Ruckus®
Simply Better Wireless.

technical note

Configuring ZoneDirector

For RADIUS Authentication

Configuring ZoneDirector

For RADIUS Authentication

Table of Contents

Configuring ZoneDirector

For RADIUS Authentication

Copyright Notice and Proprietary Information

Destination Control Statement

Technical data contained in this publication may be subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader’s responsibility to determine the applicable regulations and to comply with them.

Disclaimer

THIS DOCUMENTATION AND ALL INFORMATION CONTAINED HEREIN (“MATERIAL”) IS PROVIDED FOR GENERAL INFORMATION PURPOSES ONLY. RUCKUS AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THE MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OR THAT THE MATERIAL IS ERROR-FREE, ACCURATE OR RELIABLE. RUCKUS RESERVES THE RIGHT TO MAKE CHANGES OR UPDATES TO THE MATERIAL AT ANY TIME.

Limitation of Liability

IN NO EVENT SHALL RUCKUS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN CONTRACT OR TORT, ARISING FROM YOUR ACCESS TO, OR USE OF, THE MATERIAL.

Trademarks

Ruckus Wireless is a trademark of Ruckus Wireless, Inc. in the United States and other countries. All other product or company names may be trademarks of their respective owners.

Configuring ZoneDirector

For RADIUS Authentication

Intended Audience

This document provides an overview of how to configure a Ruckus Wi-Fi controller for RADIUS authentication. Step-by-step procedures for configuration and testing are demonstrated. Some knowledge of the ZoneDirector, RADIUS and 802.1X is recommended.

For more information on how to configure Ruckus products, please refer to the appropriate Ruckus user guide available on the Ruckus support site, http://support.ruckuswireless.com.

Configuring ZoneDirector

For RADIUS Authentication

Introduction

This document describes how to configure the ZoneDirector Wi-Fi controller for RADIUS-based authentication. The document is broken into the following main categories:

- Introduction
- RADIUS authentication overview
- Configuring AAA profiles
- Configuring 802.1X
- Configuring MAC authentication
- Troubleshooting

Requirements for this document

In order to successfully follow the steps in this document, the following equipment (at a minimum) is required and assumed:

- RADIUS server
- Ruckus ZoneDirector controller and AP

What Is NOT in this document

This document is not an exhaustive description of all possible solutions. It focuses on common RADIUS authentication scenarios and to integrate with Ruckus Wi-Fi controllers.

This document does not describe how to set up a RADIUS server and instead focuses on how to integrate a ZoneDirector into an existing environment.

Configuring ZoneDirector

For RADIUS Authentication

RADIUS Overview

This section describes common authentication scenarios that use RADIUS for authentication.

What is RADIUS?

Remote Authentication Dial-in User Service (RADIUS) is a networking protocol for centralized authentication services. RADIUS is a standard protocol described in IETF standards (RFC 2865 and others).

Although typically referred to as to a RADIUS server. It is important to understand that RADIUS is a communications protocol. It does not refer to any type of hardware or a specific server application software.

There are many options for authentication, of which RADIUS is one. Others include Active Directory (AD), LDAP, UNIX files, databases, etc.

AAA services

RADIUS provides more than simple authentication of a user or device. It also provides Authentication, Authorization and Accounting (AAA) services. AAA is described more fully in RFC 2903, 3539 and later documents.

The functions of a AAA server include:

- Authentication – check if a presented credential is valid or not
- Authorization – check if credentials are authorized for a particular service
- Accounting – log all authentication and authorization activities

It is important to note that not all authentication servers or services are AAA-compliant. i.e. not all comply with the protocol defined in the AAA standard. This does not mean that a server is not an authentication server. It does mean that type of server may not be able to participate in functions that rely on that protocol.

For example, the DIAMETER[1] protocol is AAA-compliant. Microsoft product Active Directory (AD) is not. AD is based on the LDAP protocol and does not understand or use the RADIUS protocol. Therefore, it cannot directly participate in any AAA activity in which the other party requires a AAA compliant protocol. This distinction is important to remember when designing an authentication infrastructure with networked devices.

[1] DIAMETER is a newer protocol, but is not as widely used. Even though a DIAMETER is twice a RADIUS.

RADIUS authentication scenarios

As the name implies, RADIUS was initially used primarily for user authentication via dial-up services such as modem and VPN. It has moved into a larger role as network authentication types have evolved. This guide concentrates on RADIUS as it is typically used in a Wi-Fi network. Other types of RADIUS authentication (such as wired version of these authentications) are beyond the scope of this document.

The most common scenarios that use RADIUS include:

- 802.1X
- MAC

RADIUS and AAA key concepts

A well-designed authentication infrastructure requires basic knowledge of how RADIUS functions and the various entities involved. RADIUS is usually made up of at least three components or entities.

RADIUS server – a server application that uses the RADIUS protocol and has access to a repository of user credentials

RADIUS (NAS) client[2] – a device that is responsible for sending user credentials to the server

Client – a user or device that possesses credentials and wants access to the network

Of these three entities, it is important to understand the client typically does not understand or use the RADIUS protocol. The NAS client does so on the client's behalf to the RADIUS server.

RADIUS security

The RADIUS protocol is a clear-text protocol, i.e. all messages are sent unencrypted[3]. There is authentication between the RADIUS server and the NAS client via a shared secret. A shared secret is a plain-text password that proves the NAS client's identity. RADIUS servers are typically configured to only accept certain types of devices and IP addresses. This limits authentication to devices that know the shared secret and have been preconfigured on the RADIUS server as trusted clients.

[2] The terms "RADIUS client" and "NAS client" are interchangeable for the purposes of this document

[3] RadSec is a version of RADIUS that uses TCP and TLS to secure communications. This tutorial concentrates on the original RADIUS protocol for simplicity and because it is still the most common protocol used.

RADIUS authentication flow

Simple RADIUS

The figure below shows the simplest type of RADIUS authentication. The client provides its credentials to a NAS client (an AP in this example). The AP transmits the credentials to the RADIUS server using the RADIUS protocol. How the client transmits its credentials is not relevant to this discussion. The RADIUS server does not necessarily know what the client is or the network details.

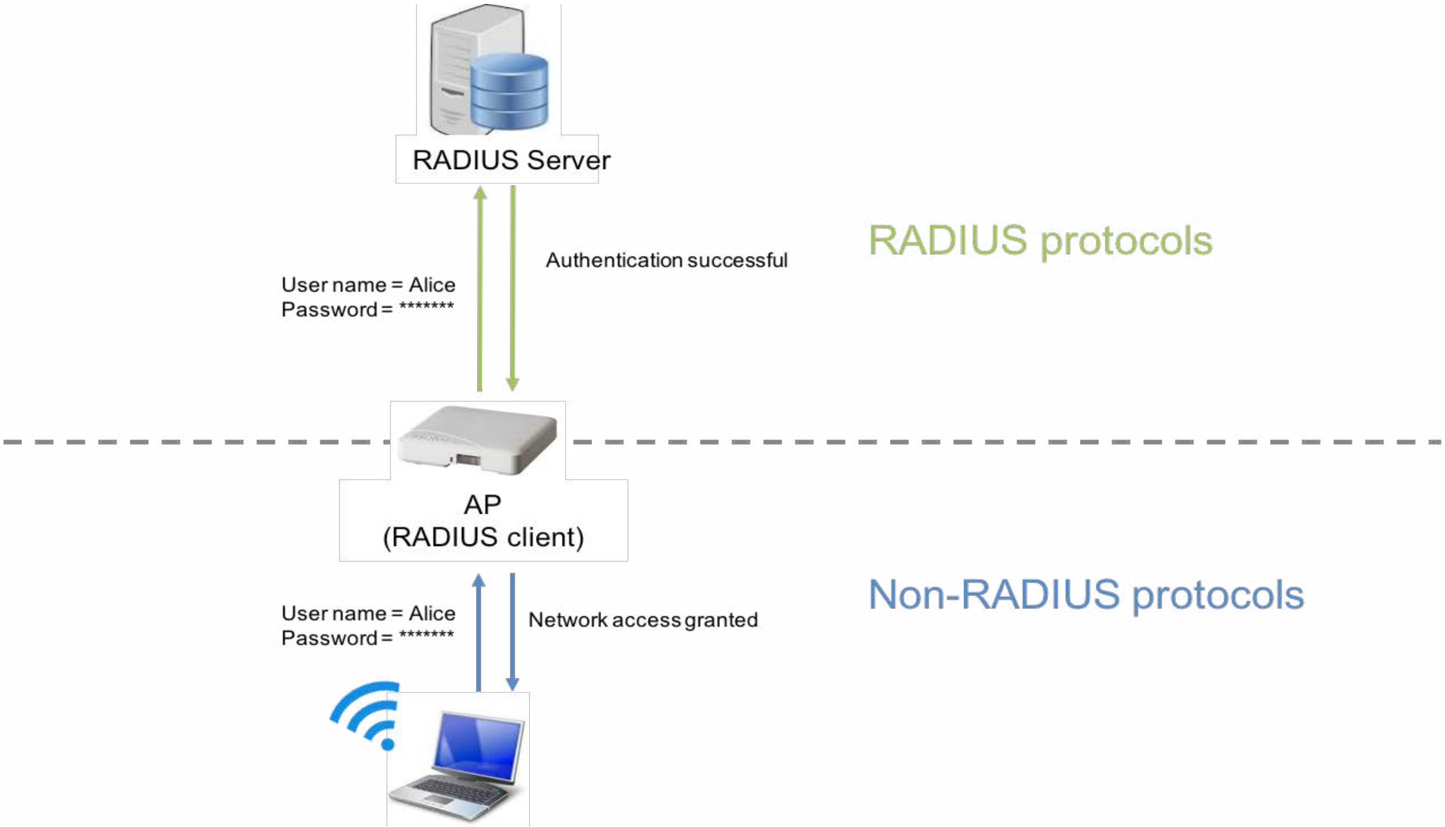


Figure 1 - Simple RADIUS authentication

Proxied RADIUS

A RADIUS server must be configured with the information for every NAS client that will connect. If there are many NAS clients – for example, a campus of APs - it can be tedious to configure on the server. To get around this, many Wi-Fi installations use an intermediate device to proxy the RADIUS communications. This has the advantage that the RADIUS server need only be configured to accept communications from the proxy client rather than every AP.

In the case of the figure below, the AP is a NAS client to the Wi-Fi controller. The Wi-Fi controller acts like a RADIUS server to the AP and a NAS client to the RADIUS server. Multiple proxies (APs, controllers, other RADIUS servers) may be used in between a client and the ultimate credential database. Each simply forwards the credentials onward.

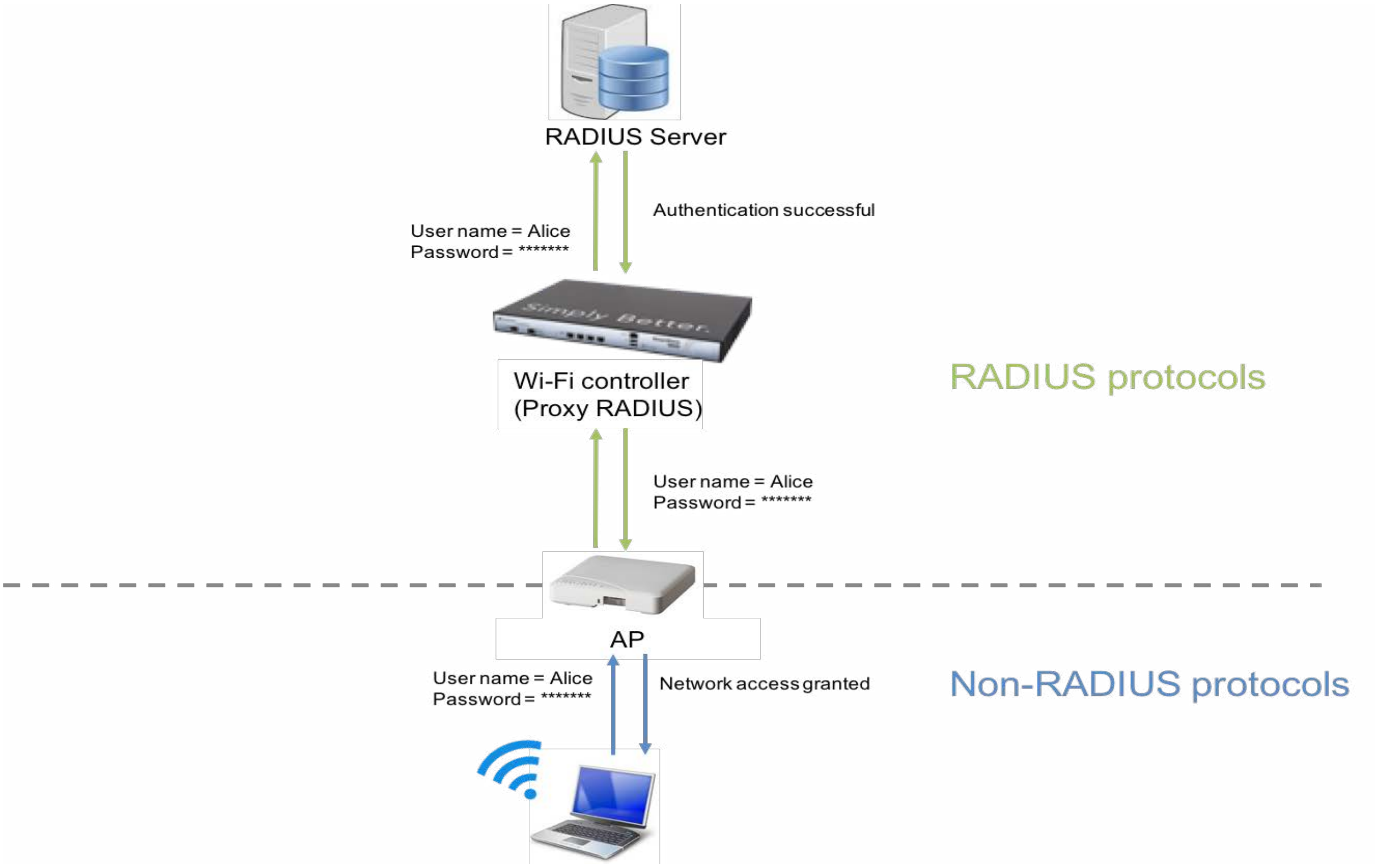


Figure 2 - AP proxies RADIUS requests to a Wi-Fi controller

Non-RADIUS credential repositories

User credentials are not always stored on the RADIUS server. In this case, many RADIUS servers support the ability to communicate with a third-party server on the backend using a different protocol. This is particularly common when the credentials are stored on an AD or LDAP server. The RADIUS server accepts the credentials from the NAS client via the RADIUS protocol. It then transmits those credentials using the appropriate protocol, e.g. LDAP to the backend server.

If the credentials are approved by the 3rd party server, the RADIUS server receives the confirmation via LDAP, then transmits the accept to the AP via the RADIUS protocol.

The figure below shows how traffic might flow between an AP to the Wi-Fi controller, to a RADIUS server and ultimately to an LDAP server. The process is reversed to communicate the authentication successful/unsuccessful result.

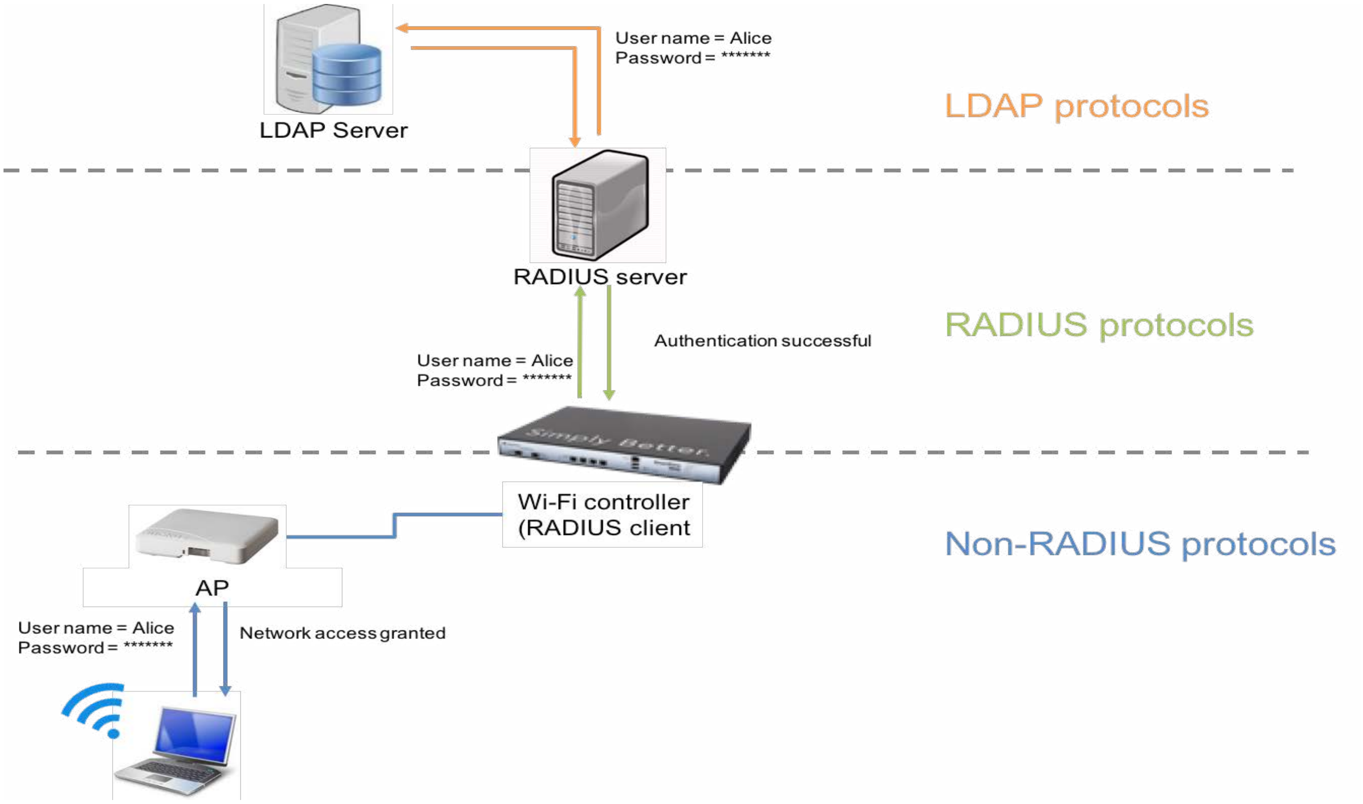


Figure 3 - RADIUS frontend for a 3rd party authentication server

MAC authentication

The simplest form of authentication is via the MAC address of the client. There must a repository of all client MACs that are allowed to access the network. In the case of MAC authentication, the repository is usually the RADIUS server.

MAC authentication flow

The steps in a MAC authentication over Wi-Fi are:

1. Client connects to the WLAN
2. AP submits the client MAC address to the RADIUS server
3. RADIUS server checks the MAC is valid
4. If the client MAC is in the database, the client is permitted network access, otherwise it is rejected

The important things to realize about MAC authentication are:

- The client knows nothing about and does not participate in the authentication. Only the AP knows MAC authentication is required and will perform the necessary steps
- MAC authentication is a low-level (layer 2) mechanism that does not offer encryption. The only way traffic can be encrypted between the client and the AP is if it is provisioned with some kind of “seed” material to base the encryption on such as a pre-shared key (PSK) or 802.1X (user name/password or certificate)

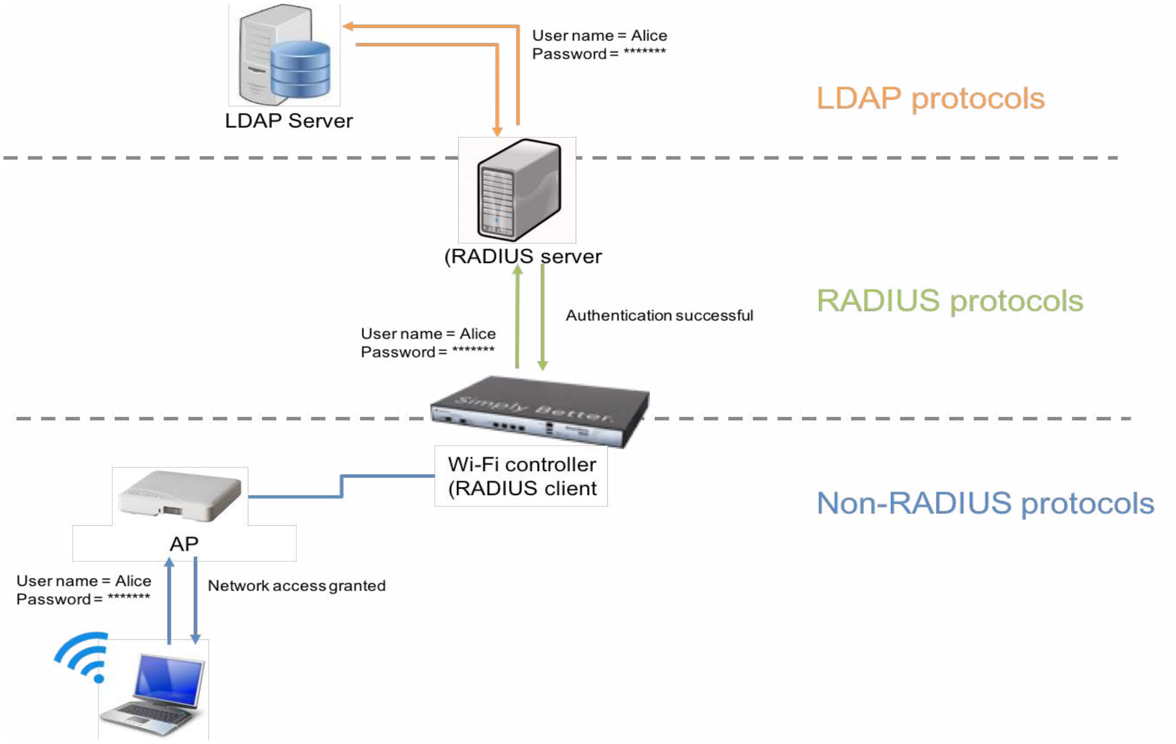


Figure 4 - Authentication via a 3rd party server

802.1X Authentication

What is 802.1X?

802.1X is an IEEE security standard for network access. Authentication is a key part of the 802.1X standard. Three devices participate in every 802.1X authentication:

- Supplicant – the client device
- Authenticator – the device that controls network access (port) and passes authentication messages to the authentication server (NAS client)
- Authentication Server – AAA-compliant authentication server

Extensible Authentication Protocol types (EAP)

802.1X provides a framework in which an authentication process and transmission of user credentials may be processed securely. It supports a wide variety of authentication types, commonly called EAP. Popular EAP types include:

- EAP-PEAP –- Protected EAP
- EAP-TLS – Transport Layer Security
- EAP-TTLS – Tunneled Transport Layer Security

There are many other EAP types available although not all are widely used. A more complete explanation of various EAP types is available on Wikipedia here.

802.1X authentication flow

802.1X authentication flow will be different depending on the type of authentication. The intent of all authentication types is to prevent the client from getting network access of any kind (including IP addresses) before it has passed authentication. It also ensures the client credentials are encrypted before transmission over the untrusted medium, e.g. Wi-Fi.

The following is a simplified authentication flow based on PEAP. PEAP uses an unsecure inner authentication protocol, MS-CHAPv2. Therefore, the credentials must be protected. This is accomplished by establishing an encrypted tunnel over which the client may safely send the credentials.

PEAP example: phase 1

1. Supplicant connects to the WLAN and sends an EAPoL-Start to begin
2. Authenticator sends an EAP request for the supplicant identity
3. Supplicant sends EAP response with a clear text outer identity name (this is not the client credentials)
4. Authenticator forwards the outer identity to the RADIUS server to begin authentication transaction
5. RADIUS server identifies itself by sending its X.509 certificate to the supplicant (avoid a Man-In-The-Middle attack)
6. Supplicant validates the server certificate
7. An encrypted TLS tunnel is created between the supplicant and the RADIUS server

PEAP example: phase 2

8. RADIUS server requests the real identity of the supplicant
9. Supplicant sends actual credentials (inner identity) within the encrypted tunnel
10. RADIUS server sends an EAP challenge request
11. Supplicant sends an EAP challenge response (hashed)
12. RADIUS server sends an EAP request with EAP-MSCHAPv2 success
13. TLS tunnel is torn down
14. 4-way Handshake

4-way handshake

Once EAP authentication is complete, there is an additional step that generates cryptographic keys between the authenticator and the supplicant. These keys are valid for the length of the session and are used to encrypt client data traffic sent between the client and the AP. This is called the 4-way handshake.

Configuring ZoneDirector

For RADIUS Authentication

Supported EAP types

Do I need to tell the ZoneDirector which EAP to use?

Some less common EAP types may require additional support on the controller, however in general the most popular EAP types do not. Examples of supported EAP types include:

- EAP-PEAP
- EAP-TTLS
- EAP-TLS
- EAP-SIM
- LEAP

If in doubt, please consult the ZoneDirector user guide for guidelines. The latest copies of the ZoneDirector user guide are available on the Ruckus support portal: http://support.ruckuswireless.com.

The Ruckus AP and ZoneDirector need to know 802.1X will be used when the WLAN is created. All the AP and ZoneDirector do are forward the messages to the RADIUS server. They do not alter them or change their actions in any way other than to allow or disallow access based on an Access-Accept or Access-Reject message from RADIUS. VLAN membership or WLAN access can also be specified via return attributes from RADIUS as well. For more information, please refer to the Ruckus Wireless application note: *Configuring Dynamic VLANs with RADIUS.*

AAA Profiles

This section describes how to configure a AAA server profile on a Ruckus ZoneDirector controller. When the controller is acting as a NAS or RADIUS client, it needs information about the RADIUS server in order to communicate. A valid AAA profile must be configured on the controller prior to creation of a WLAN.

The following is required information:

- RADIUS server IP address
- RADIUS server port number
- Shared secret
- NAS client protocol

Configuring ZoneDirector

For RADIUS Authentication

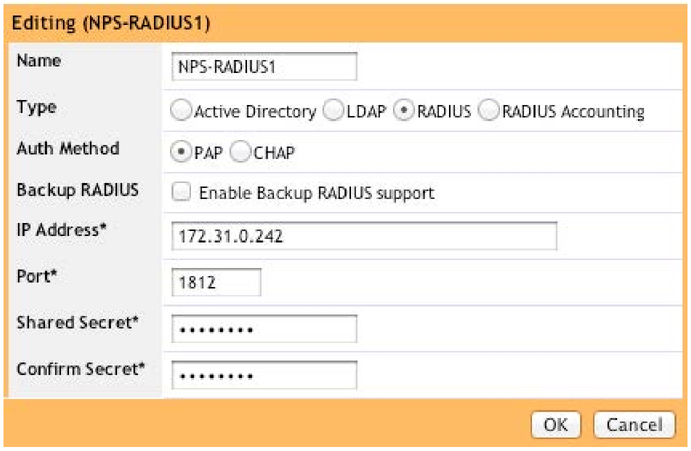
This information tells the controller how to contact the RADIUS server and authenticate itself as well as which protocol it should use for its own communications with the server. The protocol used by a ZoneDirector may be either PAP or CHAP. The controller does not act as an 802.1X supplicant.

Workflow steps

1. Create a AAA entry for the RADIUS server
2. Test the AAA entry

Create a AAA entry for the RADIUS server

1. Log on to the ZoneDirector's web UI
2. Go to Configure->AAA Servers
3. Click Create New and enter the information for the RADIUS server. Required information includes:
   - Server name
   - Type (RADIUS)
   - Authentication method for NAS client (CHAP or PAP)
   - IP Address of RADIUS server
   - Port number (most use 1812 by default)
   - Shared Secret – the secret entered on the RADIUS server for the ZoneDirector NAS client entry



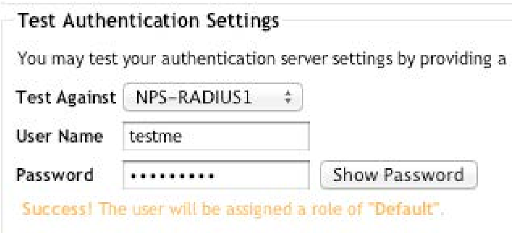
Editing (NPS-RADIUS1)

Name	NPS-RADIUS1
Type	○ Active Directory ○ LDAP ● RADIUS ○ RADIUS Accounting
Auth Method	● PAP ○ CHAP
Backup RADIUS	□ Enable Backup RADIUS support
IP Address*	172.31.0.242
Port*	1812
Shared Secret*	••••••••
Confirm Secret*	••••••••

4. Click OK

Test ZoneDirector to RADIUS server communications

If PAP or CHAP is permitted in your connection request policy on the RADIUS server, you may test communications with the server now to make sure it works[4]. You should make sure your RADIUS server is configured to allow either PAP or CHAP according to what you selected in the AAA profile above.



Test Authentication Settings

You may test your authentication server settings by providing a

Test Against: NPS-RADIUS1

User Name: testme

Password: •••••••••

Success! The user will be assigned a role of "Default".

This test checks if communications are operational between the ZoneDirector and the RADIUS server. It verifies the IP address, shared secret, etc. is correct. **Successfully testing RADIUS communications using this test does not guarantee 802.1X will work as it relies on a different protocol.** To make sure this configuration works with 802.1X please test with a supplicant. Instructions are available in Appendix A: 802.1X Testing.

[4] Enabling PAP or CHAP is a security risk as it is a very insecure protocol. However, you can always enable it temporarily to do this test and then disable

Configuring ZoneDirector

For RADIUS Authentication

802.1X Configuration

This section describes how to configure an 802.1X WLAN on a Ruckus ZoneDirector controller.

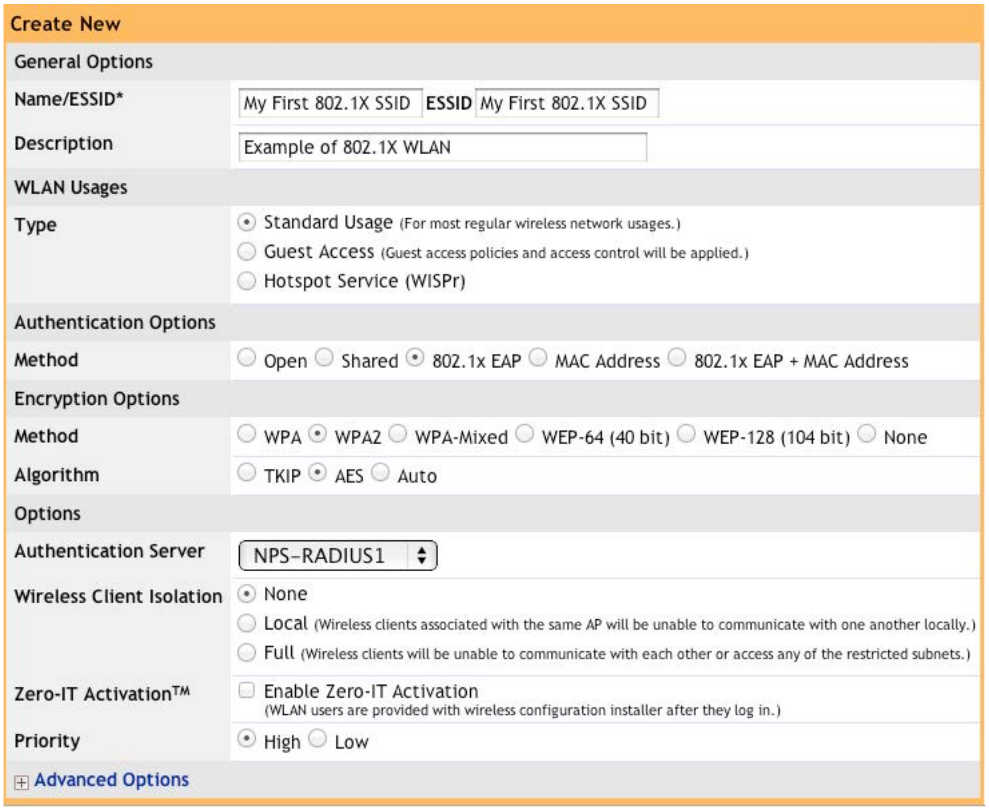
Workflow steps

1. Create a AAA entry for the RADIUS server
2. Test the AAA server communications (optional)
3. Create an 802.1X-enabled WLAN
4. Connect a supplicant and test

The section assumes the first two steps have already been performed. An 802.1X WLAN may not be configured unless a AAA profile already exists. For instructions on how to create a AAA profile, please refer to the chapter titled AAA Profiles.

Create an 802.1X-enabled WLAN

1. Log on to the ZoneDirector's web UI
2. Go to Configure->WLANs
3. Click Create New and enter the appropriate information for your SSID name, encryption, etc. The Authentication Method must be set to 802.1x EAP



Create New	
General Options	
Name/ESSID*	My First 802.1X SSID **ESSID** My First 802.1X SSID
Description	Example of 802.1X WLAN
WLAN Usages	
Type	● Standard Usage (For most regular wireless network usages.) ○ Guest Access (Guest access policies and access control will be applied.) ○ Hotspot Service (WISPr)
Authentication Options	
Method	○ Open ○ Shared ● 802.1x EAP ○ MAC Address ○ 802.1x EAP + MAC Address
Encryption Options	
Method	○ WPA ● WPA2 ○ WPA-Mixed ○ WEP-64 (40 bit) ○ WEP-128 (104 bit) ○ None
Algorithm	○ TKIP ● AES ○ Auto
Options	
Authentication Server	NPS-RADIUS1
Wireless Client Isolation	● None ○ Local (Wireless clients associated with the same AP will be unable to communicate with one another locally.) ○ Full (Wireless clients will be unable to communicate with each other or access any of the restricted subnets.)
Zero-IT Activation™	□ Enable Zero-IT Activation (WLAN users are provided with wireless configuration installer after they log in.)
Priority	● High ○ Low
⊞ Advanced Options	

4. Make sure to select the correct RADIUS server from the drop-down box

5. If using a RADIUS accounting server, open the Advanced Options and select it from the drop-down.

6. Click OK to save your changes.

That’s it for the ZoneDirector. The next step is to test with a real 802.1X supplicant client.

Test an 802.1X supplicant

At this point the system should be ready for a client test of 802.1X over wireless. Details on setting up client supplicants are very different depending on the OS. Some useful RADIUS supplicant test utilities are described in Appendix A: 802.1X Testing.

MAC Authentication Configuration

This section describes how to configure a WLAN with MAC authentication on a Ruckus ZoneDirector controller.

Workflow steps

1. Create a AAA entry for the RADIUS server

2. Test the AAA server communications (optional)

3. Create MAC authentication-enabled WLAN

4. Connect a client and test

The section assumes the first two steps have already been performed. A MAC authentication WLAN may not be configured unless a AAA profile already exists. For instructions on how to create a AAA profile, please refer to the chapter titled AAA Profiles.

Verify MAC address format

When a MAC address lookup is performed by the controller against the RADIUS server it must have the same format that is used by the server. The lookup is case sensitive. The octet separators must also match. Common formats include:

- aabbccddeeff
- aa:bb:cc:dd:ee:ff
- aa-bb-cc-dd-ee-ff
- AABBCCDDEEFF
- AA:BB:CC:DD:EE:FF
- AA-BB-CC-DD-EE-FF

Create a MAC authentication WLAN

1. Log on to the ZoneDirector’s web UI

2. Go to Configure->WLANs

3. Click Create New and enter the appropriate information for your SSID name. The Authentication Method must be set to MAC Address

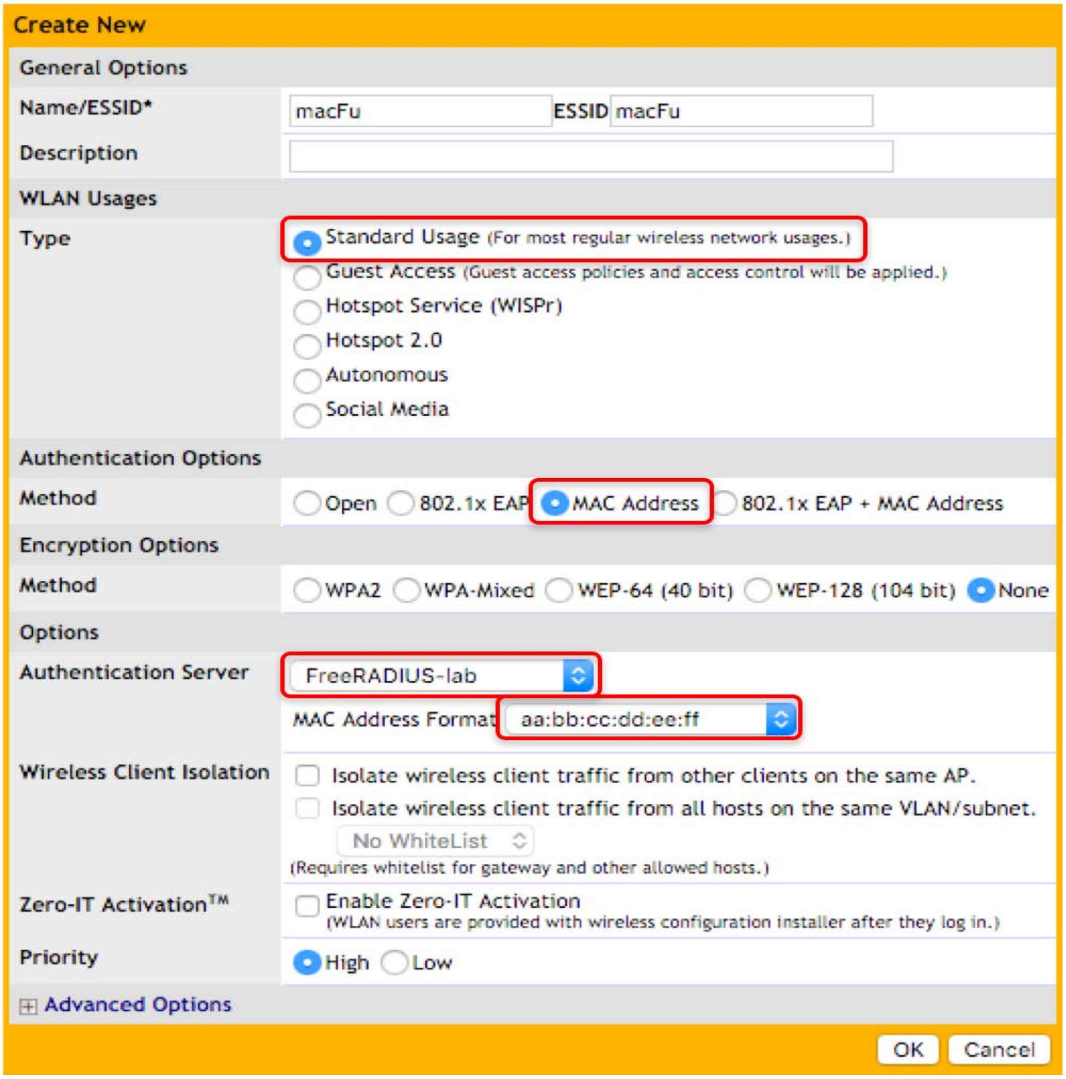
4. The Encryption Method is set to None

Configuring ZoneDirector

For RADIUS Authentication

5. Select the RADIUS server AAA profile from the Authentication Server drop-down box

6. Select the format used to store MAC addresses in your RADIUS server



Create New

General Options

Name/ESSID*	macFu ESSID macFu
Description	

WLAN Usages

Type
- ● Standard Usage (For most regular wireless network usages.)
- ○ Guest Access (Guest access policies and access control will be applied.)
- ○ Hotspot Service (WISPr)
- ○ Hotspot 2.0
- ○ Autonomous
- ○ Social Media

Authentication Options

Method: ○ Open ○ 802.1x EAP ● MAC Address ○ 802.1x EAP + MAC Address

Encryption Options

Method: ○ WPA2 ○ WPA-Mixed ○ WEP-64 (40 bit) ○ WEP-128 (104 bit) ● None

Options

Authentication Server: FreeRADIUS-lab

MAC Address Format: aa:bb:cc:dd:ee:ff

Wireless Client Isolation
- □ Isolate wireless client traffic from other clients on the same AP.
- □ Isolate wireless client traffic from all hosts on the same VLAN/subnet.

No WhiteList

(Requires whitelist for gateway and other allowed hosts.)

Zero-IT Activation™: □ Enable Zero-IT Activation (WLAN users are provided with wireless configuration installer after they log in.)

Priority: ● High ○ Low

7. Click OK to create the WLAN

Test with a client

Once the WLAN is configured and broadcast by an AP. Connect a device whose MAC address is already in the RADIUS server credentials repository to verify it works correctly.

Troubleshooting Tips

There are several components involved in 802.1X and MAC authentication. To troubleshoot, first isolate the problem component – or at least the first component failure in the process. Specific steps can be taken from there.

Troubleshooting NAS client to RADIUS communications

All 802.1X authentication will fail if the controller is unable to reach the RADIUS server and successfully authenticate itself. The following are common reasons why this may be the case:

- Shared secret is not the same on the NAS client and RADIUS server – try typing it again
- No IP connectivity – try pinging from one to the other
- Wrong RADIUS ports configured in the AAA profile or a firewall is blocking the ports
- If the AAA test (ZoneDirector) doesn't work, make sure PAP/CHAP is enabled. This test only works with PAP or CHAP

Troubleshooting 802.1X

In the case of 802.1X it is often easiest to begin troubleshooting from the endpoint (client).

Certificates (802.1X)

Client-side Validation
The number one reason an 802.1X connection fails on the client is when there is a problem with the server certificate[5]. This is particularly true if a private CA or a self-signed certificate is used. If the client cannot validate the RADIUS server certificate it will reject the connection and authentication will fail.

An easy way to check if this is the problem is to disable certificate validation on the client. If the client successfully authenticates the issue is with client validation. Resolution usually involves installing the correct root CA chain on the client.

RADIUS server configuration (802.1X)

Another common issue can occur when the RADIUS server's name does not match what is on its certificate. In this case, the two must be reconciled: either a new certificate with the correct name is installed or the server's name is changed to match the certificate.

A server may have a valid certificate that matches its name and is expired or revoked. This will also cause the client validation check to fail.

No Server Certificate
802.1X authentication requires the RADIUS server have an X.509 (SSL) certificate installed. If not server certificate is installed the server cannot perform EAP authentication such as PEAP, EAP-TLS and EAP-TTLS. A tool such as EAPTest described in Appendix A can easily show what is happening with the certificate.

Incorrect Server EAP Configuration
A common problem can occur when an EAP type is used that has not been configured on the RADIUS server. For example, a working 802.1X authentication is tested and validated for EAP-PEAP. However, that does not mean an EAP-TLS supplicant will work without some change to the RADIUS server configuration.

A working 802.1X authentication that suddenly stops working may be due to:

- Switching to a new RADIUS server that is not configured for the correct EAP
- Removing a RADIUS proxy device that was used as an intermediary. A RADIUS proxy can be used to connect to another RADIUS server and does not necessary have to be a traditional server

Client configuration (802.1X)

Most clients are able to determine the correct 802.1X authentication type on their own, but some may need manual configuration. It is important to make sure this is correct. For example, a client using WPA-TKIP will not be able to connect to a WLAN configured for WPA2-AES only.

Similarly, a client may not be able to negotiate a common authentication protocol with the RADIUS server. For example, if the client can only do PEAP but the RADIUS server is configured to only support EAP-TLS.

OS Limitations
Although most modern devices support the most popular EAP methods (PEAP, EAP-TLS and EAP-TTLS) it is not guaranteed. Always check the vendor's documentation first.

Troubleshooting certificates (802.1X)

Client
There are several reasons why a client may have a problem with a certificate. These include:

- RADIUS server presented a certificate that is signed by an unknown/untrusted CA
- RADIUS server presented a certificate that does not match its hostname
- The certificate has expired
- The client does not have a valid certificate to present to the server for an EAP type that requires client-side certificates such as EAP-TLS

Any of these problems will cause the client to fail to connect (often silently). In the case of a Windows client, the system will show messages similar to “Attempting to connect … Attempting to connect ... unable to connect”. It will repeat this cycle on and on.

The simplest way to test a server-side certificate problem is to disable server certificate validation on the client. If the client is able to connect, the problem is definitely on the server certificate side. One other thing to try is to connect with a different client (different OS) and see if it exhibits similar behavior.

Troubleshooting MAC authentication

RADIUS server configuration (MAC authentication)

If the controller to RADIUS server communication is correct and the client still fails to authentication the issue is likely a mismatch between the MAC format used in the RADIUS server’s repository vs. what the controller was configured to use the WLAN configuration.

Client configuration (MAC authentication)

There is no configuration required on the client side for MAC authentication.

Appendix A: 802.1X Testing

This section describes tools for validating an 802.1X WLAN.

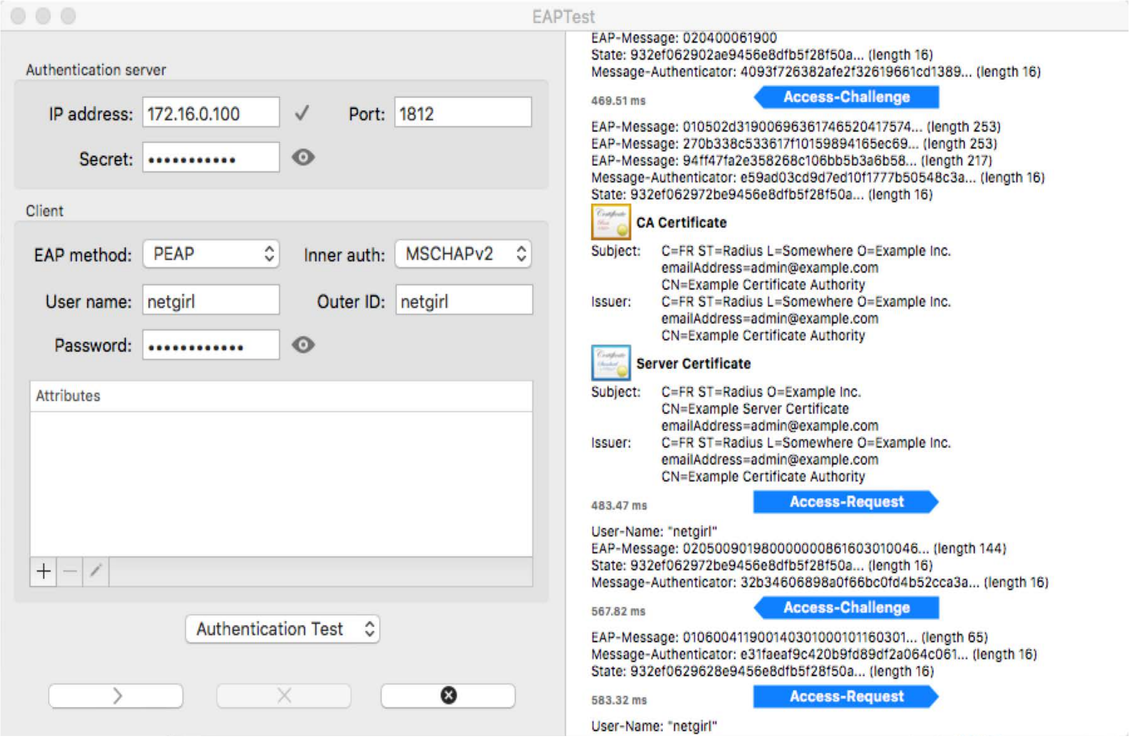
Test Clients

The only way to check if 802.1X is working correctly is with a supplicant. This can be done with a client device (recommended). There are also test utilities available that will set up an 802.1X connection.

Instructions on how to configure a specific client device are outside the scope of this document. To learn how to configure a supplicant client such as a laptop or smartphone, please consult the vendor's documentation.

EAPTest (Mac OS)

EAPTest is a commercial utility available for Mac OS on the App Store. It can be used to test wired and wireless connections using a wide variety of EAP types including EAP-PEAP, EAP-TLS and EAP-TTLS.



Authentication server

IP address: 172.16.0.100 ✓ Port: 1812

Secret: ••••••••••••

Client

EAP method: PEAP Inner auth: MSCHAPv2

User name: netgirl Outer ID: netgirl

Password: •••••••••••••

```
EAP-Message: 020400061900
State: 932ef062902ae9456e8dfb5f28f50a... (length 16)
Message-Authenticator: 4093f726382afe2f32619661cd1389... (length 16)

469.51 ms              Access-Challenge

EAP-Message: 010502d319006963617465204175 74... (length 253)
EAP-Message: 270b338c533617f10159894165ec69... (length 253)
EAP-Message: 94ff47fa2e358268c106bb5b3a6b58... (length 217)
Message-Authenticator: e59ad03cd9d7ed10f1777b50548c3a... (length 16)
State: 932ef062972be9456e8dfb5f28f50a... (length 16)
CA Certificate
Subject:   C=FR ST=Radius L=Somewhere O=Example Inc.
           emailAddress=admin@example.com
           CN=Example Certificate Authority
Issuer:    C=FR ST=Radius L=Somewhere O=Example Inc.
           emailAddress=admin@example.com
           CN=Example Certificate Authority
Server Certificate
Subject:   C=FR ST=Radius O=Example Inc.
           CN=Example Server Certificate
           emailAddress=admin@example.com
Issuer:    C=FR ST=Radius L=Somewhere O=Example Inc.
           emailAddress=admin@example.com
           CN=Example Certificate Authority

483.47 ms              Access-Request

User-Name: "netgirl"
EAP-Message: 020500901980000000861603010046... (length 144)
State: 932ef062972be9456e8dfb5f28f50a... (length 16)
Message-Authenticator: 32b34606898a0f66bc0fd4b52cca3a... (length 16)

567.82 ms              Access-Challenge

EAP-Message: 010600411900140301000101160301... (length 65)
Message-Authenticator: e31faeaf9c420b9fd89df2a064c061... (length 16)
State: 932ef0629628e9456e8dfb5f28f50a... (length 16)

583.32 ms              Access-Request

User-Name: "netgirl"
```

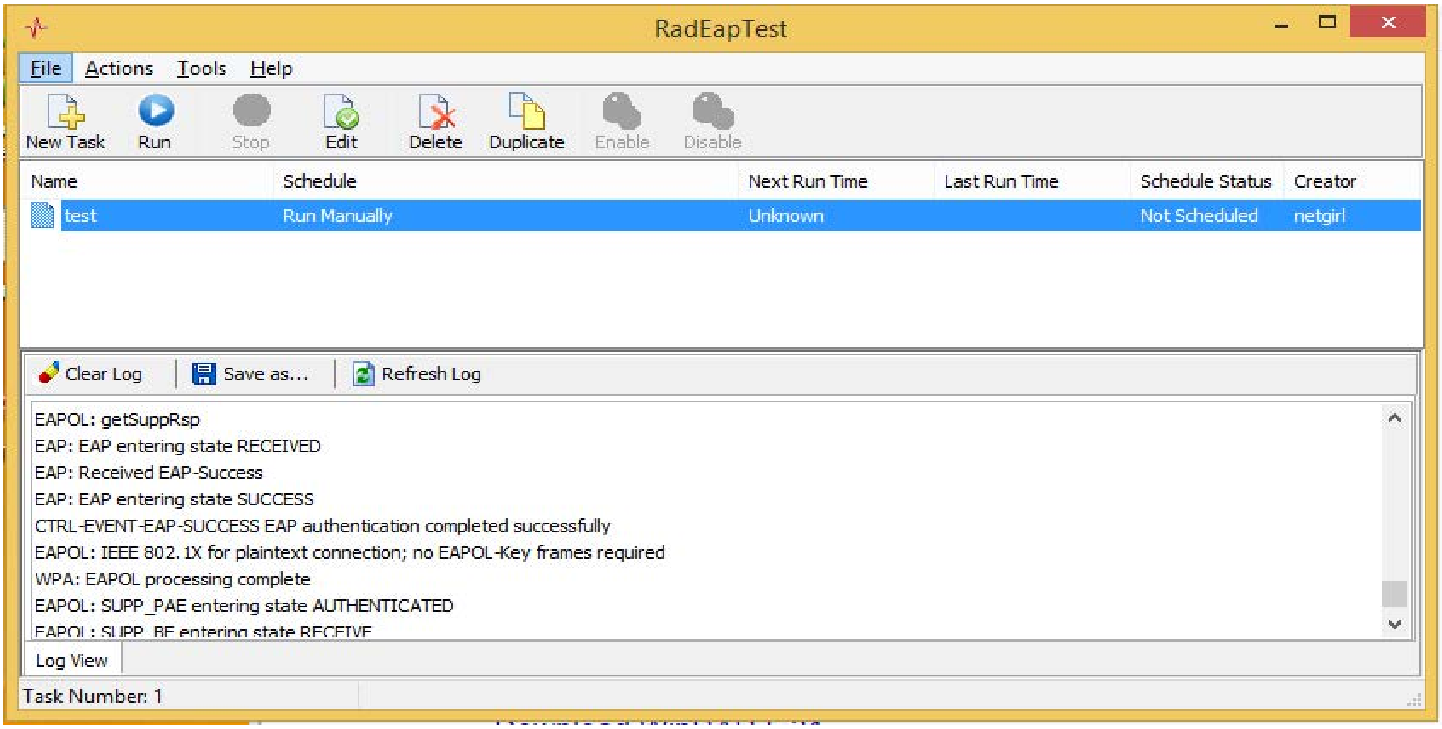
Figure 5 - EAPTest utility

The EAPTest tool is very easy to use and provides detailed information for every step of the process include (as shown above) certificate exchanges, received attributes, etc.

RadEapTest (Windows)

RadEapTest is a commercial utility available for Windows. It can be used to test wired and wireless connections using a wide variety of EAP types including EAP-PEAP, EAP-TLS and EAP-TTLS.

The RadEapTest tool provides detailed logs with packet-by-packet transactional information.



Name	Schedule	Next Run Time	Last Run Time	Schedule Status	Creator
test	Run Manually	Unknown		Not Scheduled	netgirl

```
EAPOL: getSuppRsp
EAP: EAP entering state RECEIVED
EAP: Received EAP-Success
EAP: EAP entering state SUCCESS
CTRL-EVENT-EAP-SUCCESS EAP authentication completed successfully
EAPOL: IEEE 802.1X for plaintext connection; no EAPOL-Key frames required
WPA: EAPOL processing complete
EAPOL: SUPP_PAE entering state AUTHENTICATED
EAPOL: SUPP_BE entering state RECEIVE
```

Task Number: 1

Figure 6 - RadEapTest utility