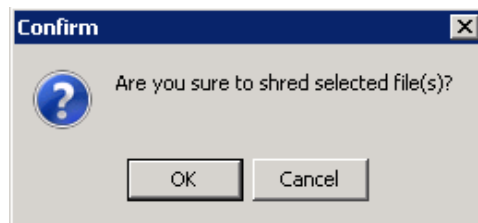


Figure 11.37:Destroying data



Figure 11.38: Destroying data

11.4.3 Securely delete data under MacOSX

There are basically two build-in steps to make to securely delete your data on Mac OSX.

1. Erase the free-space on your hard-drive containing all the data of items which are deleted in an insecure way.
2. Make sure that every file from then on is always securely deleted.

We start with the first one:

Erasing Free Space

1. Open Disk-Utility which resides in the Utilities folder inside the Applications folder.
2. Select your hard drive and click on 'Erase Free Space'.
3. Three options will appear, from top to bottom more secure, but also they take much more time to complete. Read the descriptions on each one of them to get an idea from what will happen if you use them and then choose which one might suite your needs the best and click 'Erase free Space'.

If time is no issue, then use the most secure method and enjoy your free time to get a good coffee while your Mac crunches away on this task. If the crooks are already knocking on your front-door you might want to use the fastest way.



Figure 11.39: Destroying data

Securely Erasing Files

Now that your previously deleted data is once and for ever securely erased you should make sure that you don't create any new data that might be recovered at a later date.

1. To do this open the finder preferences under the Finder Menu.
2. Go to the advanced tab and tick 'Empty trash securely'. This will make sure that every time you empty your trash all the items in it will be securely deleted and are really gone!

Note: Deleting your files securely will take longer than just deleting them. If you have to erase big portions of unimportant data (say your movie and mp3 collection) you may want to untick this option before doing so.

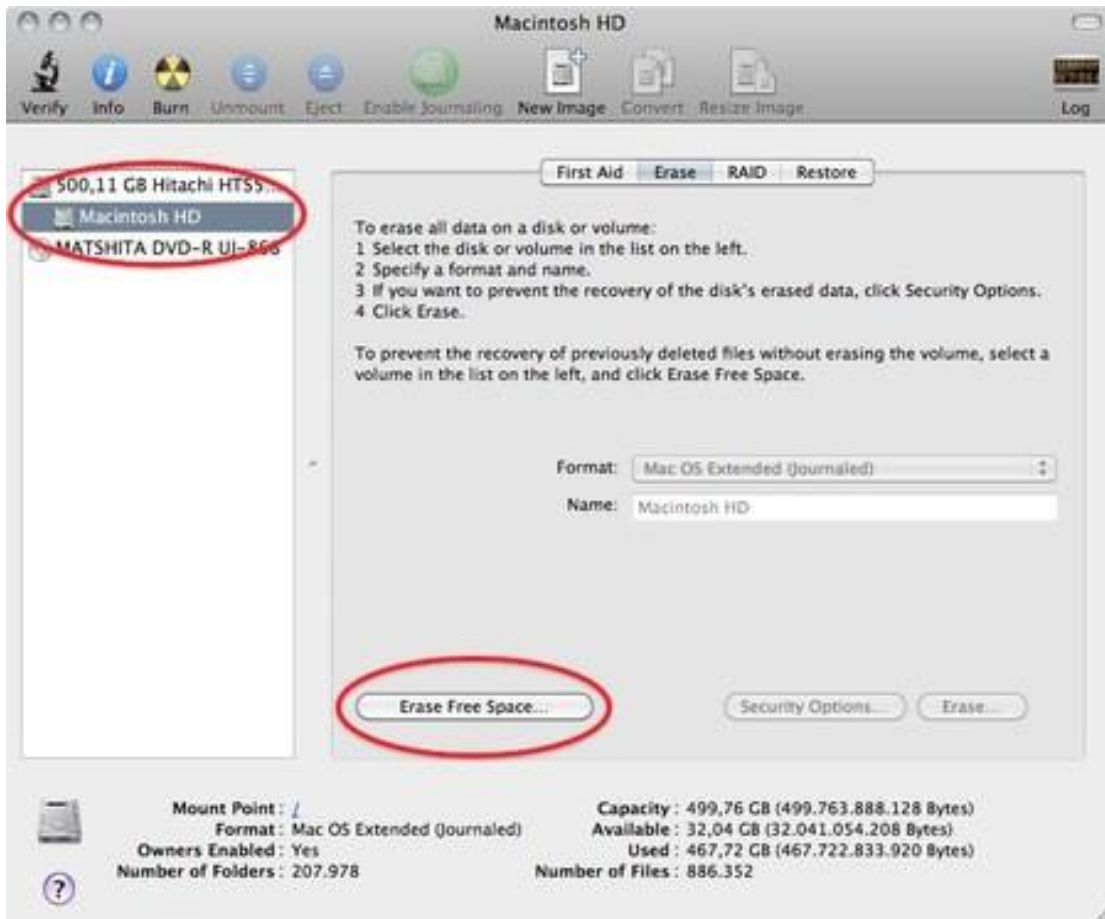


Figure 11.40: Destroying data



Figure 11.41: Destroying data



Figure 11.42: Destroying data



Figure 11.43: Destroying data

11.4.4 Securely delete data under Ubuntu/Linux

Unfortunately currently there is no graphical user interface available for Ubuntu to delete files secure. There are two command-line programs available though:

- shred
- wipe

Shred is installed in Ubuntu by default and can delete single files. Wipe is not installed by default but can easily be installed with using Ubuntu Software Center or if you understand the command line you can install it with `apt-get install wipe`. Wipe is a little more secure and has nicer options.

It is possible make access to these program's easy by adding it as an extra menu option

1. We assume you are familiar with the Ubuntu Software Center. To add the securely wipe option, it's required to install these two programs *wipe* and *nautilus-actions*

If the two programs are installed follow the following steps. If they are not installed use the Ubuntu Software Center to install them or on the command line simply type `apt-get install nautilus-actions wipe`

2. Open the "Nautilus Actions Configuration" from the System -> Preferences menu
3. We have to add a new action. To do this, start clicking on the "create new action button", the first option in the toolbar
4. Next is describing the new action. You can give the action every name you wish. Fill out this title in the "Context label" field. In this example we used "Delete file securely"
5. Click on the second tab ("Command"), here is how we specify the action we want. In the field "Path", type "wipe", in the field parameters type "`-rf %M`", please be sure about the capitalisation of all characters here, this is very important.
6. Next is specifying the conditions, click on the conditions tab and choose the option "Both" in the "Appears if selection contains. . ." box. With this option you can wipe both files and folders securely. If done, click the save button (second item on the icon bottom toolbar) or use the menu File->Save
7. Now close the Nautilus Actions Configuration tool. Unfortunately, after this, you have to re-login into your system, so either reboot or logout/login.



Figure 11.44: Destroying data

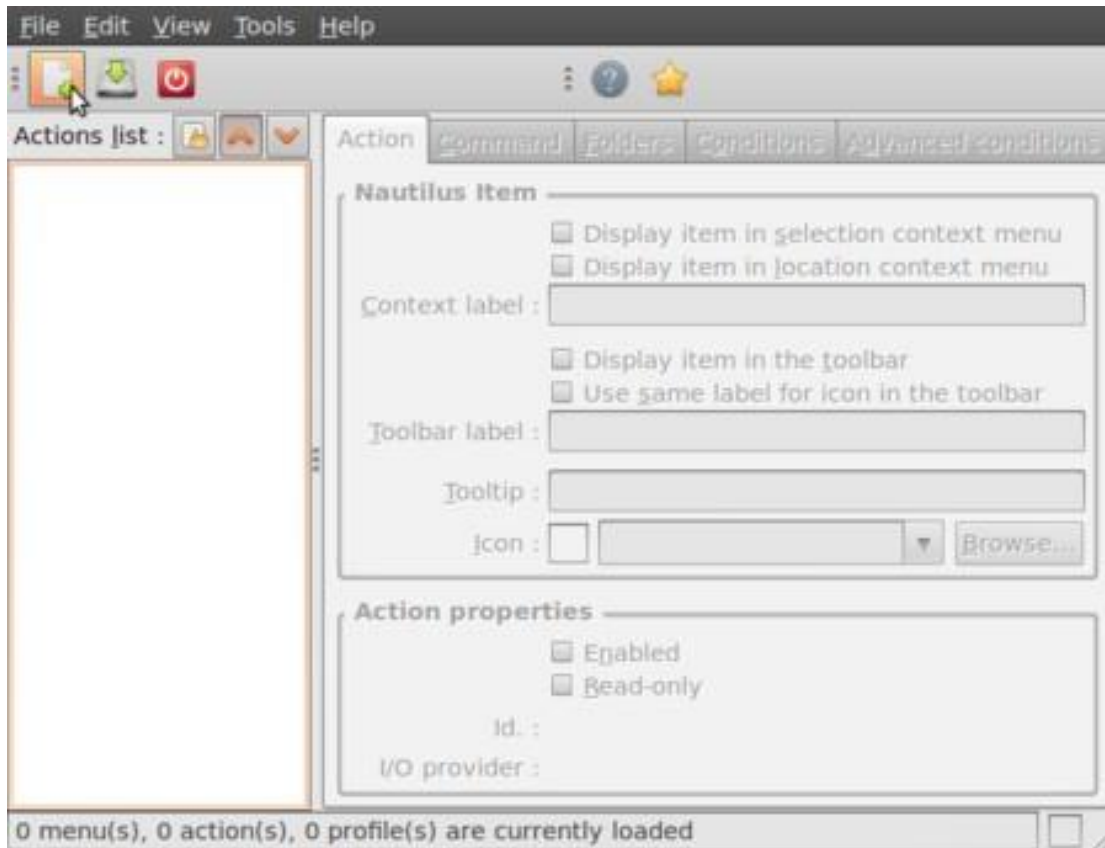


Figure 11.45: Destroying data

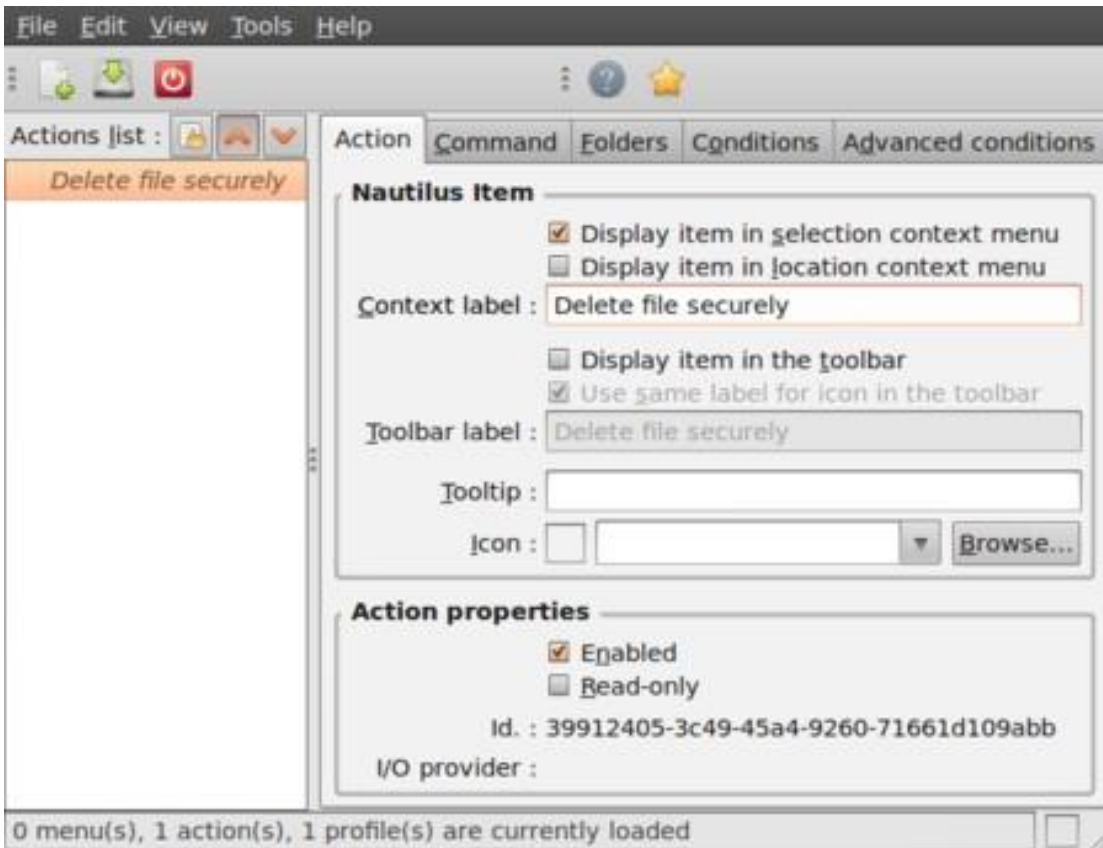


Figure 11.46: Destroying data

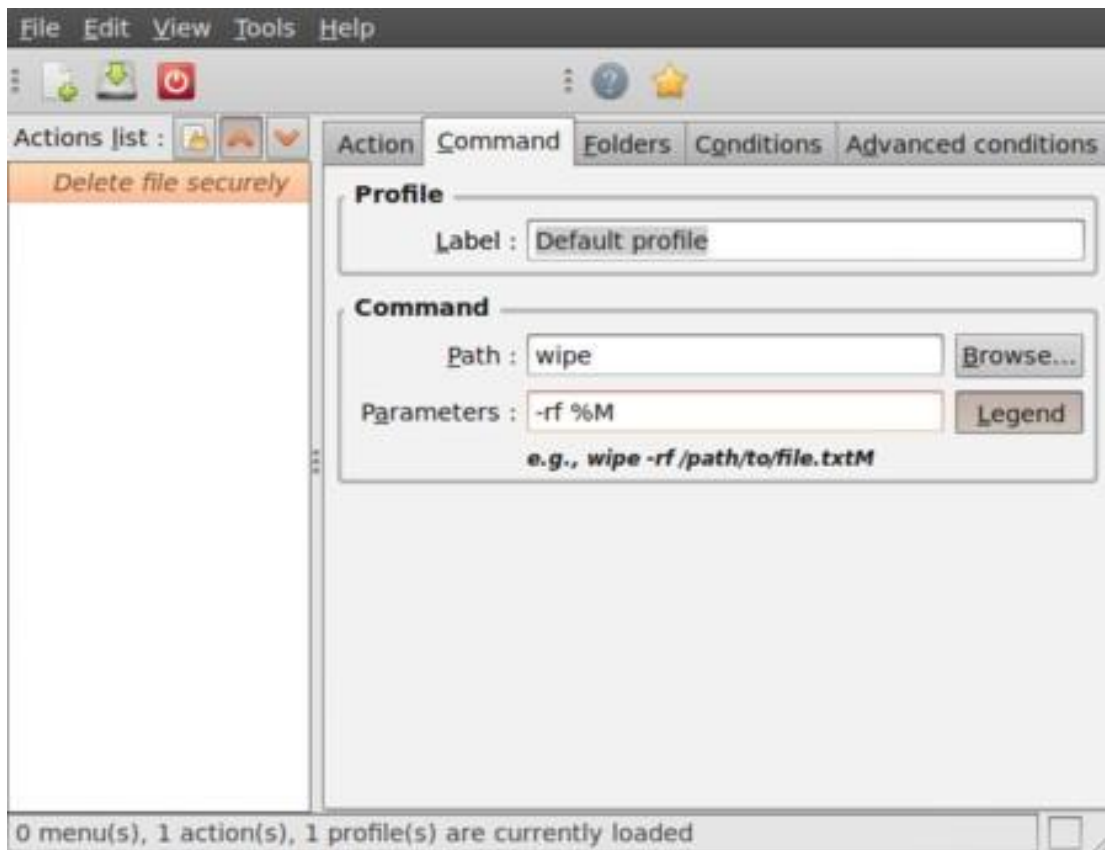


Figure 11.47: Destroying data

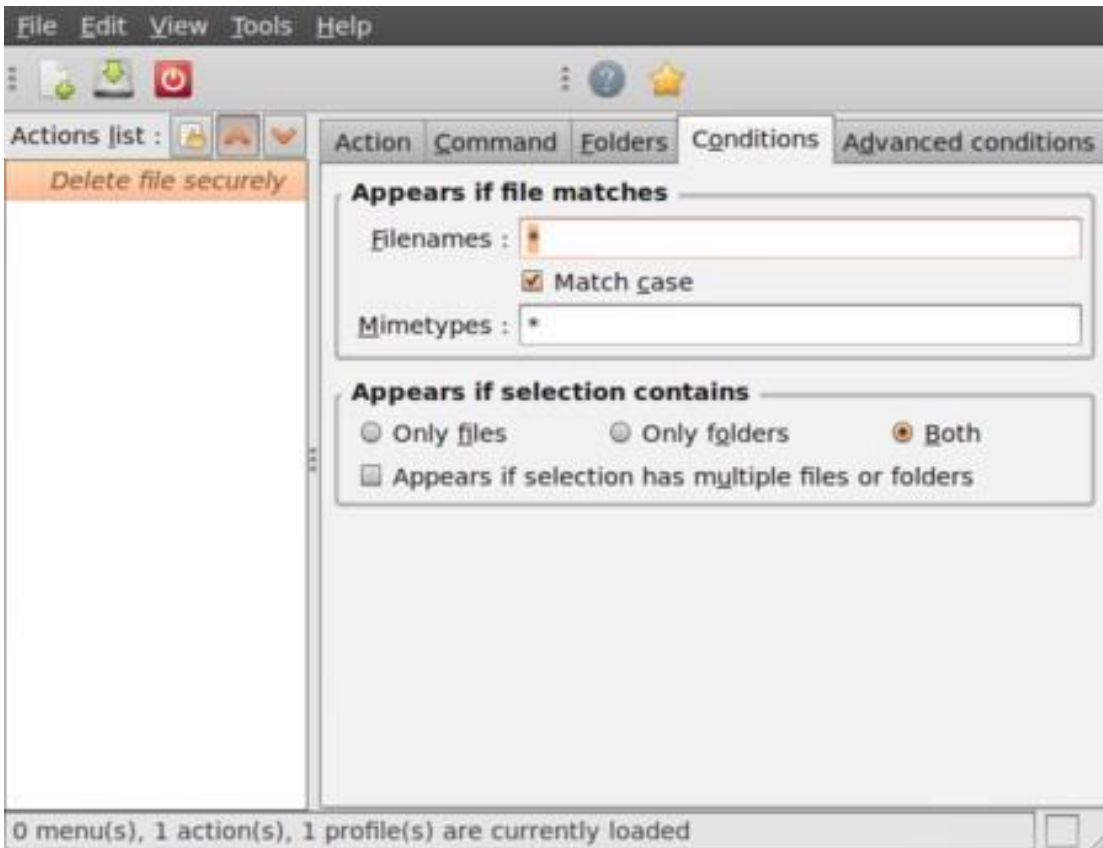


Figure 11.48: Destroying data

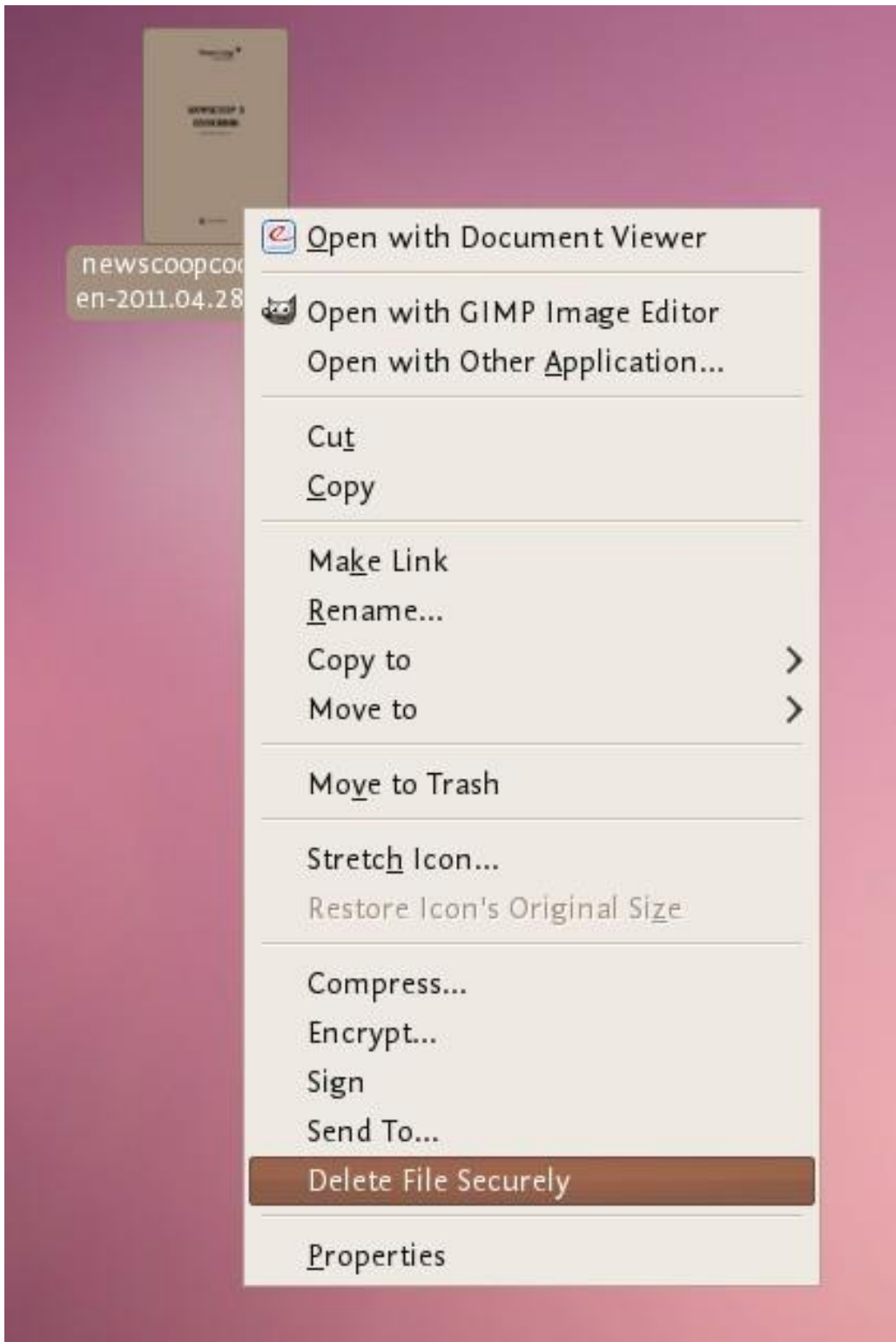


Figure 11.49: Destroying data

8. Now browse to the file you want to securely delete and right click:

Choose ‘Delete File Securely’. The file will then be wiped ‘quietly’ - you do not get any feedback or notice that the process has started or stopped. However the process is underway. It takes some time to securely delete data and the bigger the file the longer it takes. When it is complete the icon for the file to be wiped will disappear. If you would like to add some feedback you can change the parameters field in Nautilus Actions Configuration tool to this:

```
-rf %M | zenity --info --text "your wipe is underway, please be patient.
```

The file to be wiped will disappear shortly."

The above line will tell you the process is underway but you will not know the file is deleted until the icon disappears.

11.5 About LUKS

LUKS, short for *Linux Unified Key Setup*, is the default method for disk encryption on Linux. It can be used to enable *Full Disk Encryption* during installation with a single click, or to encrypt individual partitions on external hard disks or usb sticks later on. Please note that *Full Disk Encryption* is hard to enable **after** the installation as it requires moving all existing files temporarily as encrypting a device requires formatting it.

- Advantages: LUKS is available through dm-crypt which is part of the Linux kernel, so it doesn't need any further software to be installed.
- Disadvantages: Unlike with Veracrypt, it is not possible to use it with other Operating Systems (yet), so if you use LUKS to encrypt a USB drive, you can only use it on Linux machines, but not on Windows or Mac OS.

If you want to encrypt a device after the Linux installation completed, you can use the *Disks* utility which can be found in most Linux distribution's *System Settings*.

11.5.1 Starting *Disks*

On Ubuntu, start *Disks* by pressing the Windows key and A, typing “disks” and selecting the corresponding program as shown below:

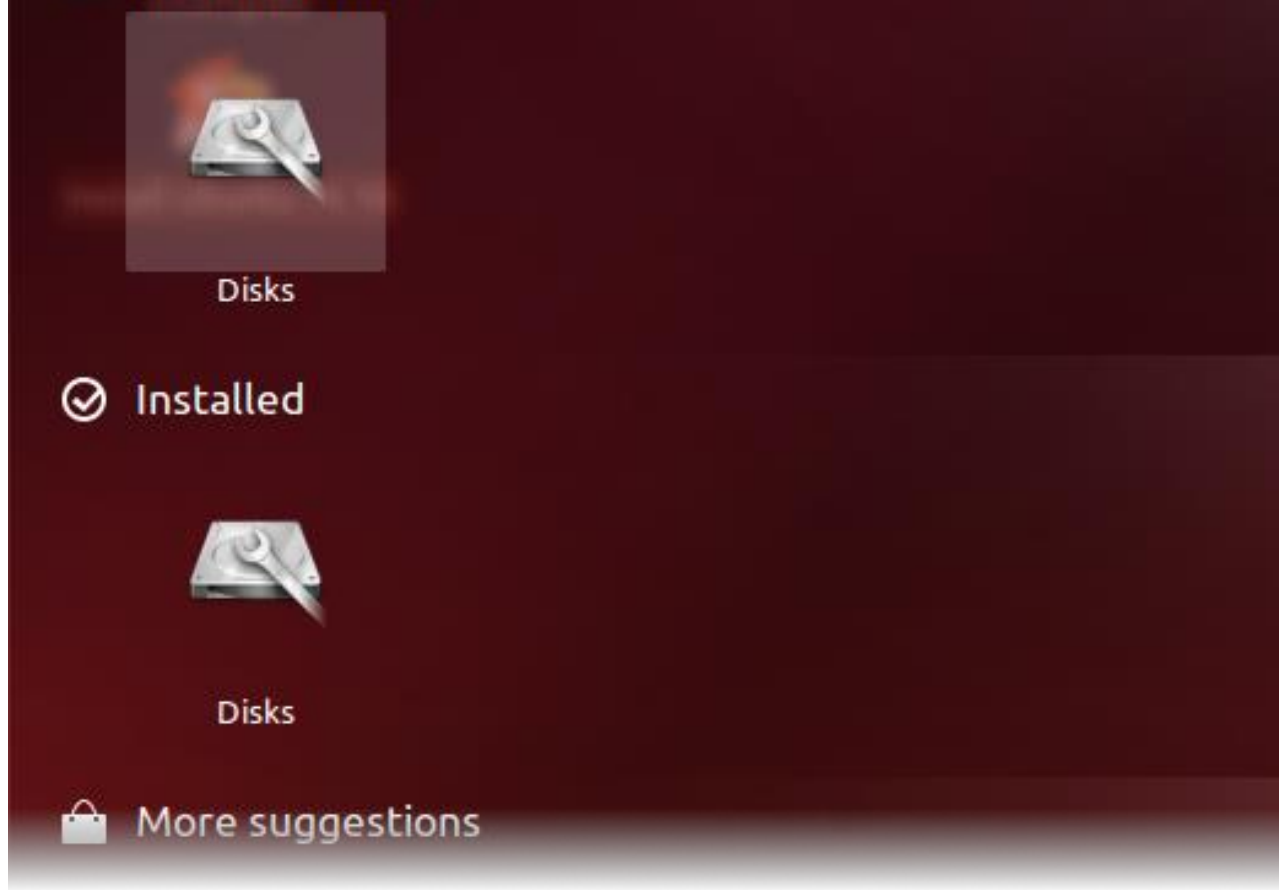


Figure 11.50: Launching Disks

11.5.2 Encrypting a device

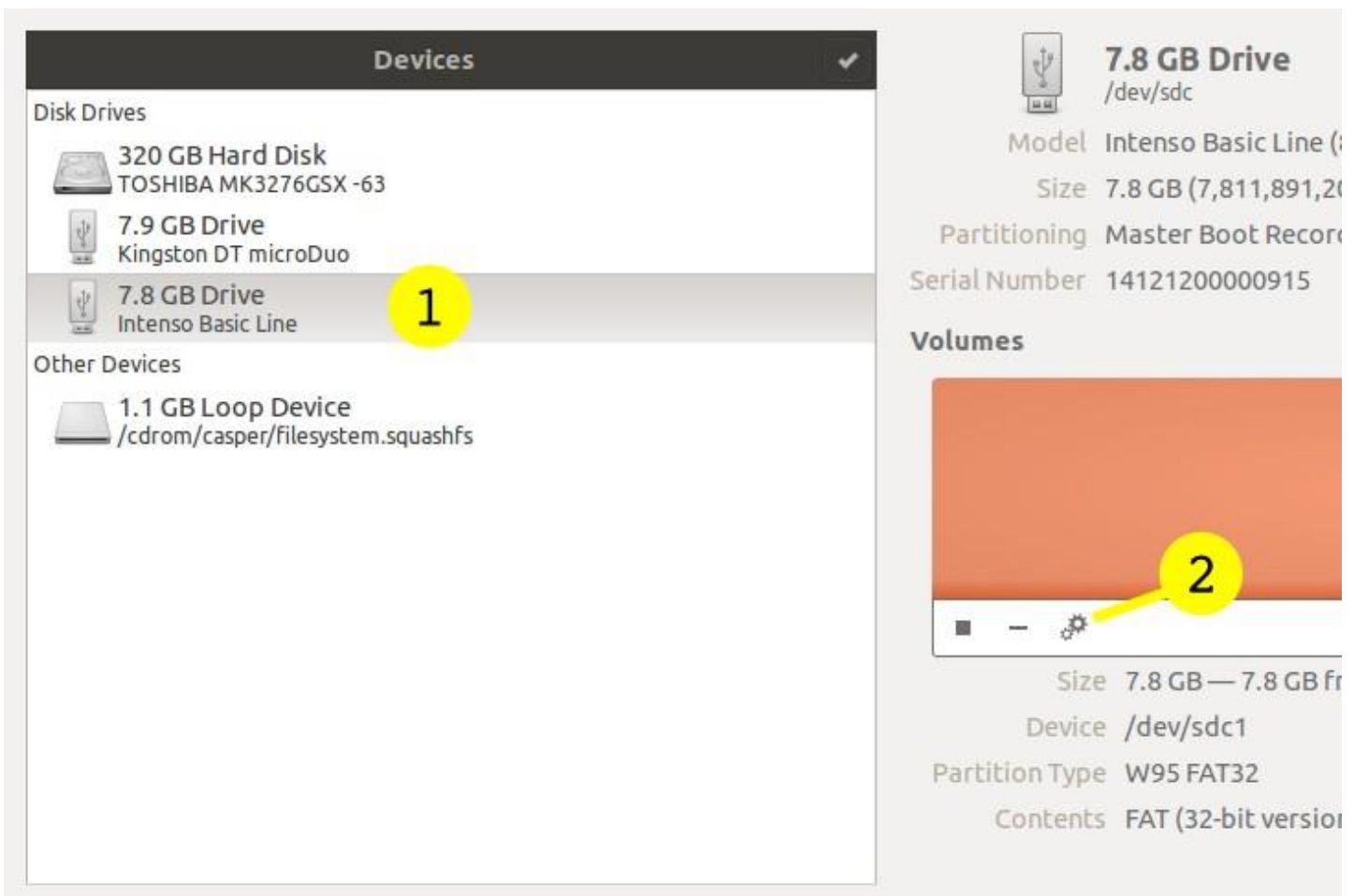


Figure 11.51:Disks main window

On the left hand side you will find a list of all storage devices plugged into your computer.

Select the one you want to encrypt (step 1) (in this case a usb stick), and then on the right hand side, click on the cog wheels and “Format. . .”. A dialog will appear where you can select if the existing data on the device shall be completely overwritten (that can take up to several hours depending on the size and performance of the device) or just formatted. Please note that even if you choose to encrypt the device, data, that was present before will be recoverable if you don’t choose to overwrite it completely.

No matter what you choose for the field *Erase*, select “Encrypted, compatible with Linux systems (LUKS+Ext4)” for *Type*, give it a name and a strong passphrase (see chapter 8 on that matter), and click *Format. . .*

On the confirmation screen make sure you selected the correct device as data recovery

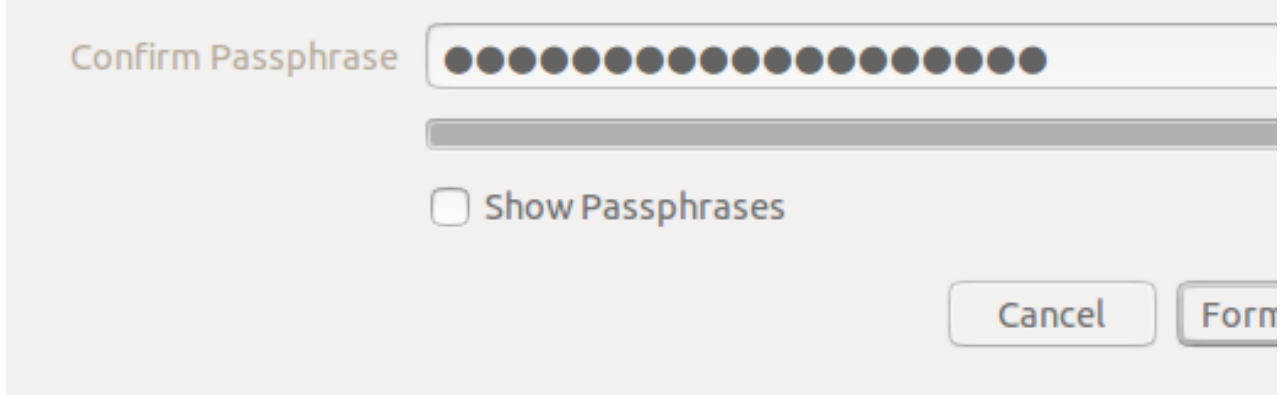


Figure 11.52:“Format. . .” dialog

is a cumbersome tasks – if possible at all.

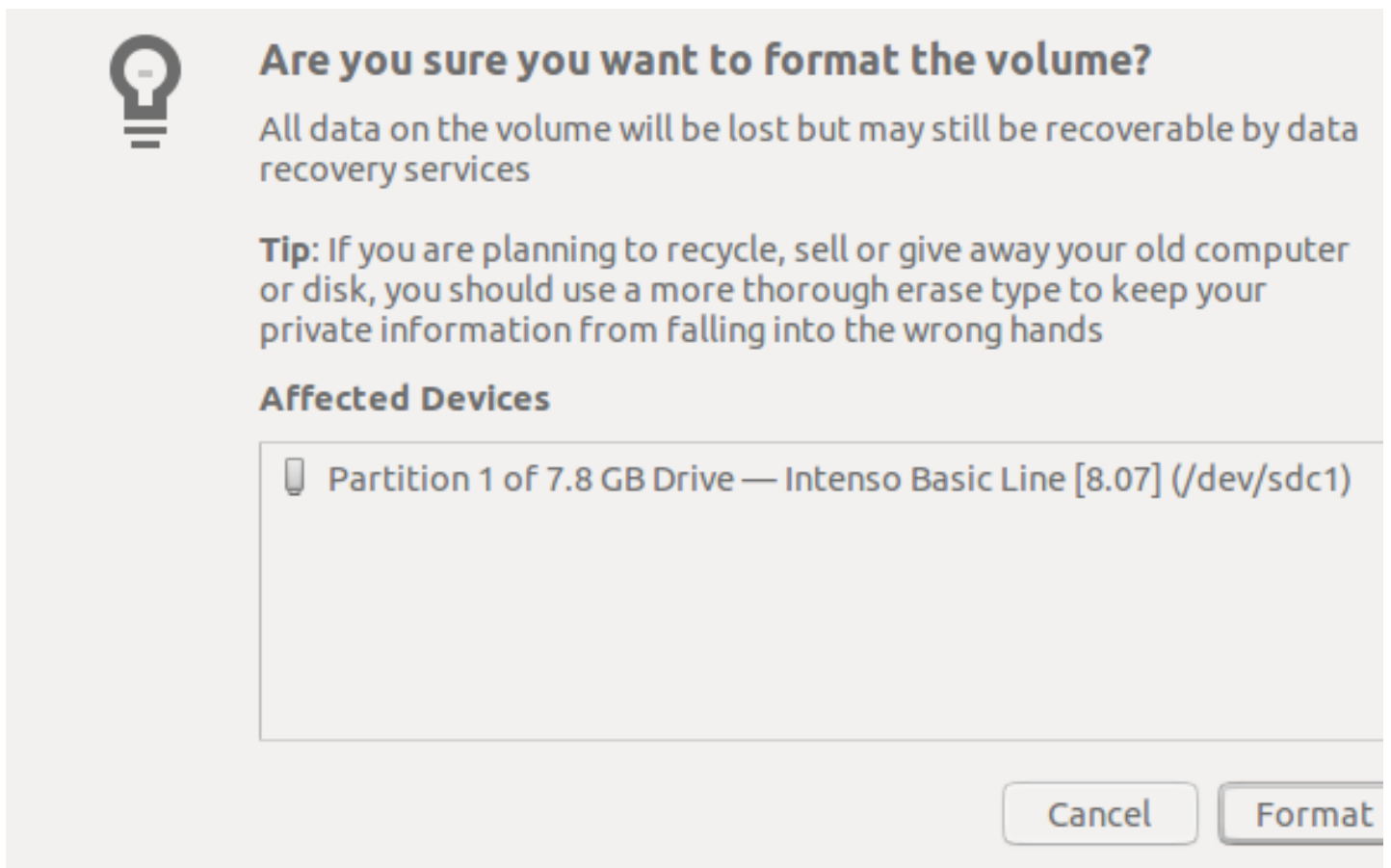
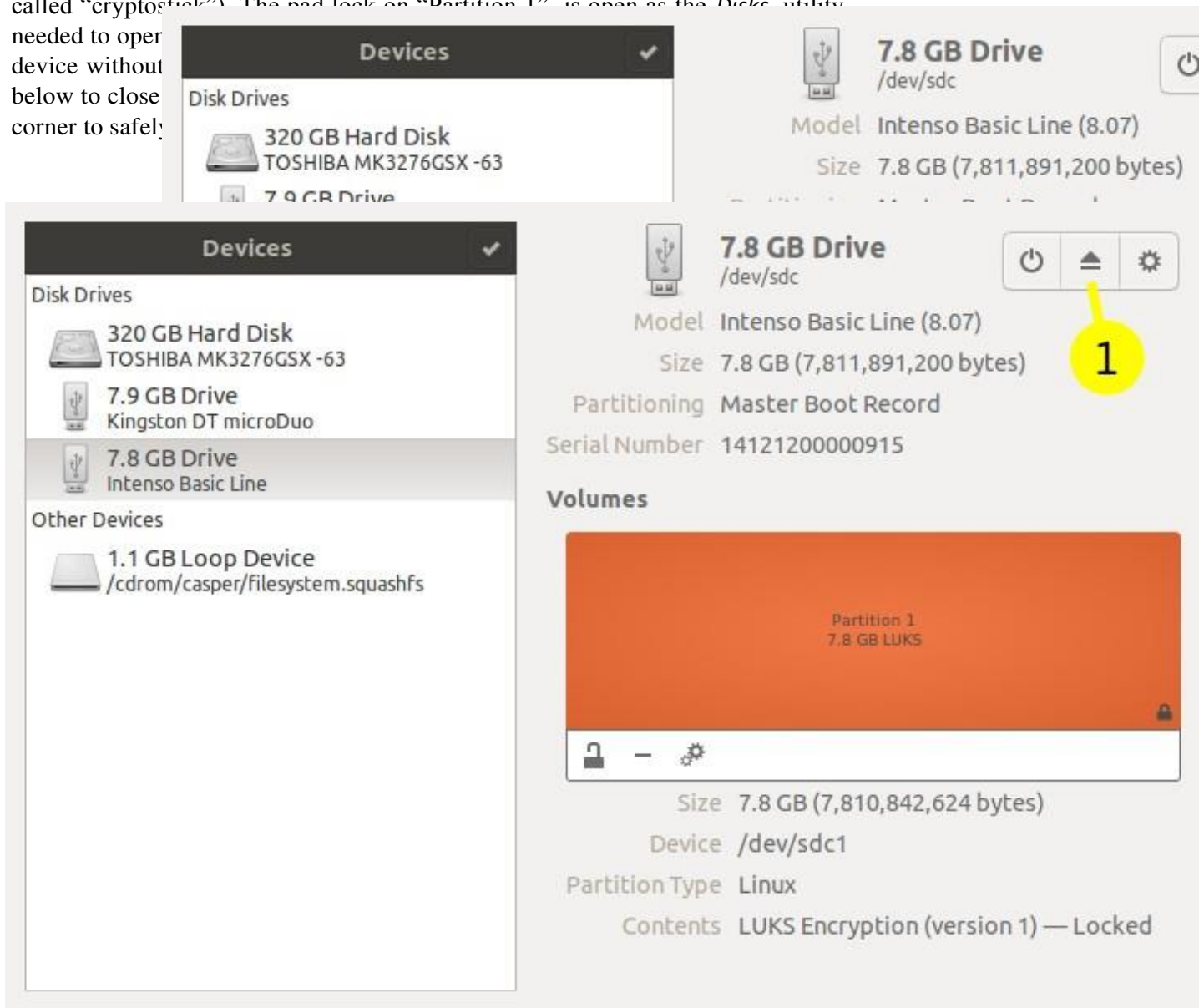


Figure 11.53:Confirmation step

Back on the main window the device now consists of two layers. One is the physical storage (here called “Partition 1”) and the other a virtual device which is

created by the LUKS system to give you access to the encrypted device (here called “crypto-stief”). The red lock on “Partition 1” is open as the *Disks* utility needed to open the device without a password. Below are the steps needed to open the device without a password. Below to close the utility in the bottom right corner to safely



11.5.3 Using an encrypted device

This is quite straight-forward. Plug it in, enter the passphrase and click *Connect*. If the file manager does not open automatically, the device will be available when you do.

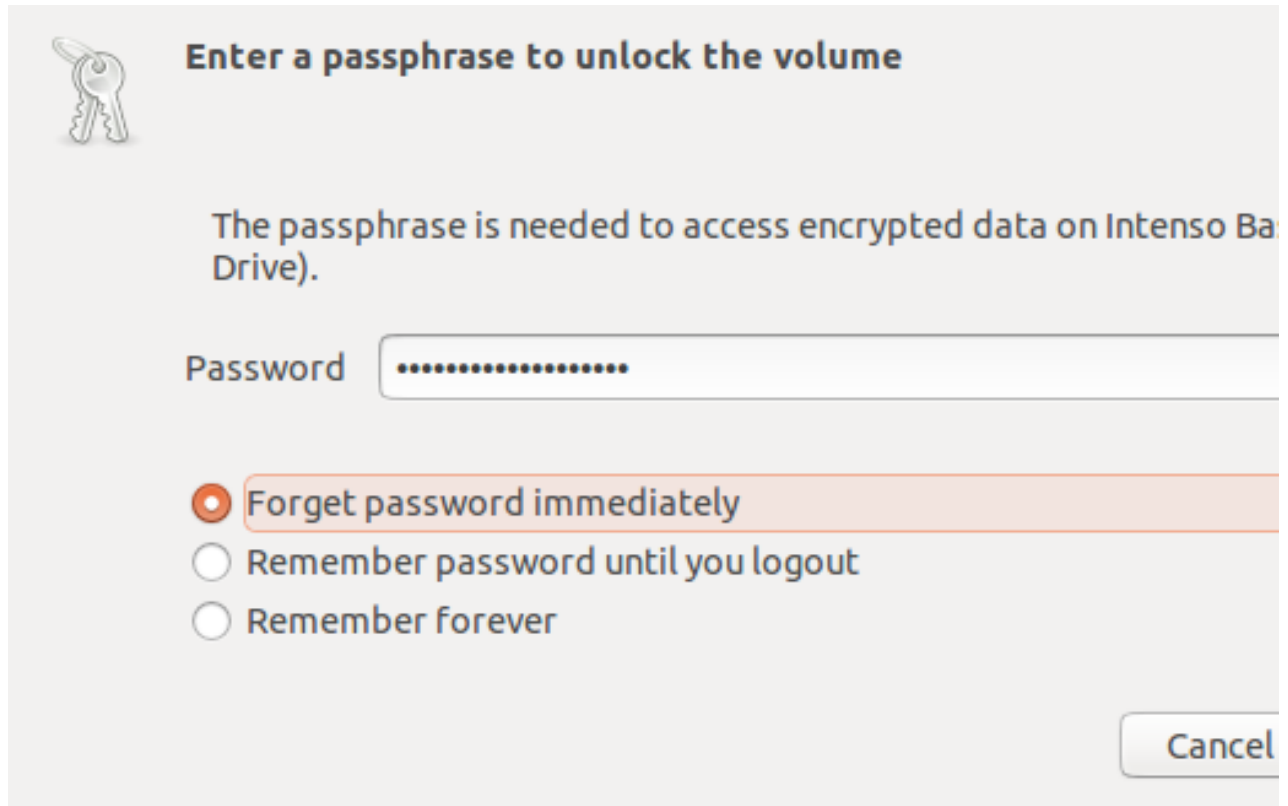


Figure 11.54:Eject the LUKS device

12 Call Encryption

12.1 Installing CSipSimple

CSipSimple is a program for Android devices that allows for making encrypted calls. Naturally the calling software isn't enough on its own and we need a communication network to enable us to make calls.

12.1.1 Introducing The OSTN Network

If you already know about OSTN and have an account, you can skip this section.

The Guardian Project's OSTN (Open {Secure, Source, Standards} Telephony Network - <https://guardianproject.info/wiki/OSTN>) is an attempt to define a standard Voice over IP (VoIP) setup using the Session Initiation Protocol (SIP) that enables end-to-end encrypted calls.

Similar to e-mail, SIP allows people to choose their service provider while still being able to call each other even if they are not using the same provider. Yet, not all SIP providers offer OSTN and some do not support ZRTP. Once a connection between two parties is established, the call is encrypted according to the Secure Real-time Transport Protocol (SRTP).

According to the Secure Real-time Transport Protocol (SRTP).

A majority of encrypting VoIP applications currently use Session Description Protocol Security Descriptions for Media Streams (SDES) with hop-by-hop Transport Layer Security (TLS) to exchange secret master keys for SRTP. This method is not end-to-end secure as the SRTP keys are sent in plaintext to any SIP proxy or provider involved in the call.

ZRTP is a cryptographic key-agreement protocol to negotiate the keys for end-to-end encryption between two parties. ZRTP end points use the media stream rather than the signaling stream to establish the SRTP encryption keys. Since the media stream is a direct connection between the calling parties, there is no way for the SIP providers or proxies to intercept the SRTP keys. ZRTP provides a visual reassurance to end-users that they have a secure line. By reading and comparing a word pair, users can be certain that the key exchange has completed.

Open Secure Telephony (<https://ostel.me/>) is a testbed for OSTN that worked well at the time of writing this book. At https://ostel.me/users/sign_up you can sign up and create an account. You can also check the OSTN page listed above for other providers.

12.1.2 CSipSimple

CSipSimple is a free and open source client for Android that works well with OSTN. You can find it at <https://market.android.com/details?id=com.csipsimple>

To use CSipSimple with ostel.me, select OSTN in the generic wizards when creating an account and enter username, password and server as provided after signing up at https://ostel.me/users/sign_up

Once you call another party with CSipSimple you see a yellow bar with ZRTP and the verification word pair. You now have established a secure voice connection that cannot be intercepted. Still, you should be aware that your phone or the phone of the other party could be set up to record the conversation.

Basic steps:

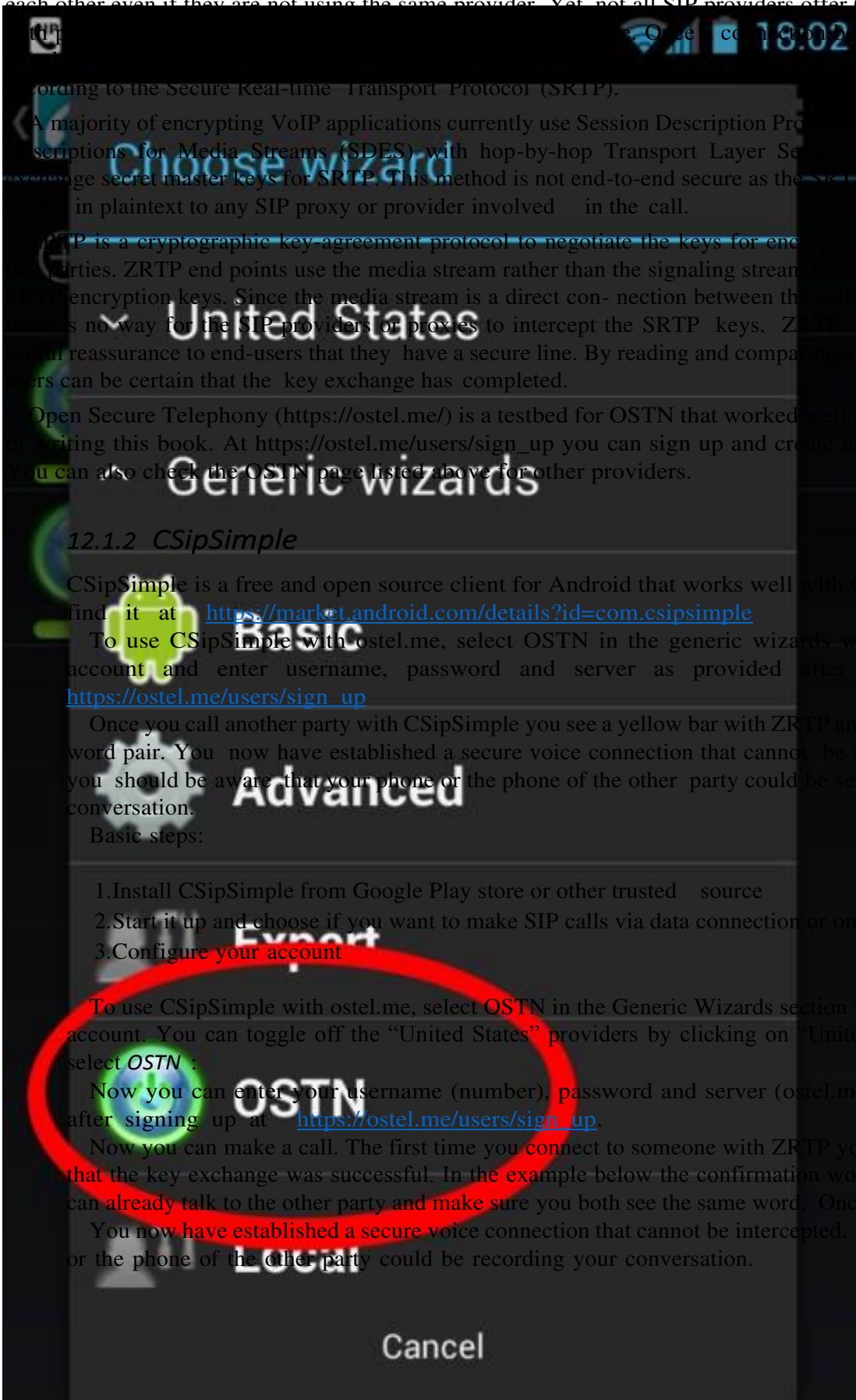
1. Install CSipSimple from Google Play store or other trusted source
2. Start it up and choose if you want to make SIP calls via data connection or only WiFi
3. Configure your account


To use CSipSimple with ostel.me, select OSTN in the Generic Wizards section when creating an account. You can toggle off the “United States” providers by clicking on “United States”. Now select *OSTN* :

Now you can enter your OSTN username (number), password and server (ostel.me) as provided after signing up at https://ostel.me/users/sign_up.

Now you can make a call. The first time you connect to someone with ZRTP you have to verify that the key exchange was successful. In the example below the confirmation word is “cieh”, you can already talk to the other party and make sure you both see the same word. Once done, press ok.

You now have established a secure voice connection that cannot be intercepted. Beware that your phone or the phone of the other party could be recording your conversation.



 Edit

Account name
OSTN

User name
Sip account login (do not write the @sip.server)

Password
Password for your account

Server
SIP server domain/IP[:port]

Cancel | Save

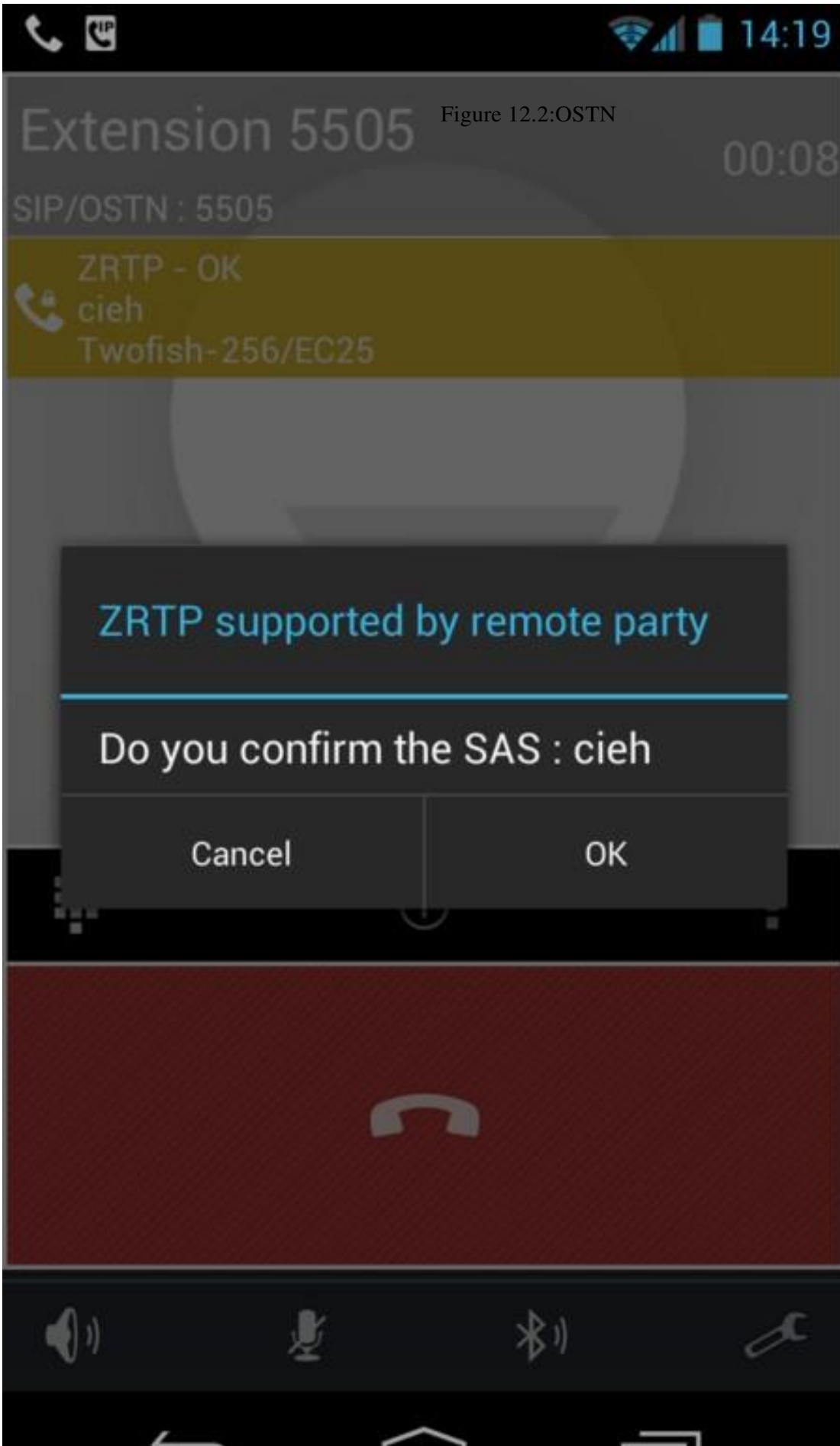


Figure 12.2:OSTN

13 Instant Messaging Encryption

13.1 *Setting up Encrypted Instant Messaging*

13.1.1 *Android - Installing Gibberbot*

<https://guardianproject.info/apps/gibber/>

Gibberbot is a secure chat client capable of end-to-end encryption. It works with Google, Facebook, any Jabber or XMPP server. Gibberbot uses the Off-The-Record encryption standard (OTR) to enable true verifiable end-to-end encrypted communications.

You can install Gibberbot through the Google Play store or from another trusted source.

You can securely chat with other programs with OTR support such as Adium, Pidgin on the desktop, Gibberbot on Android or ChatSecure on iOS.

13.1.2 *iOS - Installing ChatSecure*

<http://chrisballinger.info/apps/chatsecure/>

ChatSecure is a secure chat client capable of end-to-end encryption. It works with Google, Facebook, any Jabber or XMPP server. ChatSecure uses the Off-the-Record encryption standard (OTR) to enable true verifiable end-to-end encrypted communications.

You can install ChatSecure through the iTunes store

You can securely chat with other programs with OTR support such as Adium, Pidgin on the desktop, Gibberbot on Android or ChatSecure on iOS.

13.1.3 *Ubuntu - Installing Pidgin*

<http://pidgin.im/>

Pidgin is a secure chat client capable of end-to-end encryption. It works with Google, Facebook, any Jabber or XMPP server. Pidgin uses the Off-the-Record encryption standard (OTR) to enable true verifiable end-to-end encrypted communications.

You can install via Ubuntu Software Center, search for pidgin-otr to install pidgin and the pidgin otr plugin.

Once installed you can enable otr for any account you setup in pidgin.

You can securely chat with other programs with OTR support such as Adium, Pidgin on the desktop, Gibberbot on Android or ChatSecure on iOS.

13 Instant Messaging Encryption

13.1.4 OS X - Installing Adium

<http://www.adium.im/>

Adium is a secure chat client capable of end-to-end encryption. It works with Google, Facebook, any Jabber or XMPP server. Adium uses the Off-the-Record encryption standard (OTR) to enable true verifiable end-to-end encrypted communications.

Installing Adium is similar to installing most Mac OS X applications.

1. Download the Adium disk image from <http://www.adium.im/>.
2. If an Adium window does not open automatically, double click the downloaded file
3. Drag the Adium application to your Applications folder.
4. “Eject” the Adium disk image, which has an icon of a drive
5. The Adium disk image will still be present in your download folder (probably on your desktop). You can drag this file to the trash, as it is no longer needed.
6. To load Adium, locate it in the Applications folder and double click.

You can securely chat with other programs with OTR support such as Adium, Pidgin on the desktop, Gibberbot on Android or ChatSecure on iOS.

13.1.5 Windows - Installing Pidgin

<http://pidgin.im/>

Pidgin is a secure chat client capable of end-to-end encryption. It works with Google, Facebook, any Jabber or XMPP server. Pidgin uses the Off-the-Record encryption standard (OTR) to enable true verifiable end-to-end encrypted communications.

To use Pidgin with OTR on Windows, you have to install Pidgin and the OTR plugin for Pidgin.

1. Download the latest version of Pidgin for Windows from <http://www.pidgin.im/download/windows/>
2. Run the Pidgin Installer
3. Download the latest version of “OTR plugin for Pidgin” at <http://www.cypherpunks.ca/otr/#downloads>
4. Run the OTR Plugin Installer

Now you can use OTR with any account you setup in Pidgin.

You can securely chat with other programs with OTR support such as Adium, Pidgin on the desktop, Gibberbot on Android or ChatSecure on iOS.

13.1.6 All OS - *crypto.cat*

<https://crypto.cat>

Cryptocat is an open source web application intended to allow secure, encrypted online chatting. Cryptocat encrypts chats on the client side, only trusting the server with data

13.1 Setting up Encrypted Instant Messaging

that is already encrypted. Cryptocat is delivered as a browser extension and offers plugins for Google Chrome, Mozilla Firefox and Apple Safari.

Cryptocat intends to provide means for impromptu, encrypted communications that offer more privacy than services such as Google Talk, while maintaining a higher level of accessibility than other high-level encryption platforms, and furthermore allows for multiple users in one chat room.

13.1.7 Chat Log Files

Some of the Chat Clients listed above e.g. Adium, store plaintext, unencrypted Chat Logs, often by default, even when the OTR “security / privacy” plug-in is installed.

If you are taking OTR precautions to protect your chats from snoopers over the wire or over the air, you should either double check that you have manually switched off Chat Session Logging, or ensure that the Chat Logs you deliberately intend to keep are created on an encrypted disk drive or volume, in case your computer is lost, stolen or seized. It is also worth asking the person you are chatting with if they are inadvertently logging the chat with their own Chat Client software.

14 Secure File Sharing

14.1 *Installing I2P on Ubuntu Lucid Lynx (and newer) and derivatives like Linux Mint & Trisquel*

1. Open a terminal and enter:

```
sudo apt-add-repository ppa:i2p-maintainers/i2p
```

This command will add the PPA to /etc/apt/sources.list.d and fetch the gpg key that the repository has been signed with. The GPG key ensures that the packages have not been tampered with since being built.

2. Notify your package manager of the new PPA by entering

```
sudo apt-get update
```

This command will retrieve the latest list of software from each repository that is enabled on your system, including the I2P PPA that was added with the earlier command.

3. You are now ready to install I2P!

```
sudo apt-get install i2p
```

4. Your browser should open up with your local I2P router console, to browse i2p domains you have to configure your browser to use the i2p proxy. Also check your connection status on the left side on the router console. If your status is **Network: Firewalled** your connection will be rather slow. The first time you start I2P it may take a few minutes to integrate you into the network and find additional peers to optimize your integration, so please be patient.

From the Tools menu, select Options to bring up the Firefox settings panel. Click the icon labelled Advanced, then click on the Network tab. In the Connections section, click on the Settings button. You'll see a Window like the following:

In the Connection Settings window, click the circle next to Manual proxy configuration, then enter 127.0.0.1, port 4444 in the HTTP Proxy field. Enter 127.0.0.1, port 4445 in the SSL Proxy field. Be sure to enter localhost and 127.0.0.1 into the "No Proxy for" box.

For more information and proxy settings for other browsers check <https://www.i2p2.de/htproxyports.htm>

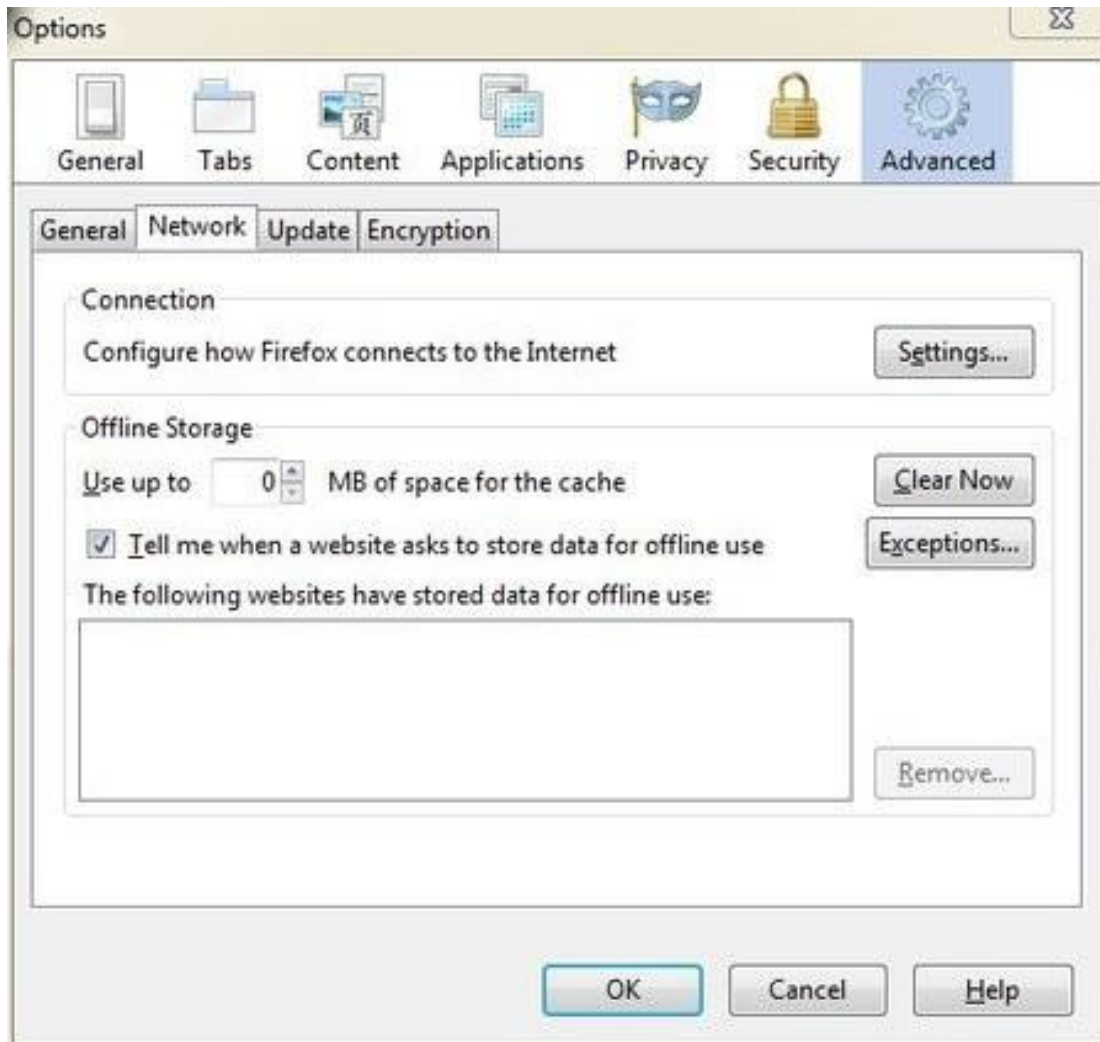


Figure 14.1:I2P

14.1 Installing I2P on Ubuntu Lucid Lynx (and newer) and derivatives like Linux Mint & Trisquel

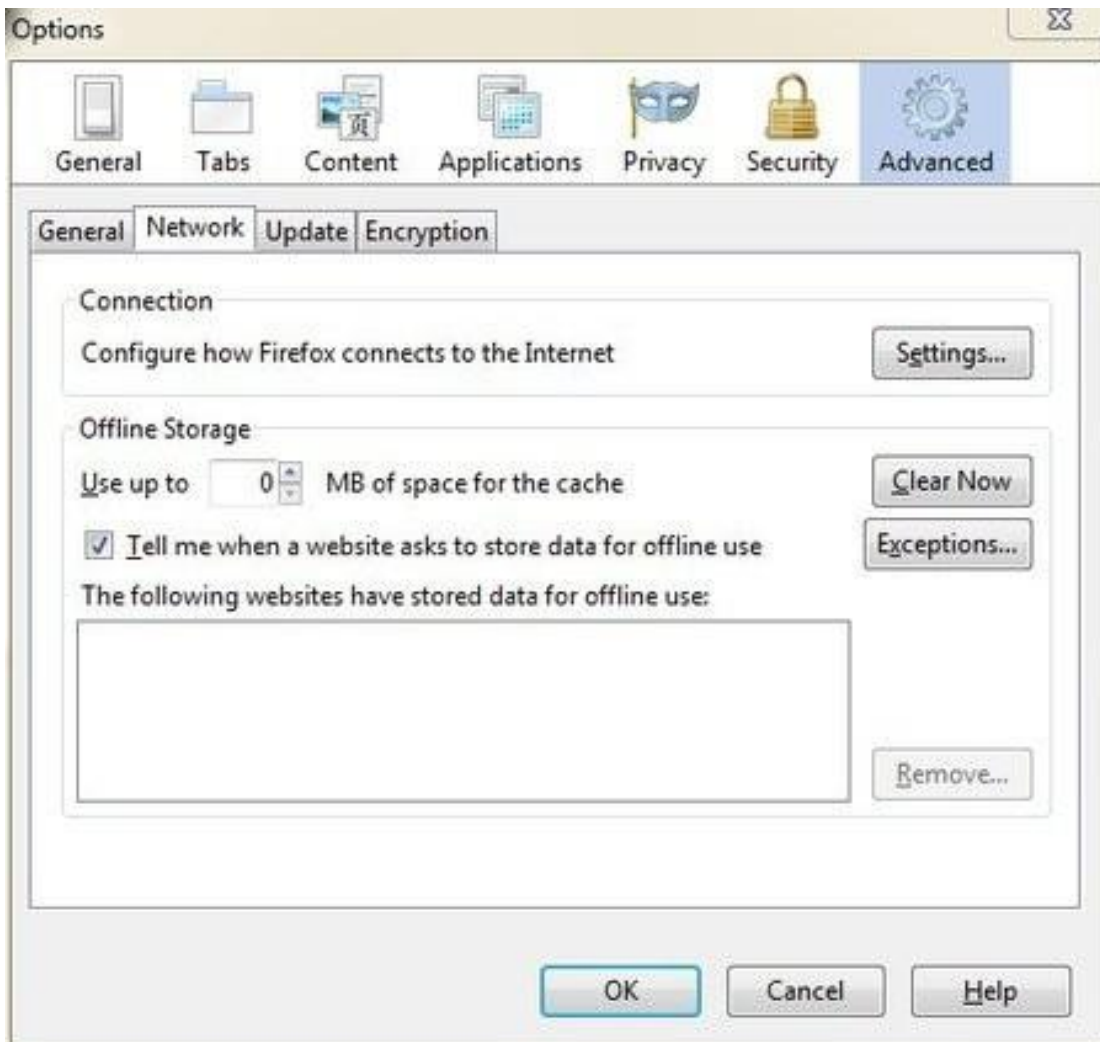


Figure 14.2:I2P

14.2 Instructions for Debian Lenny and newer

For more information visit this page <https://www.i2p2.de/debian.html>

14.3 Starting I2P

Using these I2P packages the I2P router can be started in the following three ways:

- “on demand” using the i2prouter script. Simply run “i2prouter start” from a command

- prompt. (Note: Do not use sudo or run it as root!).
- as a service that automatically runs when your system boots, even before logging in. The service can be enabled with “dpkg-reconfigure i2p” as root or using sudo. This is the recommended means of operation.

14.4 Anonymous Bittorrent with I2PSnark

We can use the I2P network to share and download files without the entire world knowing who is sharing them. or even the fact that you are running a torrent client, since i2p network is end-to-end encrypted the only thing known to outsider is you are running I2P. I2p come with a built-in torrent client that run inside the browser called I2PSnark.

You can access it through this direct link:

<http://localhost:7657/i2psnark/>

or through the router console: <http://localhost:7657/> and clicking on the torrent icon.

Once started you should see a screen similar to the following:



Figure 14.3:I2P

You can search for a torrent using one of following bittorrent trackers:

- <http://tracker.postman.i2p/>
- <http://diftracker.i2p/>

14.4 Anonymous Bittorrent with I2PSnark

Copy the torrent or magnet link and past it in the I2PSnark window, and click **Add torrent**. the file will be downloaded inside the **/home/user/.i2p/i2psnark** folder.

NOTE:

- Since I2P is a closed network, you can't download normal torrents found on regular internet with it, and it can't be used to make downloading them anonymous!
- The speed seems to be slightly lower than usual which is caused by the anonymization. I think that the download rates are still acceptable if you consider that you download and share anonymously.

15 Appendices

Adam Hyde Ahmed Mansour Alice Miller A Ravi Ariel Viera Asher Wolf AT Austin Martin Ben Weissmann Bernd Fix Brendan Howell Brian Newbold Carola Hesse Chris Pinchen Dan Hassan Daniel Kinsman Danja Vasiliev Dániel Nađandor djmat-tyg007 Douwe Schmidt Edward Cherlin Elemar Emile Denichaud Emile den Tex Erik Stein Erinn Clark Freddy Martinez Freerk Ohling Greg Broiles Haneef Mubarak Helen Varley Jamieson Janet Swisher Jan Gerber Jannette Mensch Jens Kubieziel Jmorahan Josh Datko Joshua Datko Julian Oliver Kai Engert Karen Reilly 13lackEyedAngels leoj3n Liam O Lonneke van der Velden Malte Malte Dik Marta Peirano Mart van Santen mdimitrova Michael Henriksen Nart Villeneuve Nathan Andrew Fain Nathan Houle Niels Elgaard Larsen Petter Ericson Piers Plato Punkbob Roberto Rastapopoulos Ronald Deibert Ross Anderson Sacha van Geffen Sam Tennyson Samuel Carlisle Samuel L. Tennyson Seth Schoen Steven Murdoch Stooj Story89 Ted W Ted Wood Teresa Dillon therealplato Tomas Krag Tom Boyle Travis Tueffel Uwe Lippmann WillMorrison Ximin Luo Yuval Adam zandi Zorrino Zorrinno

15.1 *Cryptography and Encryption*

Cryptography and encryption are similar terms, the former being the science and latter the implementation of it. The history of the subject can be traced back to ancient civilisations, when the first humans began to organise themselves into groups. This was driven in part by the realisation that we were in competition for resources and tribal organisation, warfare and so forth were necessary, so as to keep on top of the heap. In this respect cryptography and encryption are rooted in warfare, progression and resource management, where it was necessary to send secret messages to each other without the enemy deciphering ones moves.

Writing is actually one of the earliest forms of cryptography as not everyone could read. The word cryptography stems from the Greek words *kryptos* (hidden) and *graphein* (writing). In this respect cryptography and encryption in their simplest form refer to the writing of hidden messages, which require a system or rule to decode and read them. Essentially this enables you to protect your privacy by scrambling information in a way that it is only recoverable with certain knowledge (passwords or passphrases) or possession (a key).

Put in another way, encryption is the translation of information written in plaintext into a non-readable form (ciphertext) using algorithmic schemes (ciphers). The goal is to use the right key to unlock the ciphertext and return it back into its original plaintext form so it becomes readable again.

Although most encryption methods refer to written word, during World War Two, the US military used Navajo Indians, who traveled between camps sending messages in their native tongue. The reason the army used the Navajo tribe was to protect the information they were sending from the Japanese troops, who famously could not decipher the Navajo's spoken language. This is a very simple example of using a language to send messages that you do not want people to listen into or know what you're discussing. Why is encryption important? _____

Computer and telecommunication networks store digital echoes or footprints of our thoughts and records of personal lives.

From banking, to booking, to socialising: we submit a variety of detailed, personalised information, which is driving new modes of business, social interaction and behavior. We have now

become accustomed to giving away what was (and still is) considered private information in exchange for what is presented as more personalised and tailored services, which might meet our needs, but cater to our greed.

But how do we protect who sees, controls and uses this information?

Lets consider a scenario whereby we all thought it was fine to send all our communication on open handwritten postcards. From conversations with your doctor, to intimate moments with our lovers, to legal discussions you may have with lawyers or accountants. It's unlikely that we would want all people to be able to read such communications. So instead we have written letters in sealed envelopes, tracking methods for sending post, closed offices and confidential agreements, which help to keep such communication private. However given the shift in how we communicate, much more of this type of interaction is taking place online. More importantly it is taking place through online spaces, which are not private by default and open to people with little technical skills to snoop into the matters that can mean the most to our lives.

Online privacy and encryption is something we therefore need to be aware of and practice daily. In the same way we would put an important letter into an envelope or have a conversation behind a closed door. Given that so much of our private communication is now happening in networked and online spaces, we should consider the interface, like envelopes or seals, which protect this material as a basic necessity and human right.

15.1.1 Encryption examples

Throughout history we can find examples of cipher methods, which have been used to keep messages private and secret.

15.1.2 A Warning!

“There are two kinds of cryptography in this world: cryptography that will stop your kid sister from reading your files, and cryptography that will stop major governments from reading your files” - Bruce Schneier, *Applied Cryptography*, 1996

This chapter first explains a number of historical cryptographic systems and then provides a summary of modern techniques. The historical examples illustrate how cryp-

tography emerged, but are considered broken in the face of modern computers. They can be fun to learn, but please don't use them for anything sensitive!

15.1.3 Historical ciphers

Classical ciphers refer to historical ciphers, which are now out of popular use or no longer applicable. There are two general categories of classical ciphers: transposition and substitution ciphers.

In a transposition cipher, the letters themselves are kept unchanged, but the order within the message is scrambled according to some well-defined scheme. An example of a transposition cipher is Skytale, which was used in ancient Rome and Greece. A paperstrip was wrapped around a stick and the message written across it. That way the message could not be read unless wound around a stick of similar diameter again.



Figure 15.1:Cryptography

Image: Skytale taken from Wikimedia Commons (3.10.12)

A substitution cipher is a form of classical cipher whereby letters or groups of letters are systematically replaced throughout the message for other letters (or groups of letters). Substitution ciphers are divided into monoalphabetic and polyalphabetic substitutions. The Caesar Shift cipher is common example of amonoalphabetic substitution ciphers, where the letters in the alphabet are shifted in one direction or another.

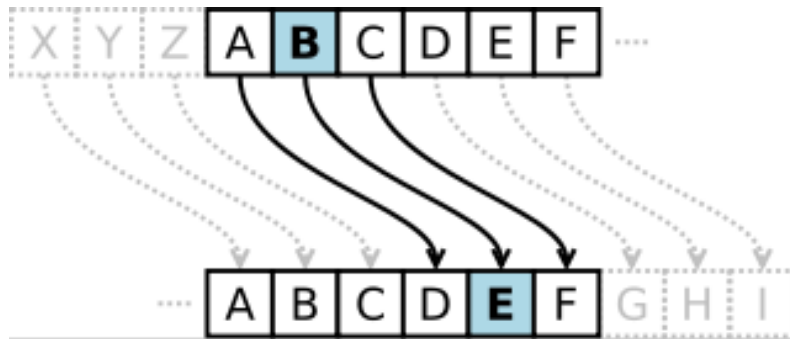


Figure 15.2:Cryptography

Image: Caesar Shift Cipher taken from Wikimedia Commons (3.10.12)

Polyalphabetic substitutions are more complex than substitution ciphers as they use more than one alphabet and rotate them. For example, The Alberti cipher, which was the

first polyalphabetic cipher was created by Leon Battista Alberti, a 15th century Italian, Renaissance polymath and humanist who is also credited as the godfather of western cryptography. His cipher is similar to the Vigenère cipher, where every letter of the alphabet gets a unique number (e.g. 1-26). The message is then encrypted by writing down the message along with the password repeatedly written beneath it.

In the Vigenère cipher the corresponding numbers of the letters of message and key are summed up (with numbers exceeding the alphabet being dragged around the back) making the message so unreadable that it couldn't be deciphered for centuries (nowadays, with the help of computers, this obviously isn't true anymore).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Figure 15.3:Cryptography

Image: Vigenère cipher taken from Wikimedia Commons (3.10.12)

During World War 2 there was a surge in cryptography, which led to the development of new algorithms such as the one-time pad (OTP). The OTP algorithm combines plaintext with a random key that is as long as the plaintext so that each character is only used once. To use it you need two copies of the pad, which are kept by each user and exchanged via a secure channel. Once the message is encoded with the pad, the pad is destroyed and the encoded message is sent. On the recipient's side, the encoded message has a duplicate copy of the pad from which the plaintext message is generated. A good way to look at OTP is to think of it as a 100% noise source, which is used to mask the message. Since both parties of the communication have copies of the noise source they are the only people who can filter it out.

OTP lies behind modern day stream ciphers, which are explained below. Claude Shannon, (a key player in modern cryptography and information theory), in his seminal 1949 paper "Communication Theory of Secrecy Systems" demonstrated that theoretically all unbreakable ciphers should include the OTP encryption, which if used correctly are impossible to crack.

15.1.4 Modern ciphers

Post the World Wars the field of cryptography became less of a public service and fell more within the domain of governance. Major advances in the field began to reemerge in the mid-1970s with the advent of personalised computers and the introduction of the Data Encryption Standard (DES, developed at IBM in 1977 and later adopted by the U.S government). Since 2001 we now use the AES, (Advanced Encryption Standard), which is based on symmetric cryptography forms.

Contemporary cryptography can be generally divided into what is called symmetric, asymmetric and quantum cryptography.

Symmetric cryptography, or secret key, cryptography refers to ciphers where the same key is used to both encrypt and decrypt the text or information involved. In this class of ciphers the key is shared and kept secret within a restricted group and therefore it is not possible to view the encrypted information without having the key. A simple analogy to secret key cryptography is having access to a community garden, which has one key to open gate, which is shared by the community. You cannot open the gate, unless you have the key. Obviously the issue here with the garden key and with symmetric cryptography is if the key falls into the wrong hands, then an intruder or attacker can get in and the security of the garden, or the data or information is compromised. Consequently one of the main issues with this form of cryptography is the issue of key management. As a result this method is best employed within single-user contexts or small group environments.

Despite this limitation symmetric key methods are considerably faster than asymmetric methods and so are the preferred mechanism for encrypting large chunks of text.

Symmetric ciphers are usually implemented using **block ciphers** or **stream ciphers**.

Block ciphers work by looking at the input data in 8 or 16 or 32 byte blocks at a time and spreading the input and key within those blocks. Different modes of operation are performed on the data in order to transform and spread the data between blocks. Such

ciphers use a secret key to convert a fixed block of plain text into cipher text. The same key is then used to decrypt the cipher text.

In comparison stream ciphers (also known as state cipher) work on each plaintext digit by creating a corresponding keystream which forms the ciphertext. The keystream refers to a stream of random characters (bits, bytes, numbers or letters) on which various additive or subtractive functions are performed and combined to a character in the plaintext message, which then produces the ciphertext. Although this method is very secure, it is not always practical, since the key of the same length as the message needs to be transmitted in some secure way so that receiver can decipher the message. Another limitation is that the key can only be used once and then its discarded. Although this can mean almost watertight security, it does limit the use of the cipher.

Asymmetric ciphers work much more complex mathematical problems with back doors, enabling faster solutions on smaller, highly important pieces of data. They also work on fixed data sizes, typically 1024-2048 bits and and 384 bits. What makes them special is that they help solve some of the issues with key distribution by allocating one public and one private pair per person, so that everyone just needs to know everyone else's public portion. Asymmetric ciphers are also used for digital signatures. Where as symmetric ciphers are generally used for message authenticity. Symmetric ciphers cannot non-repudiation signatures (i.e., signatures that you cannot later deny that you did not sign). Digital signatures are very important in modern day cryptography. They are similar to wax seals in that they verify who the message is from and like seals are unique to that person. Digital signatures are one of the methods used within public

key systems, which have transformed the field of cryptography are central to modern day Internet security and online transactions.

15.1.5 Quantum Cryptography

Quantum cryptography is the term used to describe the type of cryptography that is now necessary to deal with the speed at which we now process information and the related security measures that are necessary. Essentially it deals with how we use quantum communication to securely exchange a key and its associated distribution. As the machines we use become faster the possible combinations of public-key encryption and digital signatures becomes easier to break and quantum cryptography deals with the types of algorithms that are necessary to keep pace with more advanced networks.

15.1.6 Challenges & Implications

At the heart of cryptography lies the challenge of how we use and communicate information. The above methods describe how we encrypt written communication but obviously as shown in the Navajo example other forms of communication (speech, sound, image etc) can also be encrypted using different methods.

The main goal and skill of encryption is to apply the right methods to support trustworthy communication. This is achieved by understanding the tradeoffs, strengths and weaknesses of different cipher methods and how they relate to the level of security and

privacy required. Getting this right depends on the task and context.

Importantly when we speak about communication, we are speaking about trust. Traditionally cryptography dealt with the hypothetical scenarios, where the challenge was to address how 'Bob' could speak to 'Alice' in a private and secure manner.

Our lives are now heavily mediated via computers and the Internet. So the boundaries between Bob, Alice + the 'other' (Eve, Oscar, Big Brother, your boss, ex-boyfriend or the government) are a lot more blurred. Given the quantum leaps in computer processing, in order for 'us', Bob's and Alice's to have trust in the system, we need to know who we are talking too, we need to know who is listening and importantly who has the potential to eavesdrop. What becomes important is how we navigate this complexity and feel in control and secure, so that you can engage and communicate in a trustful manner, which respects our individual freedoms and privacy.

15.2 Glossary

Much of this content is based on <http://en.cship.org/wiki/Special:Allpages>

15.2.1 aggregator

An aggregator is a service that gathers syndicated information from one or many sites and makes it available at a different address. Sometimes called an RSS aggregator, a feed aggregator, a feed reader, or a news reader. (Not to be confused with a Usenet News reader.)

15.2.2 anonymity

(Not be confused with privacy, pseudonymity, security, or confidentiality.)

Anonymity on the Internet is the ability to use services without leaving clues to one's identity or being spied upon. The level of protection depends on the anonymity techniques used and the extent of monitoring. The strongest techniques in use to protect anonymity involve creating a chain of communication using a random process to select some of the links, in which each link has access to only partial information about the process. The first knows the user's Internet address (IP) but not the content, destination, or purpose of the communication, because the message contents and destination information are encrypted. The last knows the identity of the site being contacted, but not the source of the session. One or more steps in between prevents the first and last links from sharing their partial knowledge in order to connect the user and the target site.

15.2.3 anonymous remailer

An anonymous remailer is a service that accepts e-mail messages containing instructions for delivery, and sends them out without revealing their sources. Since the remailer has access to the user's address, the content of the message, and the destination of the

message, remailers should be used as part of a chain of multiple remailers so that no one remailer knows all this information.

15.2.4 ASP (application service provider)

An ASP is an organization that offers software services over the Internet, allowing the software to be upgraded and maintained centrally.

15.2.5 backbone

A backbone is one of the high-bandwidth communications links that tie together networks in different countries and organizations around the world to form the Internet.

15.2.6 badware

See malware.

15.2.7 bandwidth

The bandwidth of a connection is the maximum rate of data transfer on that connection, limited by its capacity and the capabilities of the computers at both ends of the connection.

15.2.8 bash (Bourne-again shell)

The bash shell is a command-line interface for Linux/Unix operating systems, based on the Bourne shell.

15.2.9 BitTorrent

BitTorrent is a peer-to-peer file-sharing protocol invented by Bram Cohen in 2001. It allows individuals to cheaply and effectively distribute large files, such as CD images, video, or music files.

15.2.10 blacklist

A blacklist is a list of forbidden things. In Internet censorship, lists of forbidden Web sites or the IP addresses of computers may be used as blacklists; censorware may allow access to all sites except for those specifically listed on its blacklist. An alternative to a blacklist is a whitelist, or a list of permitted things. A whitelist system blocks access to all sites except for those specifically listed on the whitelist. This is a less common approach to Internet censorship. It is possible to combine both approaches, using string matching or other conditional techniques on URLs that do not match either list.

15.2.11 bluebar

The blue URL bar (called the Bluebar in Psiphon lingo) is the form at the top of your Psiphon node browser window, which allows you to access blocked site by typing its URL inside.

See also Psiphon node

15.2.12 block

To block is to prevent access to an Internet resource, using any number of methods.

15.2.13 bookmark

A bookmark is a placeholder within software that contains a reference to an external resource. In a browser, a bookmark is a reference to a Web page – by choosing the bookmark you can quickly load the Web site without needing to type in the full URL.

15.2.14 bridge

See Tor bridge.

15.2.15 brute-force attack

A brute force attack consists of trying every possible code, combination, or password until you find the right one. These are some of the most trivial hacking attacks.

15.2.16 cache

A cache is a part of an information-processing system used to store recently used or frequently used data to speed up repeated access to it. A Web cache holds copies of Web page files.

15.2.17 censor

To censor is to prevent publication or retrieval of information, or take action, legal or otherwise, against publishers and readers.

15.2.18 censorware

Censorware is software used to filter or block access to the Internet. This term is most often used to refer to Internet filtering or blocking software installed on the client machine (the PC which is used to access the Internet). Most such client-side censorware is used for parental control purposes.

Sometimes the term censorware is also used to refer to software used for the same purpose installed on a network server or router.

15.2.19 CGI (Common Gateway Interface)

CGI is a common standard used to let programs on a Web server run as Web applications. Many Web-based proxies use CGI and thus are also called “CGI proxies”. (One popular CGI proxy application written by James Marshall using the Perl programming language is called CGIProxy.)

15.2.20 chat

Chat, also called instant messaging, is a common method of communication among two or more people in which each line typed by a participant in a session is echoed to all of the others. There are numerous chat protocols, including those created by specific companies (AOL, Yahoo!, Microsoft, Google, and others) and publicly defined protocols. Some chat client software uses only one of these protocols, while others use a range of popular protocols.

15.2.21 cipher

In cryptography, a cipher (or cypher) is an algorithm for performing encryption or decryption.

15.2.22 circumvention

Circumvention is publishing or accessing content in spite of attempts at censorship.

15.2.23 Common Gateway Interface

See CGI.

15.2.24 command-line interface

A method of controlling the execution of software using commands entered on a keyboard, such as a Unix shell or the Windows command line.

15.2.25 cookie

A cookie is a text string sent by a Web server to the user’s browser to store on the user’s computer, containing information needed to maintain continuity in sessions across multiple Web pages, or across multiple sessions. Some Web sites cannot be used without accepting and storing a cookie. Some people consider this an invasion of privacy or a security risk.

15.2.26 country code top-level domain (ccTLD)

Each country has a two-letter country code, and a TLD (top-level domain) based on it, such as .ca for Canada; this domain is called a country code top-level domain. Each

such ccTLD has a DNS server that lists all second-level domains within the TLD. The Internet root servers point to all TLDs, and cache frequently-used information on lower-level domains.

15.2.27 cryptography

Cryptography is the practice and study of techniques for secure communication in the presence of third parties (called adversaries). More generally, it is about constructing and analyzing protocols that overcome the influence of adversaries and which are related to various aspects in information security such as data confidentiality, data integrity, authentication, and non-repudiation. Modern cryptography intersects the disciplines of mathematics, computer science, and electrical engineering. Applications of cryptography include ATM cards, computer passwords, and electronic commerce.

15.2.28 DARPA (Defense Advanced Projects Research Agency)

DARPA is the successor to ARPA, which funded the Internet and its predecessor, the ARPAnet.

15.2.29 decryption

Decryption is recovering plain text or other messages from encrypted data with the use of a key. See also encryption.

15.2.30 disk encryption

Disk encryption is a technology which protects information by converting it into unreadable code that cannot be deciphered easily by unauthorized people. Disk encryption uses disk encryption software or hardware to encrypt every bit of data that goes on a disk or disk volume. Disk encryption prevents unauthorized access to data storage.

15.2.31 domain

A domain can be a Top-Level Domain (TLD) or secondary domain on the Internet.

See also Top-Level Domain, country code Top-Level Domain and secondary domain.

15.2.32 DNS (Domain Name System)

The Domain Name System (DNS) converts domain names, made up of easy-to-remember combinations of letters, to IP addresses, which are hard-to-remember strings of numbers. Every computer on the Internet has a unique address (a little bit like an area code+telephone number).

15.2.33 DNS leak

A DNS leak occurs when a computer configured to use a proxy for its Internet connection nonetheless makes DNS queries without using the proxy, thus exposing the user's attempts to connect with blocked sites. Some Web browsers have configuration options to force the use of the proxy.

15.2.34 DNS server

A DNS server, or name server, is a server that provides the look-up function of the Domain Name System. It does this either by accessing an existing cached record of the IP address of a specific domain, or by sending a request for information to another name server.

15.2.35 DNS tunnel

A DNS tunnel is a way to tunnel almost everything over DNS/Nameservers.

Because you “abuse” the DNS system for an unintended purpose, it only allows a very slow connection of about 3 kb/s which is even less than the speed of an analog modem. That is not enough for YouTube or file sharing, but should be sufficient for instant messengers like ICQ or MSN Messenger and also for plain text e-mail.

On the connection you want to use a DNS tunnel, you only need port 53 to be open; therefore it even works on many commercial Wi-Fi providers without the need to pay.

The main problem is that there are no public modified nameservers that you can use. You have to set up your own. You need a server with a permanent connection to the Internet running Linux. There you can install the free software OzymanDNS and in combination with SSH and a proxy like Squid you can use the tunnel. More Information on this on <http://www.dnstunnel.de>.

15.2.36 Eavesdropping

Eavesdropping is listening to voice traffic or reading or filtering data traffic on a telephone line or digital data connection, usually to detect or prevent illegal or unwanted activities or to control or monitor what people are talking about.

15.2.37 e-mail

E-mail, short for electronic mail, is a method to send and receive messages over the Internet. It is possible to use a Web mail service or to send e-mails with the SMTP protocol and receive them with the POP3 protocol by using an e-mail client such as Outlook Express or Thunderbird. It is comparatively rare for a government to block e-mail, but e-mail surveillance is common. If e-mail is not encrypted, it could be read easily by a network operator or government.

15.2.38 embedded script

An embedded script is a piece of software code.

15.2.39 encryption

Encryption is any method for recoding and scrambling data or transforming it mathematically to make it unreadable to a third party who doesn't know the secret key to decrypt it. It is possible to encrypt data on your local hard drive using software like TrueCrypt (<http://www.truecrypt.org>) or to encrypt Internet traffic with TLS/SSL or SSH.

See also decryption.

15.2.40 exit node

An exit node is a Tor node that forwards data outside the Tor network.

See also middleman node.

15.2.41 file sharing

File sharing refers to any computer system where multiple people can use the same information, but often refers to making music, films or other materials available to others free of charge over the Internet.

15.2.42 file spreading engine

A file spreading engine is a Web site a publisher can use to get around censorship. A user only has to upload a file to publish once and the file spreading engine uploads that file to some set of sharehosting services (like Rapidshare or Megaupload).

15.2.43 filter

To filter is to search in various ways for specific data patterns to block or permit communications.

15.2.44 Firefox

Firefox is the most popular free and open source Web browser, developed by the Mozilla Foundation.

15.2.45 forum

On a Web site, a forum is a place for discussion, where users can post messages and comment on previously posted messages. It is distinguished from a mailing list or a Usenet newsgroup by the persistence of the pages containing the message threads. Newsgroup and mailing list archives, in contrast, typically display messages one per page, with navigation pages listing only the headers of the messages in a thread.

15.2.46 frame

A frame is a portion of a Web page with its own separate URL. For example, frames are frequently used to place a static menu next to a scrolling text window.

15.2.47 FTP (File Transfer Protocol)

The FTP protocol is used for file transfers. Many people use it mostly for downloads; it can also be used to upload Web pages and scripts to some Web servers. It normally uses ports 20 and 21, which are sometimes blocked. Some FTP servers listen to an uncommon port, which can evade port-based blocking.

A popular free and open source FTP client for Windows and Mac OS is FileZilla. There are also some Web-based FTP clients that you can use with a normal Web browser like Firefox.

15.2.48 *full disk encryption*

see disk encryption.

15.2.49 *gateway*

A gateway is a node connecting two networks on the Internet. An important example is a national gateway that requires all incoming or outgoing traffic to go through it.

15.2.50 *GNU Privacy Guard*

GNU Privacy Guard (GnuPG or GPG) is a GPL Licensed alternative to the PGP suite of cryptographic software. GnuPG is compliant with RFC 4880, which is the current IETF standards track specification of OpenPGP.

see also Pretty Good Privacy (PGP).

15.2.51 *GPG*

see GNU Privacy Guard.

15.2.52 *honeypot*

A honeypot is a site that pretends to offer a service in order to entice potential users to use it, and to capture information about them or their activities.

15.2.53 *hop*

A hop is a link in a chain of packet transfers from one computer to another, or any computer along the route. The number of hops between computers can give a rough measure of the delay (latency) in communications between them. Each individual hop is also an entity that has the ability to eavesdrop on, block, or tamper with communications.

15.2.54 *HTTP (Hypertext Transfer Protocol)*

HTTP is the fundamental protocol of the World Wide Web, providing methods for requesting and serving Web pages, querying and generating answers to queries, and accessing a wide range of services.

15.2.55 *HTTPS (Secure HTTP)*

Secure HTTP is a protocol for secure communication using encrypted HTTP messages. Messages between client and server are encrypted in both directions, using keys generated when the connection is requested and exchanged securely. Source and destination IP addresses are in the headers of every packet, so HTTPS cannot hide the fact of the communication, just the contents of the data transmitted and received.

15.2.56 *IANA (Internet Assigned Numbers Authority)*

IANA is the organization responsible for technical work in managing the infrastructure of the Internet, including assigning blocks of IP addresses for top-level domains and licensing domain

registrars for ccTLDs and for the generic TLDs, running the root name servers of the Internet, and other duties.

15.2.57 ICANN (Internet Corporation for Assigned Names and Numbers)

ICANN is a corporation created by the US Department of Commerce to manage the highest levels of the Internet. Its technical work is performed by IANA.

15.2.58 Instant Messaging (IM)

Instant messaging is either certain proprietary forms of chat using proprietary protocols, or chat in general. Common instant messaging clients include MSN Messenger, ICQ, AIM or Yahoo! Messenger.

15.2.59 Intermediary

See man in the middle.

15.2.60 Internet

The Internet is a network of networks interconnected using TCP/IP and other communication protocols.

15.2.61 IP (Internet Protocol) Address

An IP address is a number identifying a particular computer on the Internet. In the previous version 4 of the Internet Protocol an IP address consisted of four bytes (32 bits), often represented as four integers in the range 0-255 separated by dots, such as

74.54.30.85. In IPv6, which the Net is currently switching to, an IP address is four times longer, and consists of 16 bytes (128 bits). It can be written as 8 groups of 4 hex digits separated by colons, such as 2001:0db8:85a3:0000:0000:8a2e:0370:7334.

15.2.62 IRC (Internet relay chat)

IRC is a more than 20-year-old Internet protocol used for real-time text conversations (chat or instant messaging). There exist several IRC networks – the largest have more than 50 000 users.

15.2.63 ISP (Internet Service Provider)

An ISP (Internet service provider) is a business or organization that provides access to the Internet for its customers.

15.2.64 JavaScript

JavaScript is a scripting language, commonly used in Web pages to provide interactive functions.

15.2.65 KeePass, KeePassX

KeePass and KeePassX are types of Password Manager.

15.2.66 keychain software

see Password Manager.

15.2.67 keyword filter

A keyword filter scans all Internet traffic going through a server for forbidden words or terms to block.

15.2.68 latency

Latency is a measure of time delay experienced in a system, here in a computer network. It is measured by the time between the start of packet transmission to the start of packet reception, between one network end (e.g. you) to the other end (e.g. the Web server). One very powerful way of Web filtering is maintaining a very high latency, which makes lots of circumvention tools very difficult to use.

15.2.69 log file

A log file is a file that records a sequence of messages from a software process, which can be an application or a component of the operating system. For example, Web servers or proxies may keep log files containing records about which IP addresses used these services when and what pages were accessed.

15.2.70 low-bandwidth filter

A low-bandwidth filter is a Web service that removes extraneous elements such as advertising and images from a Web page and otherwise compresses it, making page download much quicker.

15.2.71 malware

Malware is a general term for malicious software, including viruses, that may be installed or executed without your knowledge. Malware may take control of your computer for purposes such as sending spam. (Malware is also sometimes called badware.)

15.2.72 man in the middle

A man in the middle or man-in-the-middle is a person or computer capturing traffic on a communication channel, especially to selectively change or block content in a way that undermines cryptographic security. Generally the man-in-the-middle attack involves impersonating a Web site, service, or individual in order to record or alter communications. Governments can run man-in-the-middle attacks at country gateways where all traffic entering or leaving the country must pass.

15.2.73 middleman node

A middleman node is a Tor node that is not an exit node. Running a middleman node can be safer than running an exit node because a middleman node will not show up in third parties' log files. (A middleman node is sometimes called a non-exit node.)

15.2.74 monitor

To monitor is to check a data stream continuously for unwanted activity.

15.2.75 network address translation (NAT)

NAT is a router function for hiding an address space by remapping. All traffic going out from the router then uses the router's IP address, and the router knows how to route incoming traffic to the requestor. NAT is frequently implemented by firewalls. Because incoming connections are normally forbidden by NAT, NAT makes it difficult to offer a service to the general public, such as a Web site or public proxy. On a network where NAT is in use, offering such a service requires some kind of firewall configuration or NAT traversal method.

15.2.76 network operator

A network operator is a person or organization who runs or controls a network and thus is in a position to monitor, block, or alter communications passing through that network.

15.2.77 node

A node is an active device on a network. A router is an example of a node. In the Psiphon and Tor networks, a server is referred to as a node.

15.2.78 non-exit node

See middleman node.

15.2.79 obfuscation

Obfuscation means obscuring text using easily-understood and easily-reversed transformation techniques that will withstand casual inspection but not cryptanalysis, or making minor changes in text strings to prevent simple matches. Web proxies often use obfuscation to hide certain names and addresses from simple text filters that might be fooled by the obfuscation. As another example, any domain name can optionally contain a final dot, as in "somewhere.com.", but some filters might search only for "somewhere.com" (without the final dot).

15.2.80 open node

An open node is a specific Psiphon node which can be used without logging in. It automatically loads a particular homepage, and presents itself in a particular language, but can then be used to browse elsewhere.

See also Psiphon node.

15.2.81 OTR/Off-the-Record messaging

Off-the-Record Messaging, commonly referred to as OTR, is a cryptographic protocol that provides strong encryption for instant messaging conversations.

15.2.82 packet

A packet is a data structure defined by a communication protocol to contain specific information in specific forms, together with arbitrary data to be communicated from one point to another. Messages are broken into pieces that will fit in a packet for transmission, and reassembled at the other end of the link.

15.2.83 password manager

A password manager is software that helps a user organize passwords and PIN codes. The software typically has a local database or a file that holds the encrypted password data for secure logon onto computers, networks, web sites and application data files. KeePass <http://keepass.info/> is an example of a password manager.

15.2.84 pastebin

A web service where any kind of text can be dumped and read without registration. All text will be visible publicly.

15.2.85 peer-to-peer

A peer-to-peer (or P2P) network is a computer network between equal peers. Unlike client-server networks there is no central server and so the traffic is distributed only among the clients. This technology is mostly applied to file sharing programs like BitTorrent, eMule and Gnutella. But also the very old Usenet technology or the VoIP program Skype can be categorized as peer-to-peer systems.

See also file sharing.

15.2.86 perfect forward secrecy

In an authenticated key-agreement protocol that uses public key cryptography, perfect forward secrecy (or PFS) is the property that ensures that a session key derived from a set of long-term public and private keys will not be compromised if one of the (long-term) private keys is compromised in the future.

15.2.87 Pretty Good Privacy (PGP)

Pretty Good Privacy (PGP) is a data encryption and decryption computer program that provides cryptographic privacy and authentication for data communication. PGP is often used for signing, encrypting and decrypting texts, e-mails, files, directories and whole disk partitions to increase the security of e-mail communications.

PGP and similar products follow the OpenPGP standard (RFC 4880) for encrypting and decrypting data.

15.2.88 PHP

PHP is a scripting language designed to create dynamic Web sites and web applications. It is installed on a Web server. For example, the popular Web proxy PHPProxy uses this technology.

15.2.89 plain text

Plain text is unformatted text consisting of a sequence of character codes, as in ASCII plain text or Unicode plain text.

15.2.90 plaintext

Plaintext is unencrypted text, or decrypted text.

See also encryption, TLS/SSL, SSH.

15.2.91 privacy

Protection of personal privacy means preventing disclosure of personal information without the permission of the person concerned. In the context of circumvention, it means preventing observers from finding out that a person has sought or received information that has been blocked or is illegal in the country where that person is at the time.

15.2.92 private key

see public key encryption/public-key cryptography.

15.2.93 POP3

Post Office Protocol version 3 is used to receive mail from a server, by default on port 110 with an e-mail program such as Outlook Express or Thunderbird.

15.2.94 port

A hardware port on a computer is a physical connector for a specific purpose, using a particular hardware protocol. Examples are a VGA display port or a USB connector.

Software ports also connect computers and other devices over networks using various protocols, but they exist in software only as numbers. Ports are somewhat like numbered doors into different rooms, each for a special service on a server or PC. They are identified by numbers from 0 to 65535.

15.2.95 protocol

A formal definition of a method of communication, and the form of data to be transmitted to accomplish it. Also, the purpose of such a method of communication. For example, Internet Protocol (IP) for transmitting data packets on the Internet, or Hypertext Transfer Protocol for interactions on the World Wide Web.

15.2.96 proxy server

A proxy server is a server, a computer system or an application program which acts as a gateway between a client and a Web server. A client connects to the proxy server to request a Web page from a different server. Then the proxy server accesses the resource by connecting to the specified server, and returns the information to the requesting site. Proxy servers can serve many different purposes, including restricting Web access or helping users route around obstacles.

15.2.97 Psiphon node

A Psiphon node is a secured web proxy designed to evade Internet censorship. It is developed by Psiphon inc. Psiphon nodes can be open or private.

15.2.98 private node

A private node is a Psiphon node working with authentication, which means that you have to register before you can use it. Once registered, you will be able to send invitations to your friends and relatives to use this specific node.

See also Psiphon node.

15.2.99 public key

see public key encryption/public-key cryptography.

15.2.100 public key encryption/public-key cryptography

Public-key cryptography refers to a cryptographic system requiring two separate keys, one of which is secret and one of which is public. Although different, the two parts of the key pair are mathematically linked. One key locks or encrypts the plaintext, and the other unlocks or decrypts the ciphertext. Neither key can perform both functions. One of these keys is published or public, while the other is kept private.

Public-key cryptography uses asymmetric key algorithms (such as RSA), and can also be referred to by the more generic term “asymmetric key cryptography.”

15.2.101 publicly routable IP address

Publicly routable IP addresses (sometimes called public IP addresses) are those reachable in the normal way on the Internet, through a chain of routers. Some IP addresses are private, such as the 192.168.x.x block, and many are unassigned.

15.2.102 regular expression

A regular expression (also called a regexp or RE) is a text pattern that specifies a set of text strings in a particular regular expression implementation such as the UNIX grep utility. A text string “matches” a regular expression if the string conforms to the pattern, as defined by the regular expression syntax. In each RE syntax, some characters have special meanings, to allow one pattern to match multiple other strings. For example, the regular expression lo+se matches lose, loose, and looose.

15.2.103 remailer

An anonymous remailer is a service which allows users to send e-mails anonymously. The remailer receives messages via e-mail and forwards them to their intended recipient after removing information that would identify the original sender. Some also provide an anonymous return

address that can be used to reply to the original sender without disclosing her identity. Well-known Remailer services include Cypherpunk, Mixmaster and Nym.

15.2.104 router

A router is a computer that determines the route for forwarding packets. It uses address information in the packet header and cached information on the server to match address numbers with hardware connections.

15.2.105 root name server

A root name server or root server is any of thirteen server clusters run by IANA to direct traffic to all of the TLDs, as the core of the DNS system.

15.2.106 RSS (Real Simple Syndication)

RSS is a method and protocol for allowing Internet users to subscribe to content from a Web page, and receive updates as soon as they are posted.

15.2.107 scheme

On the Web, a scheme is a mapping from a name to a protocol. Thus the HTTP scheme maps URLs that begin with HTTP: to the Hypertext Transfer Protocol. The protocol determines the interpretation of the rest of the URL, so that <http://www.example.com/dir/content.html> identifies a Web site and a specific file in a specific directory, and <mailto:user@somewhere.com> is an e-mail address of a specific person or group at a specific domain.

15.2.108 shell

A UNIX shell is the traditional command line user interface for the UNIX/Linux operating systems. The most common shells are sh and bash.

15.2.109 SOCKS

A SOCKS proxy is a special kind of proxy server. In the ISO/OSI model it operates between the application layer and the transport layer. The standard port for SOCKS proxies is 1080, but they can also run on different ports. Many programs support a connection through a SOCKS proxy. If not you can install a SOCKS client like FreeCap, ProxyCap or SocksCap which can force programs to run through the Socks proxy using dynamic port forwarding. It is also possible to use SSH tools such as OpenSSH as a SOCKS proxy server.

15.2.110 screenlogger

A screenlogger is software able to record everything your computer displays on the screen. The main feature of a screenlogger is to capture the screen and log it into files to view at any time in the future. Screen loggers can be used as powerful monitoring tool. You should be aware of any screen logger running on any computer you are using, anytime.

15.2.111 *script*

A script is a program, usually written in an interpreted, non-compiled language such as JavaScript, Java, or a command interpreter language such as bash. Many Web pages include scripts to manage user interaction with a Web page, so that the server does not have to send a new page for each change.

15.2.112 *smartphone*

A smartphone is a mobile phone that offers more advanced computing ability and connectivity than a contemporary feature phone, such as Web access, ability to run elaborated operating systems and run built-in applications.

15.2.113 *spam*

Spam is messages that overwhelm a communications channel used by people, most notably commercial advertising sent to large numbers of individuals or discussion groups. Most spam advertises products or services that are illegal in one or more ways, almost always including fraud. Content filtering of e-mail to block spam, with the permission of the recipient, is almost universally approved of.

15.2.114 *SSH (Secure Shell)*

SSH or Secure Shell is a network protocol that allows encrypted communication between computers. It was invented as a successor of the unencrypted Telnet protocol and is also used to access a shell on a remote server.

The standard SSH port is 22. It can be used to bypass Internet censorship with port forwarding or it can be used to tunnel other programs like VNC.

15.2.115 *SSL (Secure Sockets Layer)*

SSL (or Secure Sockets Layer), is one of several cryptographic standards used to make Internet transactions secure. It is used as the basis for the creation of the related Transport Layer Security (TLS). You can easily see if you are using SSL by looking at the URL in your Browser (like Firefox or Internet Explorer): If it starts with https instead of http, your connection is encrypted.

15.2.116 *steganography*

Steganography, from the Greek for hidden writing, refers to a variety of methods of sending hidden messages where not only the content of the message is hidden but the very fact that something covert is being sent is also concealed. Usually this is done by concealing something within something else, like a picture or a text about something innocent or completely unrelated. Unlike cryptography, where it is clear that a secret message is being transmitted, steganography does not attract attention to the fact that someone is trying to conceal or encrypt a message.

15.2.117 *subdomain*

A subdomain is part of a larger domain. If for example “wikipedia.org” is the domain for the Wikipedia, “en.wikipedia.org” is the subdomain for the English version of the Wikipedia.

15.2.118 threat analysis

A security threat analysis is properly a detailed, formal study of all known ways of attacking the security of servers or protocols, or of methods for using them for a particular purpose such as circumvention. Threats can be technical, such as code-breaking or exploiting software bugs, or social, such as stealing passwords or bribing someone who has special knowledge. Few companies or individuals have the knowledge and skill to do a comprehensive threat analysis, but everybody involved in circumvention has to make some estimate of the issues.

15.2.119 Top-Level Domain (TLD)

In Internet names, the TLD is the last component of the domain name. There are several generic TLDs, most notably .com, .org, .edu, .net, .gov, .mil, .int, and one two-letter country code (ccTLD) for each country in the system, such as .ca for Canada. The European Union also has the two-letter code .eu.

15.2.120 TLS (Transport Layer Security)

TLS or Transport Layer Security is a cryptographic standard based on SSL, used to make Internet transactions secure.

15.2.121 TCP/IP (Transmission Control Protocol over Internet Protocol)

TCP and IP are the fundamental protocols of the Internet, handling packet transmission and routing. There are a few alternative protocols that are used at this level of Internet structure, such as UDP.

15.2.122 Tor bridge

A bridge is a middleman Tor node that is not listed in the main public Tor directory, and so is possibly useful in countries where the public relays are blocked. Unlike the case of exit nodes, IP addresses of bridge nodes never appear in server log files and never pass through monitoring nodes in a way that can be connected with circumvention.

15.2.123 traffic analysis

Traffic analysis is statistical analysis of encrypted communications. In some circumstances traffic analysis can reveal information about the people communicating and the information being communicated.

15.2.124 tunnel

A tunnel is an alternate route from one computer to another, usually including a protocol that specifies encryption of messages.

15.2.125 UDP (User Datagram Packet)

UDP is an alternate protocol used with IP. Most Internet services can be accessed using either TCP or UDP, but there are some that are defined to use only one of these alternatives. UDP is especially useful for real-time multimedia applications like Internet phone calls (VoIP).

15.2.126 URL (Uniform Resource Locator)

The URL (Uniform Resource Locator) is the address of a Web site. For example, the URL for the World News section of the NY Times is <http://www.nytimes.com/pages/world/index.html>. Many censoring systems can block a single URL. Sometimes an easy way to bypass the block is to obscure the URL. It is for example possible to add a dot after the site name, so the URL <http://en.cship.org/wiki/URL> becomes <http://en.cship.org./wiki/URL>. If you are lucky with this little trick you can access blocked Web sites.

15.2.127 Usenet

Usenet is a more than 20-year-old discussion forum system accessed using the NNTP protocol. The messages are not stored on one server but on many servers which distribute their content constantly. Because of that it is impossible to censor Usenet as a whole, however access to Usenet can and is often blocked, and any particular server is likely to carry only a subset of locally-acceptable Usenet newsgroups. Google archives the entire available history of Usenet messages for searching.

15.2.128 VoIP (Voice over Internet Protocol)

VoIP refers to any of several protocols for real-time two-way voice communication on the Internet, which is usually much less expensive than calling over telephone company voice networks. It is not subject to the kinds of wiretapping practiced on telephone networks, but can be monitored using digital technology. Many companies produce software and equipment to eavesdrop on VoIP calls; securely encrypted VoIP technologies have only recently begun to emerge.

15.2.129 VPN (virtual private network)

A VPN (virtual private network) is a private communication network used by many companies and organizations to connect securely over a public network. Usually on the Internet it is encrypted and so nobody except the endpoints of the communication can look at the data traffic. There are various standards like IPSec, SSL, TLS. The use of a

VPN provider is a very fast, secure and convenient method to bypass Internet censorship with little risks but it generally costs money every month. Further, note that the VPN standard PPTP is no longer considered secure, and should be avoided.

15.2.130 whitelist

A whitelist is a list of sites specifically authorized for a particular form of communication. Filtering traffic can be done either by a whitelist (block everything but the sites on the list), a blacklist (allow everything but the sites on the list), a combination of the two, or by other policies based on specific rules and conditions.

15.2.131 World Wide Web (WWW)

The World Wide Web is the network of hyperlinked domains and content pages accessible using the Hypertext Transfer Protocol and its numerous extensions. The World Wide Web is the most famous part of the Internet.

15.2.132 Webmail

Webmail is e-mail service through a Web site. The service sends and receives mail messages for users in the usual way, but provides a Web interface for reading and managing messages, as an alternative to running a mail client such as Outlook Express or Thunderbird on the user's computer. For example a popular and free webmail service is <https://mail.google.com/>

15.2.133 Web proxy

A Web proxy is a script running on a Web server which acts as a proxy/gateway. Users can access such a Web proxy with their normal Web browser (like Firefox) and enter any URL in the form located on that Web site. Then the Web proxy program on the server receives that Web content and displays it to the user. This way the ISP only sees a connection to the server with the Web proxy since there is no direct connection.

15.2.134 WHOIS

WHOIS (who is) is the aptly named Internet function that allows one to query remote WHOIS databases for domain registration information. By performing a simple WHOIS search you can discover when and by whom a domain was registered, contact information, and more.

A WHOIS search can also reveal the name or network mapped to a numerical IP address

15.3 The necessity of Open Source

15.3 The necessity of Open Source

The last 20 years have seen network technology reaching ever more deeply into our lives, informing how we communicate and act within the world. With this come inherent risks: the less we understand the network environment we depend upon, the more vulnerable we are to exploitation.

This ignorance is something traditionally enjoyed by criminals. In recent years however some corporations and governments have exploited civilian ignorance in a quest for increased control. This flagrant and often covert denial of dignity breaches many basic rights, the right to privacy, in particular.

Closed source software has been a great boon to such exploitation – primarily due to the fact there is no code available for open, decentralised security auditing by the community. Under the auspices of hiding trade secrets, closed-source software developers have proven to be unwilling to explain to users how their programs work. This might not always be an issue were it not for the high stakes: identity theft, the distribution of deeply personal opinion and sentiment, a person's diverse interests and even his/her home increasingly come into close contact with software in a world-wide network context. As such, many people find themselves using software for personal purposes with full trust that it are secure. The Windows operating system itself is the most obvious

real-world example. Apple's OS X follows close behind, with large portions of the operating system's inner-workings barred from public inspection.

In Cryptography there is a strong principle, established in the 19th century by *Auguste Kerckhoff* (and hence named after him) which demands that

“[the encryption method] must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience”.

While this principle has been taken further by most scientific and (of course) open source communities – publishing their methods and inner-workings upfront, so potential weaknesses can be pointed out and fixed before further distribution – most distributors of proprietary software rely on obfuscation to hide the weaknesses of their software. As such they often prove to address newly discovered vulnerabilities in a non-transparent way – leaving many trusting users at risk of exploitation.

Of course it must be said that Open Source Software is as secure as you make it (and there is a lot of OSS written by beginners). However there are many good examples of well written, well managed software which have such a large (and concerned) user base that even the tiniest of mistakes are quickly found and dealt with. This is especially the case with software depended upon in a network context.

To use closed source software in a network context is not only to be a minority, it is to be overlooked by a vast community of concerned researchers and specialists that have your privacy and safety in mind.

N.B. There is also a more cynical view of Open Source Software, which points out that since nobody is paid full time to constantly review and regression test the latest tinkering by unskilled or deliberately malicious programmers, it can also suffer from major security

weaknesses which go undetected for long periods of time in complicated software, leaving it vulnerable to hackers, criminals and intelligence agencies etc. e.g. the (now fixed) Debian Linux predictable random number generator problem which led to the creation of lots of weak cryptographic keys.