



Academy Cloud Foundations (ACF)
Module 03 Student Guide
Version 1.0.5

100-ACFNDS-10-EN-SG

© 2018 Amazon Web Services, Inc. or its affiliates. All rights reserved.

This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited.

Corrections or feedback on the course, please email us at:

aws-course-feedback@amazon.com.

For all other questions, contact us at:

<https://aws.amazon.com/contact-us/aws-training/>.

All trademarks are the property of their owners.

Contents

[Module 03: AWS Cloud Security](#)

4



Welcome to Module 3 – AWS Cloud Security.

What's In This Module



- 📁 Part 1: AWS Shared Responsibility Model
- 📁 Part 2: AWS Identity and Access Management (IAM)
- 📁 Part 3: AWS Trusted Advisor
- 📁 Part 4: AWS CloudTrail
- 📁 Part 5: AWS Config
- 📁 Part 6: AWS Day One Best Practice Review
- 📁 Part 7: AWS Security and Compliance Programs
- 📁 Part 8: AWS Security Resources
- 📁 Optional: AWS Day One Demonstration

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Security is of the utmost importance to AWS. AWS delivers a scalable cloud computing environment designed for high availability and dependability, while providing the tools that enable you to run a wide range of applications. Helping to protect the confidentiality, integrity, and availability of your systems and data is critical to AWS, as is maintaining customer trust and confidence. This module is intended to provide an introduction to AWS's approach to security, including the controls in the AWS environment and some of the products and features that AWS makes available to customers to meet their security objectives.

In part one, we will review the Shared Responsibility Model. In part two, we'll examine IAM including users, groups, and roles. In part three, we'll explore AWS Trusted Advisor. In part four, we'll explore AWS CloudTrail. In part five, we'll explore AWS Config. In part six, we'll review Day One best practices. In parts seven and eight, we'll discuss security, compliance issues and resources. An optional step by step walk through of Day One is available at the end of this module.

Module Overview



Review and understand the key security concepts related to the Shared Responsibility Model and IAM:

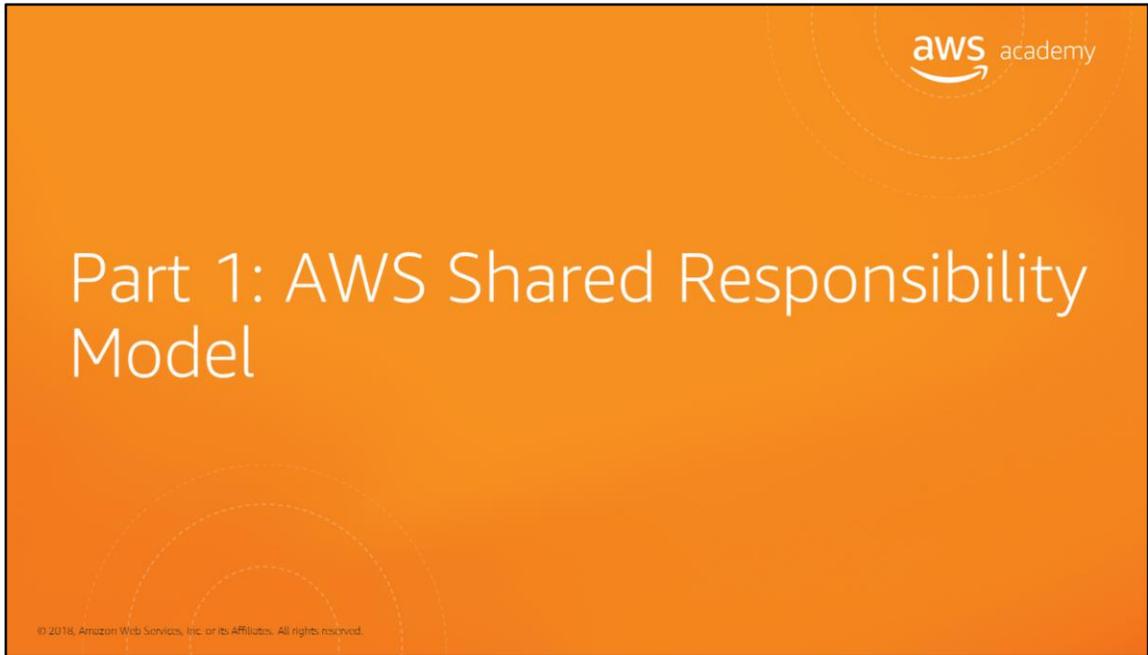
- 📌 Describe the AWS Shared Responsibility Model.
- 📌 Examine IAM users, groups and roles.
- 📌 Describe different types of security credentials.
- 📌 Review the AWS Trusted Advisor checks.
- 📌 Discuss security compliance.
- 📌 Understand best practices on Day 1 with a new AWS account.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

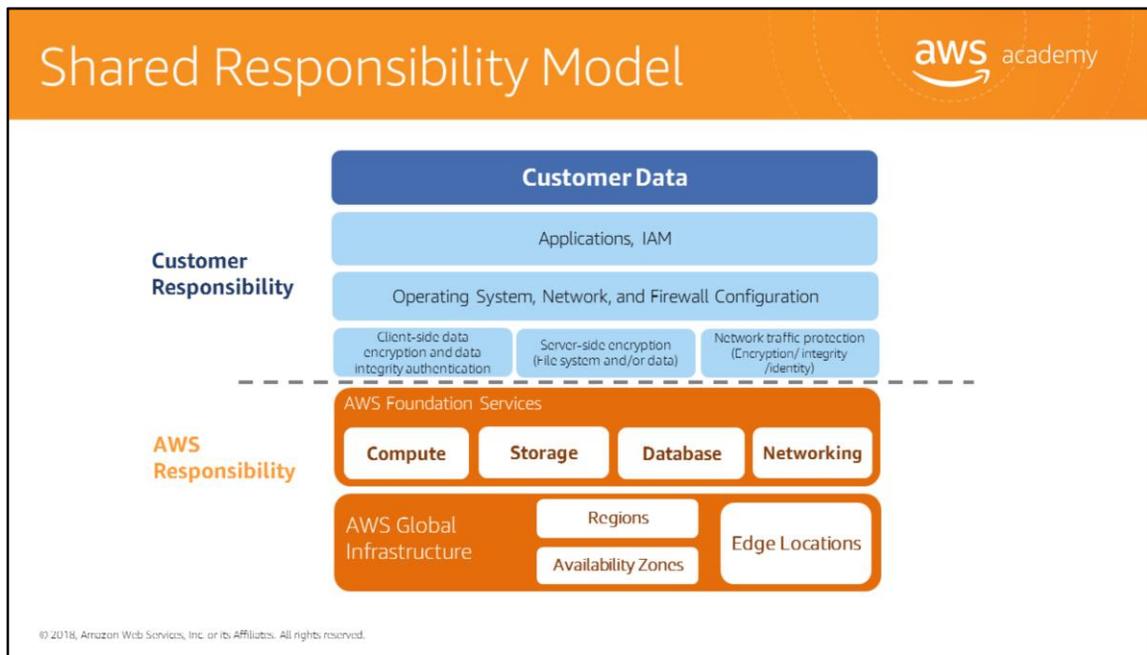
The goal of this module is to familiarize you with all of the security considerations for your cloud solution. We'll review security tools and best practices to help you understand key security concepts related to the Shared Responsibility Model and IAM.

In this module, we'll describe the AWS Shared Responsibility Model, examine IAM users, groups and roles and describe different types of security credentials. We'll review the AWS Trusted Advisor checks and discuss security compliance.

The demonstration of Day 1 best practices will walk you through the best practices for setting up an account from beginning to end.



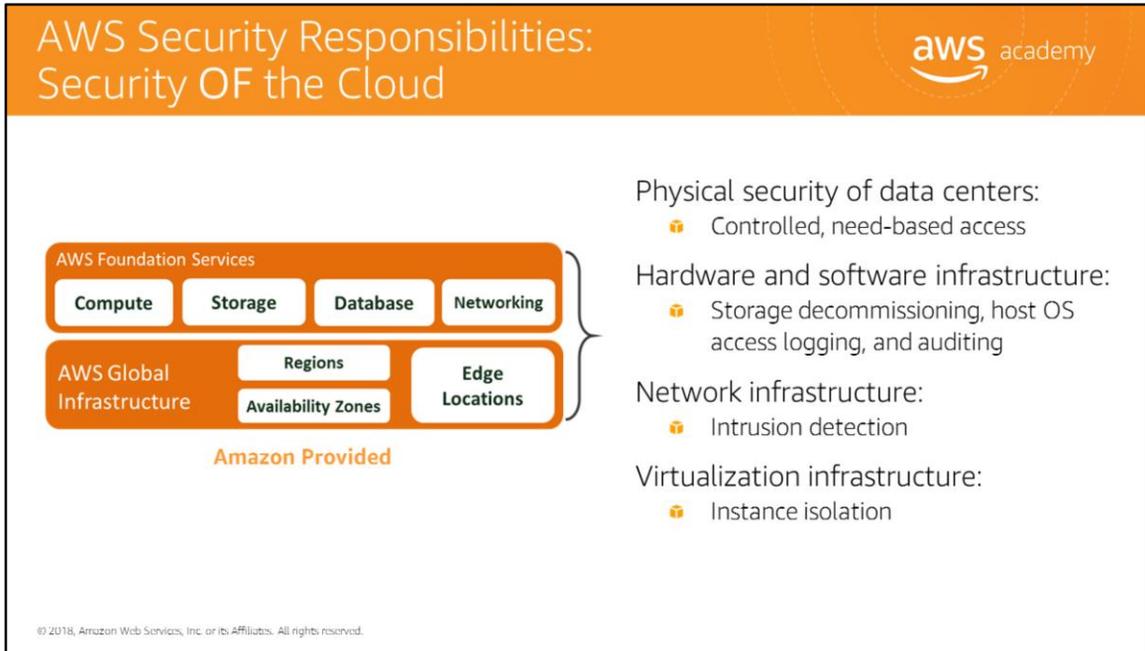
Introducing Part 1, the AWS Shared Responsibility Model.



AWS provides the same approach to security that companies have been using for decades, while allowing the flexibility and low cost of cloud computing. There is nothing inherently inconsistent about providing on-demand infrastructure while also providing the security isolation that companies expect in their existing, privately owned environments.

Once the customer starts using AWS, Amazon shares the responsibility of securing the customer's data in AWS cloud with its customers, making AWS security a shared responsibility. This concept is known as the Shared Responsibility Model.

Let's take a closer look at who's responsible for which aspects of security in the Shared Responsibility Model.



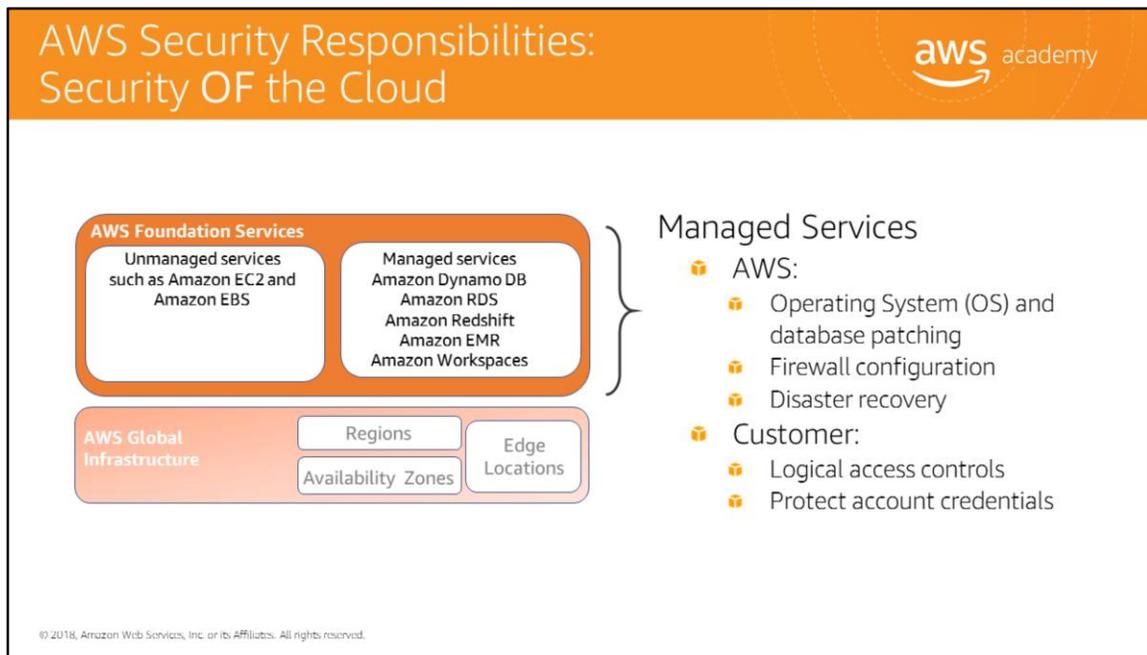
AWS is responsible for security *of* the cloud. But what does that mean?

Under the Shared Responsibility Model, AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. It means that AWS is responsible for protecting the global infrastructure that runs all of the services offered in the AWS cloud which include AWS Regions, Availability Zones, and Edge Locations.

For example, AWS handles the security of the cloud, specifically the physical infrastructure that hosts your resources, which include:

- **Physical security of data centers** with controlled, need-based access, located in nondescript facilities, 24/7 security guards, two-factor authentication, access logging and review, video surveillance, and disk degaussing and destruction.
- **Hardware infrastructure** including servers, storage devices, and other appliances that Amazon Web Services rely on.
- **Software infrastructure** which hosts operating systems, service applications, and virtualization software.
- **Network infrastructure** including routers, switches, load balancers, firewalls, and cabling. This includes continuous network monitoring at external boundaries, secure access points, and redundant infrastructure with intrusion detection.
- **Virtualization** infrastructure including instance isolation.

Protecting this infrastructure is the number one priority for AWS. While you can not visit AWS data centers or offices to see this protection firsthand, Amazon provides several reports from third-party auditors who have verified our compliance with a variety of computer security standards and regulations.



In addition to protecting this global infrastructure, AWS is responsible for the security configuration of its products that are considered foundational, which include Compute, Storage, Database, and Networking. Managed services include Amazon DynamoDB, Amazon Relational Database Service (or RDS), Amazon Redshift, Amazon Elastic MapReduce, Amazon WorkSpaces, and others.

For these services, AWS will handle basic security tasks like Operating System and database patching, firewall configuration, and Disaster Recovery. As a customer, this is valuable because you do not need to worry about patching, maintaining, or installing antivirus software. Amazon takes care of it so the customer can focus on what goes into the environment. For most of the managed services, the customer needs to configure logical access controls and protect account credentials. Some managed services may require additional tasks such as setting up database user accounts, but overall the security configuration work is performed by AWS.

Here are some examples of controls that are managed by AWS, AWS customers, or both.

Inherited Controls - Controls that a customer fully inherits from AWS, such as physical and environmental controls.

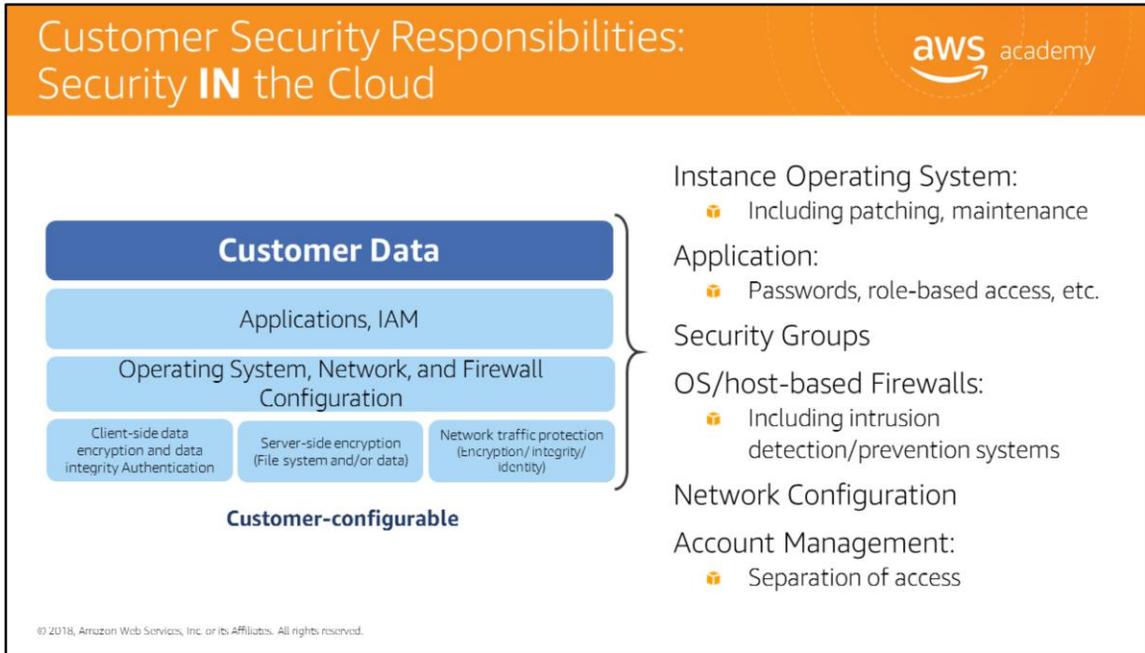
Shared Controls - Controls that apply to both the infrastructure layer and customer layers, but in completely separate contexts or perspectives. In a shared control, AWS provides the requirements for the infrastructure, and the customer must provide their own control implementation within their use of AWS services.

Examples include:

- **Patch Management** - AWS is responsible for patching and fixing flaws within the infrastructure, but customers are responsible for patching their guest OS and applications.
- **Configuration Management** - AWS maintains the configuration of its infrastructure devices, but customers are responsible for configuring their own guest operating systems, databases, and

applications.

- **Awareness and Training** - AWS trains AWS employees, but a customer must train their own employees.
- **Customer-specific** - Controls that are solely the responsibility of the customer based on the application they are deploying within AWS services.
- **Service and Communications Protection** or **Zone Security**, which may require a customer to route or zone data within specific security environments.



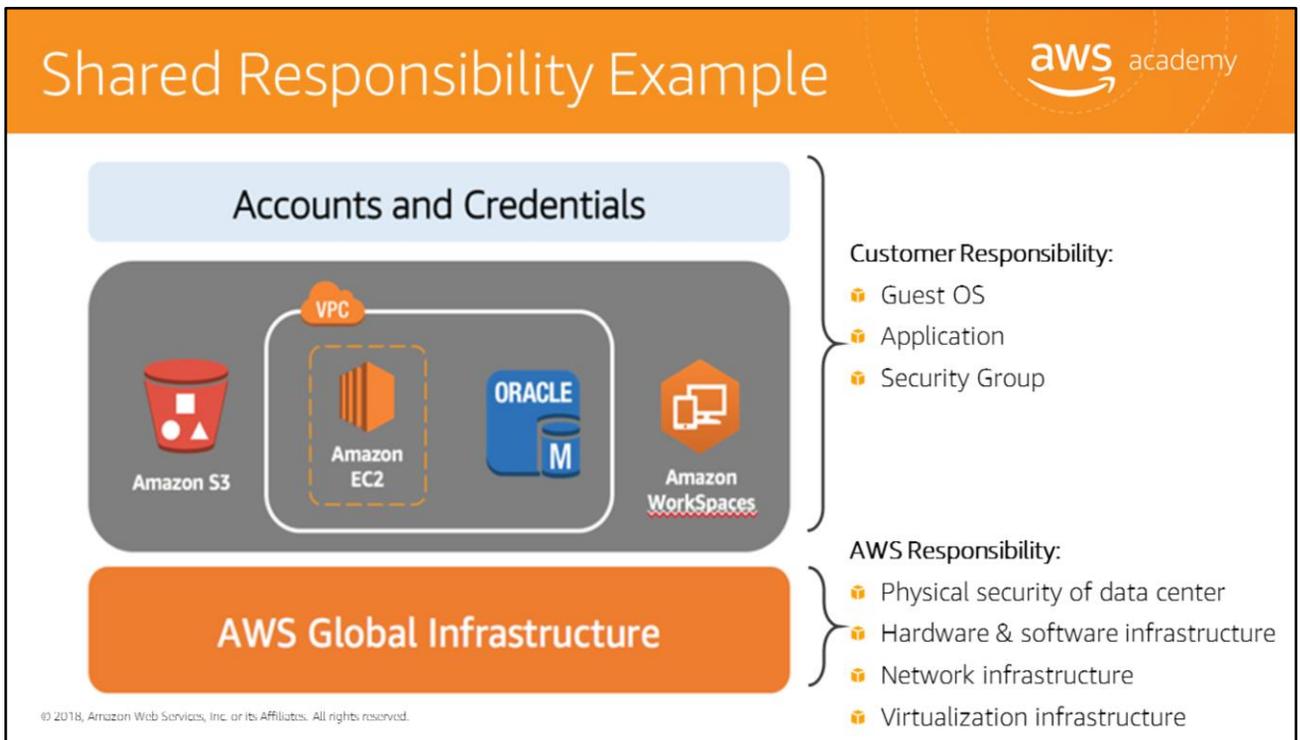
While the cloud infrastructure is secured and maintained by AWS, customers are responsible for security of everything they put **in** the cloud. The customer is responsible for what is implemented using AWS and for the applications connected to AWS. The security steps you need to take depend on the services you use and the complexity of your system.

This includes selecting the instance operating system, the application, security groups, firewalls, network configuration and account management.

When using AWS services, customers maintain complete control over their content and are responsible for managing critical content security requirements, including:

- What content they choose to store on AWS.
- Which AWS services are used with the content?
- In what country that content is stored.
- The format and structure of that content and whether it is masked, anonymized, or encrypted.
- Who has access to that content and how those access rights are granted, managed, and revoked?

Customers retain control of what security they choose to implement to protect their own data, environment, applications, IAM, and operating systems. This basically means that the Shared Responsibility Model changes depending on the AWS services the customer uses.



In order to visualize the AWS Shared Responsibility Model, let's take a look at an example.

Let's say a customer is using Amazon S3 for storage and Amazon Workspaces for desktop and application streaming. They also have a Virtual Private Cloud (VPC), which consists of their Amazon EC2 instance and Oracle database instance.

The customer is responsible for managing the guest operating system, the applications, and security groups. AWS is responsible for protecting the global infrastructure that runs all of the services offered in the AWS cloud. The AWS global infrastructure is designed and managed according to security best practices as well as a variety of security compliance standards.

AWS products that fall into the category of Infrastructure as a Service (IaaS), such as Amazon EC2 and Amazon VPC, are completely under your control and require you to perform all the necessary security configuration and management tasks. For example, for EC2 instances, you are responsible for management of the guest OS, which includes updates and security patches, any application software or utilities you install on the instances, and the configuration of the AWS-provided firewall, also called a security group, on each instance. These are basically the same security tasks that you are used to performing no matter where your servers are located.

Amazon RDS database instances like Oracle are pre-configured with parameters and settings appropriate for the engine and class you have selected. AWS manages time-consuming database administration tasks including provisioning, backups, software patching, monitoring, and hardware scaling. You focus on the tasks need for application development.

Amazon WorkSpaces is a fully managed, secure Desktop-as-a-Service (DaaS) solution. You provision virtual, cloud-based Microsoft Windows desktops for your users, providing them access to the documents, applications, and resources they need anywhere, anytime, and from any supported device.

It is best practice for customers to protect their AWS account credentials and set up individual user accounts with IAM so that each user has their own credentials.

Shared Responsibility Summary



- 📌 AWS and the customer share security responsibilities:
 - 📌 AWS is responsible for security *of* the cloud
 - 📌 Customer is responsible for security *in* the cloud
- 📌 Customers have full control over security measures they choose to implement.
- 📌 Customers can use AWS Service Catalog to manage catalogs of IT services.
- 📌 Security configurations of infrastructure services are completely under the customer's control.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Let's quickly review the key points about the shared responsibility model:

- The Shared Responsibility Model consists of AWS and the customer working together to secure data in the cloud. AWS is responsible for security *of* the cloud, while the customer is responsible for security *in* the cloud.
- Customers have full control of what security measures they choose to implement, given the AWS services they are using.
- Customers can use **AWS Service Catalog** to create and manage catalogs of IT services that have been approved for use on AWS.
- AWS products that fall into the category of IaaS, such as Amazon EC2 and Amazon VPC, are completely under the customer's control and that requires them to perform all of the necessary security configuration and management tasks.
- By applying the Shared Responsibility Model, AWS and its customers can ensure secure and compliant data.



Introducing Part: AWS Identity and Access Management (or IAM).

Core AWS Services: IAM



Amazon VPC



Amazon EC2



Amazon S3



Amazon EBS



Amazon EFS



Amazon Glacier



Amazon RDS



Amazon DynamoDB

Storage



IAM

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

IAM allows you to control access to compute, storage, database, and application services in the AWS cloud. This is known as **authentication** and how they can use resources is known as **authorization**. IAM uses access control concepts you will already be familiar with such as users, groups, and permissions so you can specify which users get to access which services.

IAM


Centrally **manage access and authentication** of your users to your AWS resources.

- 📌 Offered as a feature of your AWS account for no charge.
- 📌 Create **users, groups**, and **roles**, and attach **policies** to them to control their access to AWS resources.
- 📌 Manage what resources can be accessed by who and how they can be accessed (e.g., terminating EC2 instances).
- 📌 Define required credentials based on context (e.g., **who** is accessing **which service** and **what** are they trying to do?).







© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

IAM is a tool to centrally manage access to launching, configuring, managing, and terminating your resources. It allows extremely granular control over access permissions, not just based on resource, but all the way down to determining exactly which **Application Programming Interface (API)** calls for each service can be made.

Think of the access control concepts you're already familiar with such as users (think of these as your end users), groups (think of these as a collection of users by job function), permissions, which can be applied to users or groups, and roles (think of these as trusted entities). That's exactly what IAM uses which makes it so powerful!

Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources. IAM allows you to:

- **Manage IAM users and their access** – You can create users in IAM, assign them individual security credentials (in other words, access keys, passwords, and multi-factor authentication devices), or request temporary security credentials to provide users access to AWS services and resources. You can manage permissions in order to control which operations a user can perform.
- **Manage IAM roles and their permissions** – You can create roles in IAM and manage permissions to control which operations can be performed by the entity, or AWS service, that assumes the role.
- **Manage federated users and their permissions** – You can enable identity federation to allow

existing identities (users, groups, and roles) from your corporate directory to access the AWS Management Console, call AWS APIs, and access resources, without the need to create an IAM user for each identity.

You can manage what resources can be accessed by who and how they can be accessed, such as terminating EC2 instances and define required credentials based on context, including who is accessing which service and what they are trying to do.

Types of Security Credentials



Email address and password	Associated with your AWS account (root).
IAM user name and password	Used for accessing the AWS Management Console.
Access and Secret Access keys	Typically used with Command Line Interface (CLI) and programmatic requests like APIs and Software Development Kit (SDKs).
Multi-Factor Authentication	Extra layer of security. Can be enabled for root account and Identity and Access Management (IAM) users.
Key pairs	Used only for specific AWS services like Amazon EC2.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

You use different types of security credentials depending on how you interact with AWS. For example, to sign in to the console, you use a user name and password. In contrast, to make programmatic calls to AWS Application Programming Interface (API) actions, you use access keys. The table on the slide summarizes the different types of AWS security credentials and when you might use each one.

Root Account Access vs. IAM Access

Root Account

- Privileges cannot be controlled
- Full access to all resources

IAM

- Integration with other AWS services
- Identity federation
- Secure access for applications
- Granular permissions

- 🔑 IAM allows you to follow the **least privilege** principle.
- 🔑 **Best practices:**
 - 🔑 Delete root user access keys.
 - 🔑 Create an IAM user.
 - 🔑 Grant administrator access.
 - 🔑 Enable Multi-Factor Authentication (MFA).
 - 🔑 Use IAM credentials to interact with AWS.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the **AWS account root user** and is accessed by signing in with the email address and password that you used to create the account. AWS root accounts have **full** access to all resources in the account and you cannot control the privileges of the root account credentials. Therefore, AWS strongly recommends that you not use root account credentials for day-to-day interactions with AWS.

Use IAM to create additional users and assign permissions to these users, following the least privilege principle. With IAM, you can securely control access to AWS services and resources for users in your AWS account. For example, if you require administrator-level permissions, you can create an IAM user, grant that user full access, and then use those credentials to interact with AWS. Later, if you need to revoke or modify your permissions, you can delete or modify any policies that are associated with that IAM user.

Additionally, if you have multiple users that require access to your AWS account, you can create unique credentials for each user and define who has access to which resources. In other words, you don't need to share credentials. For example, you can create IAM users with read-only access to resources in your AWS account and distribute those credentials to users that require read access.

IAM: Authentication



- 📦 Programmatic access:
 - 📦 Authenticates access key ID and secret access key.
 - 📦 Provides access to API, CLI, SDK, and other development tools.
- 📦 Console access:
 - 📦 Uses account ID or alias, IAM user name and password.
 - 📦 If enabled, MFA prompts for authentication code.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

When adding users, you get to select how users will access AWS. There are two different types of access you can assign users: Programmatic access and AWS Management Console access.

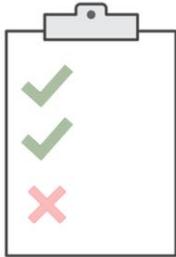
Programmatic access enables an access key ID and secret access key for the AWS API, Command Line Interface (CLI), SDK, and other development tools.

Another option is to give the user console access, which allows them to sign in to the console. The console provides a simple web interface for AWS. You can log in using your AWS account name and password. If you've enabled Multi-Factor Authentication (MFA), you will be prompted for your device's authentication code.

IAM: Authorization



- 🔑 Allows users to access AWS services by granting authorization
- 🔑 Assign permissions by creating an IAM policy.
- 🔑 Permissions determine **which resources and operations** are allowed to be used:
 - 🔑 All permissions are implicitly denied by default.
 - 🔑 If something is explicitly denied, it can never be allowed.



Best practice: Follow the least privilege principle.

NOTE: IAM is **global**. It is not on a per region basis. It applies across all regions.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

After a user has been authenticated, they then must be authorized to access an AWS service.

In order to assign permission to a user, group or role, you have to create an IAM policy, which is a document that explicitly lists permissions. There are no default permissions. All actions are denied by default (*implicit deny*) unless they are explicitly allowed. Any actions that you didn't explicitly allow are denied. Any actions that you explicitly deny are always denied.

The principle of least privilege is an important concept in computer security, promoting minimal user profile privileges based on users' job necessities. When you create IAM policies, follow the standard security advice of granting *least privilege*, that is, granting only the permissions required to perform a task. Determine what users need to do and then craft policies for them that let the users perform *only* those tasks. Start with a minimum set of permissions and grant additional permissions as necessary. Doing so is more secure than starting with permissions that are too lenient and then trying to tighten them later.

Note that IAM is global. It is not on a per region basis. It applies across all regions.

IAM Multi-Factor Authentication (MFA)

aws academy

- MFA provides increased security.
- In addition to user name and password, MFA requires a unique authentication code to access AWS services.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS services and resources can be accessed using the console, CLI, or through SDKs and APIs from a wide range of supported environments. For increased security, we recommend enabling Multi Factor Authentication (MFA). With MFA, users and systems have to be authenticated before they can access AWS services and resources. There are two options for authentication devices: hardware devices and virtual MFA-compliant applications (Google Authenticator or Authy 2-Factor Authentication). SMS is another authentication alternative where you use your mobile device that can receive Short Message Service (SMS) messages to receive a code.

The AWS Security Token Service (STS) is a web service that also enables you to request temporary, limited-privilege credentials for IAM users or for users that you authenticate.

Visit the link to learn more.

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_mfa.html.

IAM Users



- 📦 An entity you create in AWS.
- 📦 Provides a way to interact with AWS.
- 📦 No default security credentials for IAM users:
 - 📦 You have to assign them specifically.
- 📦 IAM users are not necessarily people.



Julie@email.com
Password



Account

John

Jessica

DevApp1

Best practice: Create a separate IAM user account with administrative privileges instead of using the root account user.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

An IAM **user** is an entity that you create in AWS that provides a way to interact with AWS. An IAM user primarily gives people you work with identities that they can use to sign in to the console and make requests to AWS services.

Newly created IAM users have no default credentials to use to authenticate themselves and access AWS resources. You first need to assign security credentials to them for authentication and then attach permissions authorizing them to perform any AWS actions or to access any AWS resources. The credentials you create for users are what they use to uniquely identify themselves to AWS.

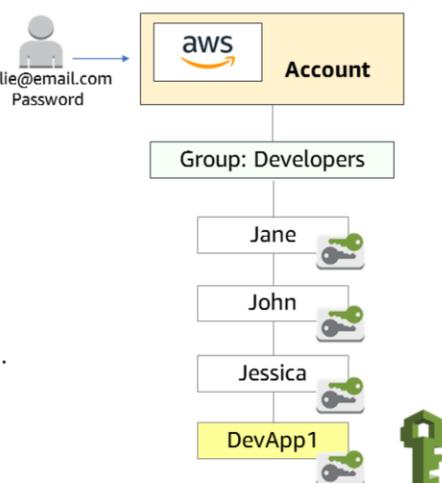
An IAM user is really just an identity with associated permissions. You might create an IAM user to represent an application that must have credentials in order to make requests to AWS. An application might have its own identity in your account and its own set of permissions, the same way that processes have their own identities and permissions in an operating system like Windows or Linux.

A best practice is to create a separate IAM user account with administrative privileges instead of using the root account user.

IAM Groups



- ❏ Collection of IAM users.
- ❏ Specify permissions for the entire group. Julie@email.com Password
- ❏ No default groups.
- ❏ Groups cannot be nested.
- ❏ A user can belong to multiple groups.
- ❏ Permissions are defined using IAM policies.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

A **group** is a collection of IAM users. Groups let you specify permissions for a collection of users, which can make it easier to manage the permissions for those users. For example, you could have a group called **Developers** and give that group the types of permissions that developers typically need. Any user in that group automatically has the permissions that are assigned to the group. If a new user joins your organization and should have developer privileges, add that user to the Developers group. That automatically gives them the appropriate permissions. Similarly, if a person changes jobs in your organization, instead of editing that user's permissions, you can remove him or her from the old group and add him or her to the new group.

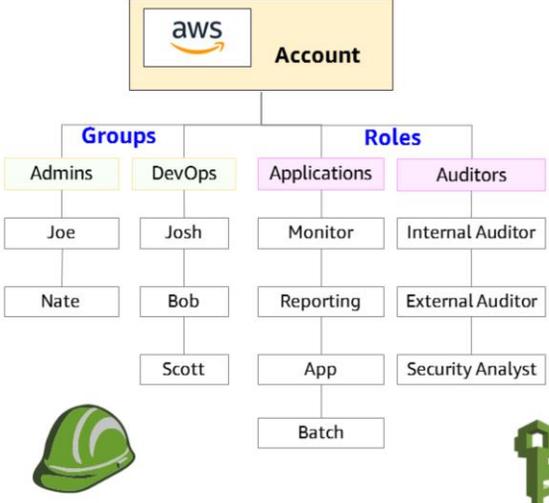
Important characteristics of groups:

- A group can contain many users, and a user can belong to multiple groups.
- Groups can't be nested; they can contain only users, not other groups.
- There's no default group that automatically includes all users in the AWS account. If you want to have a group like that, you need to create it and assign each new user to it.

IAM Roles



- Used to **delegate** access to AWS resources:
 - Provides temporary access.
 - Eliminates the need for static AWS credentials.
- Permissions are:
 - Defined using IAM policies.
 - Attached to the role, not to an IAM user or group.



```

graph TD
    Account[aws Account] --> Groups[Groups]
    Account --> Roles[Roles]
    Groups --> Admins[Admins]
    Groups --> DevOps[DevOps]
    Roles --> Applications[Applications]
    Roles --> Auditors[Auditors]
    Admins --> Joe[Joe]
    Admins --> Nate[Nate]
    DevOps --> Josh[Josh]
    DevOps --> Bob[Bob]
    DevOps --> Scott[Scott]
    Applications --> Monitor[Monitor]
    Applications --> Reporting[Reporting]
    Applications --> App[App]
    Applications --> Batch[Batch]
    Auditors --> InternalAuditor[Internal Auditor]
    Auditors --> ExternalAuditor[External Auditor]
    Auditors --> SecurityAnalyst[Security Analyst]
  
```

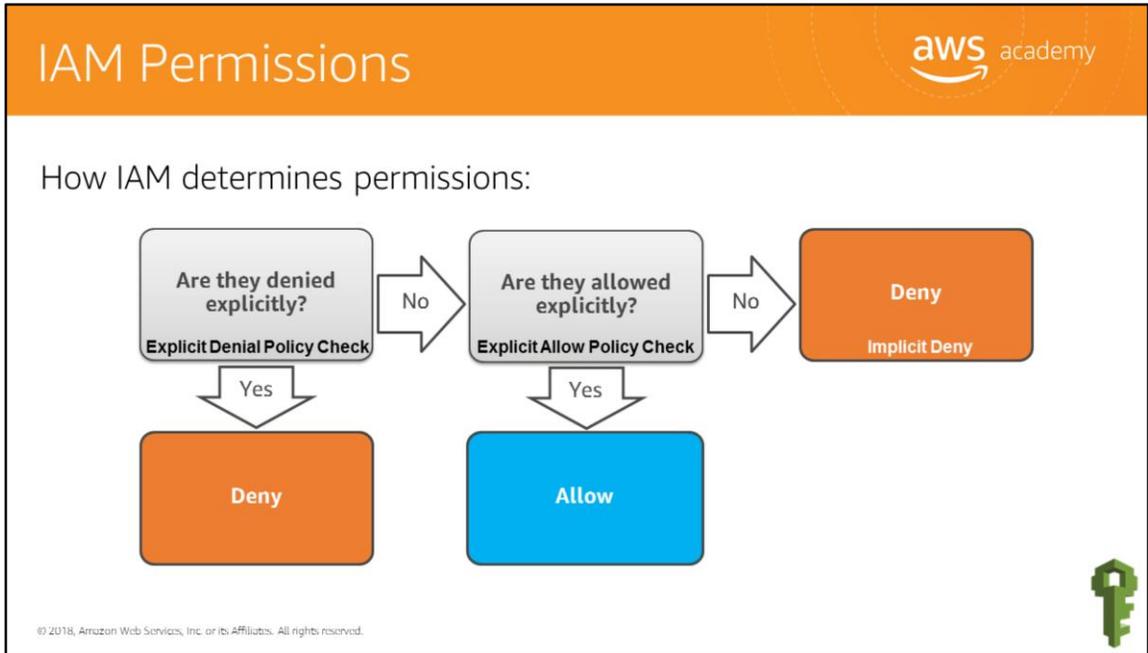
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Federated users do not have permanent identities in your AWS account the way that IAM users do. To assign permissions to federated users, you can create an entity referred to as a **role**.

A role lets you define a set of permissions to access the resources that a user or service needs, but the permissions are not attached to an IAM user or group. Instead, at run time, applications or AWS services can programmatically assume a role. When a role is assumed, AWS returns temporary security credentials that the user or application can use to make programmatic requests to AWS. Consequently, you don't have to share long-term security credentials, for example, by creating an IAM user, for each entity that requires access to a resource.

You create a role in the AWS account that contains the resources that you want to allow access to. When you create the role, you specify two policies:

- The **trust** policy specifies who is allowed to assume the role, such as the trusted entity or principal.
- The **access** (or **permissions**) policy defines what actions and resources the principal is allowed access to. The principal can be an AWS account, an AWS service, such as Amazon EC2, a SAML provider, or an Identity Provider (IdP) that may include Login with Amazon, Facebook, or Google. The principal can also be an IAM user, group, or role from other AWS accounts, including the ones not owned by you.



Policies give you the opportunity to fine-tune privileges granted to IAM users, groups, and roles. Because policies are stored in JSON format, they can be used in conjunction with a version control system. It's a good idea to define least-privilege access to each user, group, or role. That way, you can customize access to specific resources using an authorization policy.

When determining whether permission is allowed, IAM first checks for an **explicit denial policy**. If one does not exist, it then checks for an **explicit allow policy**. If neither an explicit deny or explicit allow policy exists, IAM reverts to the default: **implicit deny**.

IAM Policies



An IAM policy is a formal statement of one or more permissions:

- 📁 You attach a policy to any IAM entity, such as user, group, or role.
- 📁 Policies authorize the actions that may, or may not, be performed by the entity that enables fine-grained access control.
- 📁 A single policy can be attached to multiple entities.
- 📁 A single entity can have multiple policies attached to it.



Best practice: When attaching the same policy to multiple IAM users, put the users in a group and attach the policy to the group instead.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

An IAM policy is a formal statement of one or more permissions. Policies can be attached to any IAM entity, which includes a user, group, role, or resource. For example, you can attach a policy to your AWS resources to block all requests that do not come from an approved Internet Protocol (IP) address range. Policies specify what actions are allowed, which resources to allow the actions on, and what the effect will be when the user requests access to the resources.

The order in which the policies are evaluated has no effect on the outcome of the evaluation. All policies are evaluated, and the result is always that the request is either allowed or denied. When there is a conflict, the most restrictive policy wins.

There are two types of IAM policies. **Identity-based policies** are permission policies that you can attach to a principal, or identity, such as an IAM user, role or group. These policies control what actions that identity can perform, on which resources, and under what conditions. Identity-based policies can be further categorized as:

- **Managed policies:** Standalone identity-based policies that you can attach to multiple users, groups, and roles in your AWS account
- **Inline policies:** Policies that you create and manage and that are embedded directly into a single user group or role.

Resource-based policies are JSON policy documents that you attach to a resource, such as an Amazon S3 Bucket. These policies control what actions a specified principal can perform on that resource and under what conditions. Resource-based policies are inline policies. There are not managed resource-based policies. For more information, select the link.

https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html

When attaching the same policy to multiple IAM users, put the users in a group and attach the policy to the group instead. Additionally, you can use the IAM policy simulator to test and troubleshoot IAM and resource-based policies. Select the link to learn more about the IAM policy simulator.

https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_testing-policies.html

IAM Policy Example



```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": ["DynamoDB:*", "s3:*"],
    "Resource": [
      "arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",
      "arn:aws:s3::bucket-name",
      "arn:aws:s3::bucket-name/*"
    ]
  },
  {
    "Effect": "Deny",
    "Action": ["dynamodb:*", "s3:*"],
    "NotResource": [
      "arn:aws:dynamodb:region:account-number-without-hyphens:table/table-name",
      "arn:aws:s3::bucket-name",
      "arn:aws:s3::bucket-name/*"
    ]
  }
]
}

```

Explicit allow gives users access to a specific DynamoDB table and...

...Amazon S3 buckets.

Explicit deny ensures that the users cannot use any other AWS actions or resources other than that table and those buckets.

An explicit deny statement **takes precedence** over an allow statement.

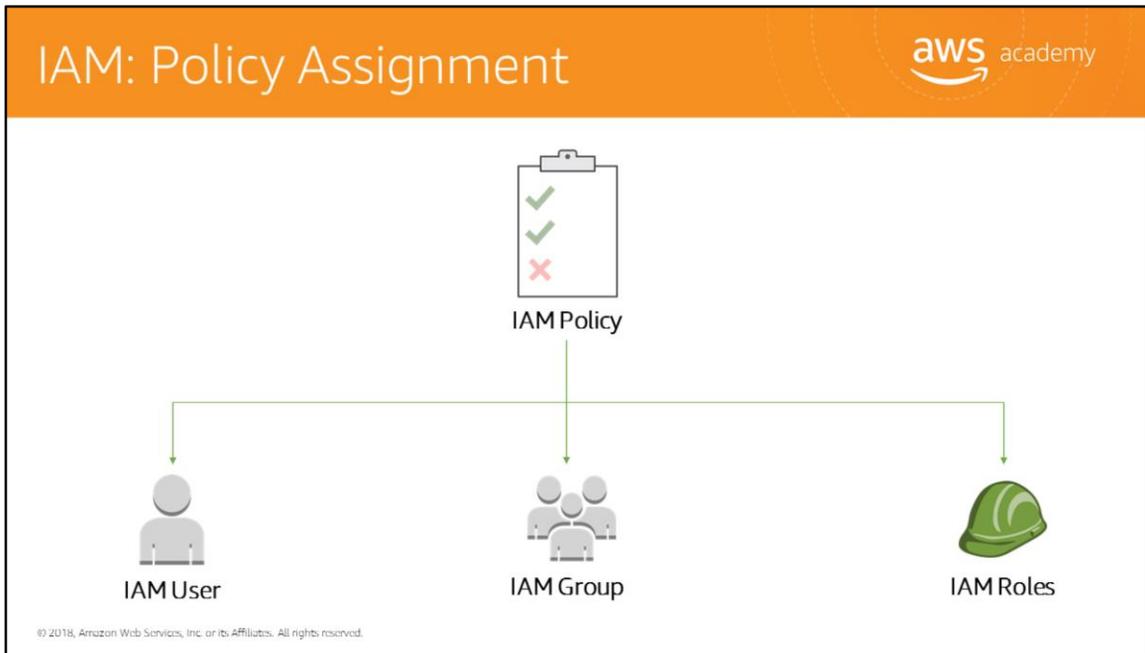


© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

The policy gives users access to only the following:

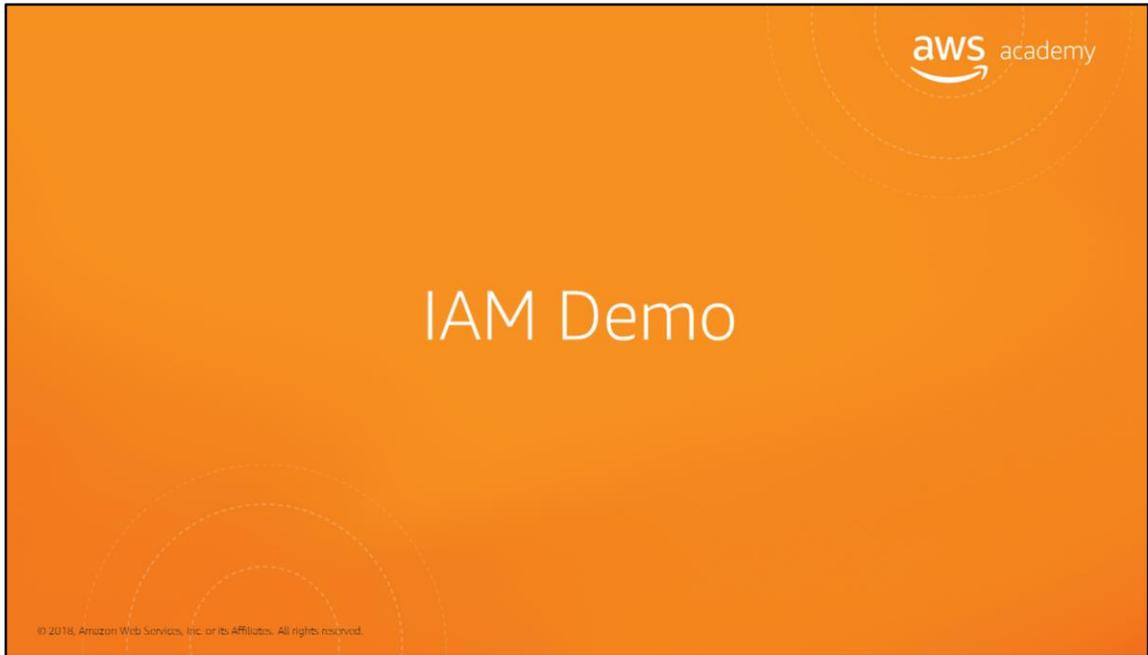
- The DynamoDB table whose name is represented by table-name.
- The AWS account's corporate Amazon S3 bucket, whose name is represented by bucket-name and all the objects that it contains.

The policy includes an explicit deny ("Effect": "Deny") element. In conjunction with the **NotResource** element, which helps to ensure that the users can not use any AWS actions or resources except those specified in the policy, even if permissions have been granted in another policy. An explicit deny statement takes precedence over an allow statement.

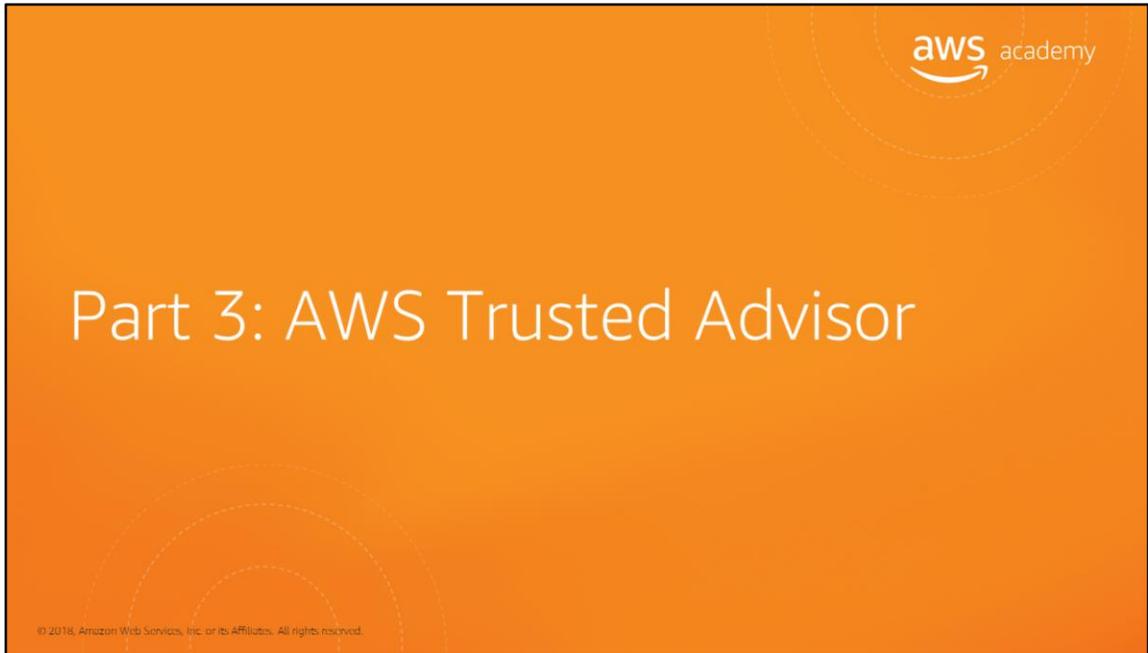


It's important to note that one policy can be assigned to an IAM user, IAM group, and IAM roles.

Now that we have covered the basic concepts of IAM, let's log in to the console, create a user, assign the user to a group, and apply permissions.



Now, please take a moment to review the IAM console demonstration located in the learning management system.



Now let's look at some additional service that can be used to improve overall security and compliance.

AWS Trusted Advisor is like your customized cloud expert. It provides four of the most popular performance and security recommendations to all AWS customers. Let's look at details and a case study to understand this service.

Introduction to Trusted Advisor



AWS Trusted Advisor provides best practices, or checks, in five categories:

Cost Optimization	Performance	Security	Fault Tolerance	Service Limits
				
0  9  0 	3  7  0 	2  4  11 	0  15  5 	37  0  1 
\$7,516.85 Potential monthly savings				

 Green: No problem detected.  Yellow: Investigation recommended.  Red: Action recommended.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Trusted Advisor is an online resource to help you reduce cost, increase performance, and improve security by optimizing your AWS environment. It provides best practices, or checks, in five categories including:

- 1. Cost Optimization:** See how you can save money on AWS by eliminating unused and idle resources or making commitments to reserved capacity.
- 2. Performance:** Improve the performance of your service by checking your service limits, ensuring you take advantage of provisioned throughput, and monitoring for over-utilized instances.
- 3. Security:** Improve the security of your application by closing gaps, enabling various AWS security features, and examining your permissions.
- 4. Fault Tolerance:** Increase the availability and redundancy of your AWS application by taking advantage of automatic scaling, health checks, multiple Availability Zones, and backup capabilities.
- 5. Service Limits:** Checks for service usage that is more than 80% of the service limit.

The status of the check is shown by using color coding on the dashboard page:

- **Red:** Action is recommended
- **Yellow:** Investigation is recommended
- **Green:** No problem has been detected

You can visit the Trusted Advisor Console by selecting the link.

<https://console.aws.amazon.com/trustedadvisor/>

Using AWS Trusted Advisor



- 📌 Six best practices available to all customers:
 1. Service Limits
 2. Security Groups – Specific Ports Unrestricted
 3. IAM Use
 4. Multi-Factor Authentication (MFA) on Root Account
 5. Elastic Block Store (EBS) Public Snapshots
 6. Relational Database Service (RDS) Public Snapshots

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Trusted Advisor provides popular performance and security recommendations to all AWS customers. These six Trusted Advisor checks are available to all customers at no cost:

1. Service Limits
2. Security Groups – Specific Ports Unrestricted
3. IAM Use
4. Multi-Factor Authentication (MFA) on Root Account
5. Elastic Block Store (EBS) Public Snapshots
6. Relational Database Service (RDS) Public Snapshots

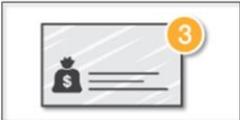
The complete set of checks and guidance is available with Business and Enterprise Support plans. AWS Trusted Advisor helps you to provision your resources following best practices to improve system performance and reliability, increase security, and look for opportunities to save money.

For more information about Trusted Advisor best practices, select the link.
<https://aws.amazon.com/premiumsupport/trustedadvisor/best-practices/>.

Trusted Advisor Features and Functionalities

AWS Trusted Advisor provides a suite of features for you to customize recommendations and to proactively monitor your AWS resources.

Notifications



Access Management



AWS Support API



Action Links



Recent Changes



Exclude Items



5-Min Refresh



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Trusted Advisor provides a suite of features for you to customize recommendations and to proactively monitor your AWS resources including:

- **Trusted Advisor Notifications** allows you stay up-to-date with your AWS resource deployment. You will be notified by a weekly email when you opt in for this service when you opt in for this free service.
- You can use **IAM** to control access to specific checks or check categories.
- You can retrieve and refresh Trusted Advisor results programmatically using the **AWS Support API**.
- **Action Links** are hyperlinks on items within a Trusted Advisor report that take you directly to the console, where you can take action on the Trusted Advisor recommendations.
- With the **Recent Changes** feature, you can track recent changes of check status on the console dashboard. The most recent changes appear at the top of the list to bring them to your attention.
- The **Exclude Items** feature allows you to customize the Trusted Advisor report. You can exclude items from the check result if they are not relevant.
- You can refresh individual checks or refresh all the checks at once by clicking the Refresh All button in the summary dashboard. A check is eligible for **5-Minute Refresh** after it was last refreshed.

For more information about Trusted Advisor select the link.

<https://aws.amazon.com/premiumsupport/trustedadvisor/>.

Hungama Uses AWS Trusted Advisor to Optimize Usage and Cut Costs



Using AWS Trusted Advisor helped us save 33% on our monthly bill, and we'll continue to use it to optimize our infrastructure and costs on AWS.

Amit Vora
CTO, Hungama Digital Media



Hungama is a leading aggregator, developer, publisher and distributor of Bollywood and South-Asian entertainment content.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

- ❏ Hungama has used AWS for server and storage management since 2008.
- ❏ They deliver content to consumers in 47 countries across mobile, Internet, and Internet Protocol Television (IPTV) services.
- ❏ The company uses Amazon S3 to host more than 60 TB of content and Amazon EC2 and Amazon RDS for server and storage management.
- ❏ As the company grew rapidly, more departments used AWS for development, causing an increase in monthly costs.

Let's review a case study by Hungama, who has used AWS for server and storage management since 2008, saving 33% on their monthly bill. They deliver content to consumers in 47 countries across mobile, Internet and Internet Protocol Television services. They use Amazon S3 to host more than 60 terabytes of content and Amazon EC2 and Amazon RDS for server and storage management. As the company grew rapidly, more departments used AWS for development, causing an increase in monthly costs.

For more information on how Hungama uses AWS, select the link.
<https://aws.amazon.com/solutions/case-studies/hungama/>.

Hungama Uses AWS Trusted Advisor to Optimize Usage and Cut Costs



"

Using AWS Trusted Advisor helped us save 33% on our monthly bill, and we'll continue to use it to optimize our infrastructure and costs on AWS.

Amit Vora
CTO, Hungama Digital Media



"

Hungama is a leading aggregator, developer, publisher and distributor of Bollywood and South-Asian entertainment content.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Hungama reduced monthly costs by 33% by using Trusted Advisor's Cost Optimizing checks:

- ❏ Revealed over-provisioned instance sizes, and instances spun up for special projects not terminated after completion.
- ❏ Identified additional opportunities for optimization of the Reserved Instances (RIs) they had purchased.
- ❏ Identified a number of unused or underutilized EBS volume that were leftover from previous test projects.

Three AWS Trusted Advisor checks were particularly helpful in optimizing usage and cutting costs.

The **Low Utilization Amazon EC2 Instances** check revealed over-provisioned instance sizes and instances spun up for special projects that were not terminated after completion. In response, the audit team used this information to **right-size** their instances. They also **categorized** production and development servers and **automated** the process of shutting down development servers during non-business hours.

The **Reserved Instance Optimization** check identified additional opportunities for optimization of the RI instances they had purchased. In response, Hungama changed how they reserved their instances and based reservations on the **specific usage patterns** of their different instance categories (dev/prod/test/etc.).

The **Underutilized Amazon EBS** volumes check identified a number of unused or underutilized EBS volumes that were often leftover from previous test projects. In response, the audit team created **snapshots** of many of the underutilized EBS volumes, which they stored on Amazon S3, and then **deleted the volumes**. This resulted in a reduction of over 90% on the number of snapshots generated weekly.

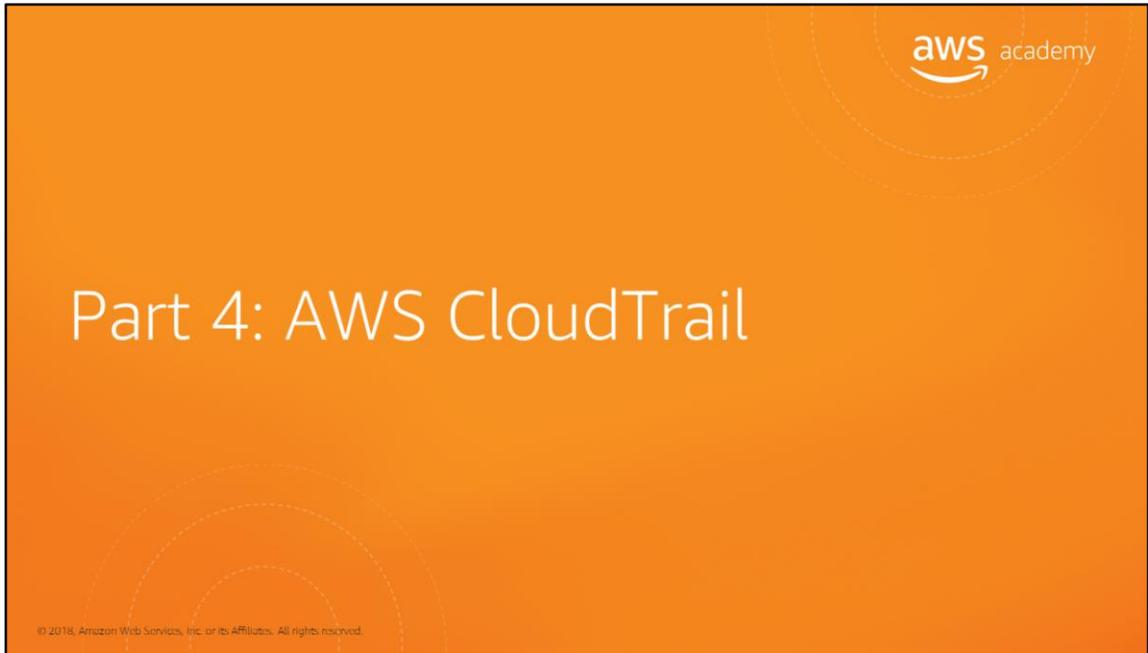
In Review



- 📦 Trusted advisor is a customized cloud expert:
 - 📦 Helps you follow best practices
 - 📦 Inspects your AWS environment
 - 📦 Helps close security gaps
- 📦 Finds opportunities and best practices in:
 - 📦 Cost optimization
 - 📦 Performance
- 📦 Security:
 - 📦 Fault tolerance
 - 📦 Service limits

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

In summary, AWS Trusted Advisor is an online tool that acts like a customized cloud expert, helping you to configure your resources to follow best practices. Trusted Advisor inspects your AWS environment to help close security gaps, and finds opportunities to save money, optimize costs, improve system performance, and increase reliability. This increases security with fault tolerance and service limits.



Introducing part 4: AWS CloudTrail.

AWS CloudTrail enables you to simplify governance, compliance, and risk auditing. CloudTrail accelerates analysis of operational and security issues by providing visibility into both API and non-API actions in your AWS account.

Introduction to CloudTrail

aws academy

CloudTrail is a web service that records API calls for your account and delivers log files to you.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

CloudTrail is a web service that records API calls for your account and delivers log files to you.

CloudTrail is a crucial tool for simplifying governance, compliance, and risk auditing. Everything in AWS is an API call and CloudTrail logs the API calls made in an AWS account across regions whether that action was performed via the CLI, an SDK, the console, or an API directly. This includes actions like starting and stopping instances, creating or modifying Amazon RDS databases, or uploading a file to Amazon S3. This logging accelerates analysis of operational and security issues by providing visibility into actions in your AWS account.

Amazon CloudWatch is a monitoring service for AWS cloud resources that can be used to collect and track metrics, collect and monitor log files, set alarms, and automatically react to AWS resource changes. Integrating CloudTrail with Amazon CloudWatch provides comprehensive, secure, and searchable event history of activities made by the console, AWS SDKs, command line tools, and other AWS services.

AWS CloudTrail Benefits



User and Resource Activity



Simplified Compliance



Always On



Security Automation



Analysis and Troubleshooting

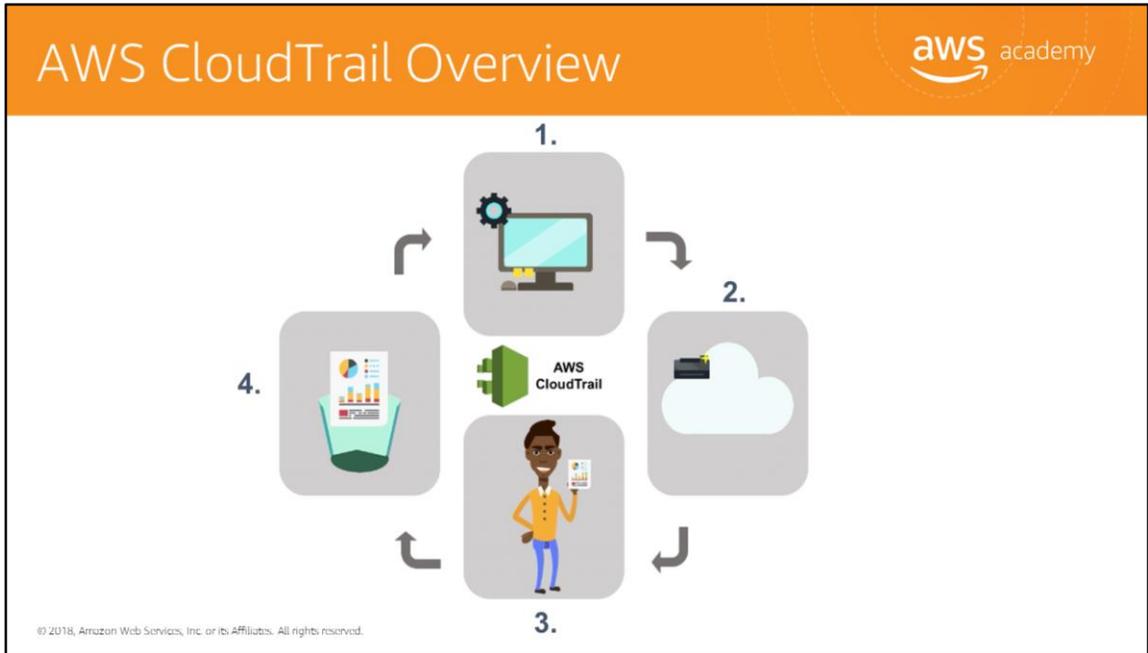
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

CloudTrail has several key benefits!

It increases your visibility into user and resource activity that allows you to identify who did what and when in your account.

Compliance audits are simplified because they are automatically recording and storing event logs. This allows you to search through log data, identify actions that are out of compliance, accelerate investigations into incidents, and then expedite a response.

Because you are able to capture a comprehensive history of changes made within your account, you can analyze and troubleshoot any operational issues in your account.



How does this work?

First, an activity happens in your account. Next, CloudTrail captures and records that activity and calls it a **CloudTrail event**. The event will contain details about who performed the request, the date and time of the request, the source IP and how the request was made, the action performed, the region in which the action was taken, and the response. By default, the logs are stored for 7 days. The activity log can be sent to other AWS services, so the activity history can be retained for as long as you like.

Using CloudTrail Best Practices



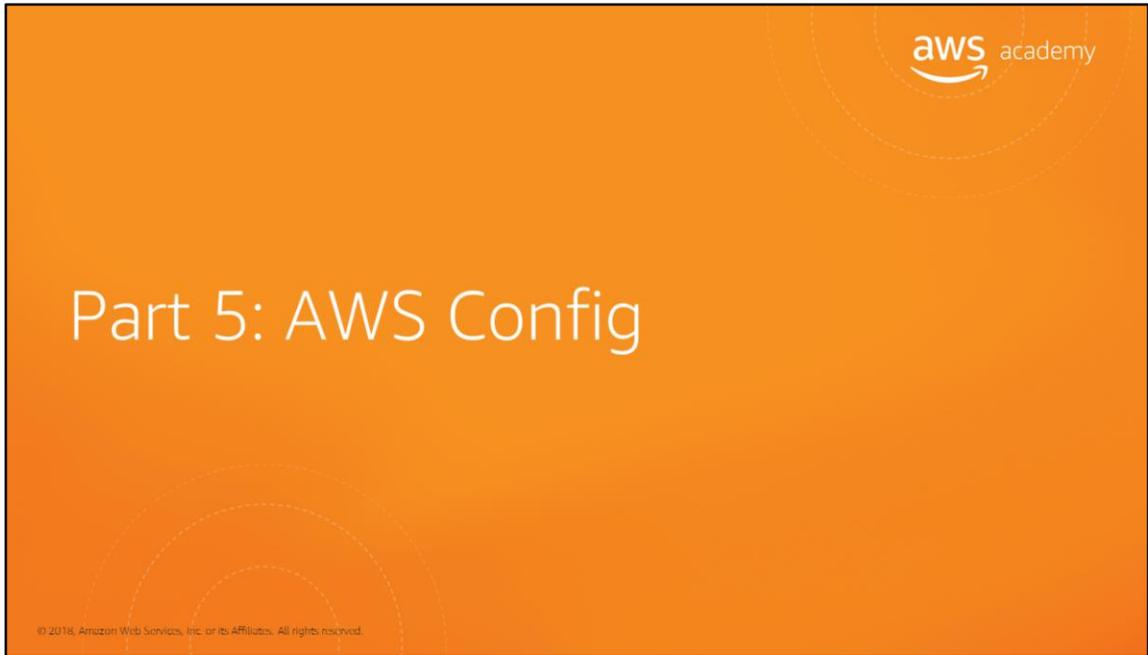
- 📁 Turn on CloudTrail log file validation.
- 📁 Aggregate log files to a single Amazon S3 bucket.
- 📁 Ensure that it is enabled across AWS globally.
- 📁 Restrict access to CloudTrail Amazon S3 buckets.
- 📁 Integrate with Amazon CloudWatch.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

To get the most out of CloudTrail, turn on CloudTrail log file validations. When you are configuring CloudTrail, you can aggregate all log files to a single Amazon S3 bucket.

Additionally, a configuration that applies to all regions ensures that your settings are applied consistently across all existing and newly launched regions.

You can also validate the integrity of log files by detecting whether or not they were changed or deleted after they were sent to the S3 bucket. It is also a good idea to run MFA to delete a CloudTrail bucket. This can be accomplished by restricting access to where they are stored. Lastly, integrating this service with Amazon CloudWatch enables you to define actions to execute when specific events are logged by CloudTrail.



Now, let's look at AWS Config to see how it can help with security.

Introduction to AWS Config



AWS Config is a fully managed service that enables you to assess, audit, and evaluate the configuration of your AWS resources.



- Continuous monitoring
- Continuous assessment
- Change management
- Operation troubleshooting

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Config is a fully managed service that enables you to assess, audit, and evaluate the configuration of your AWS resources. It provides continuous monitoring, continuous assessment, change management and operation troubleshooting.

AWS Config provides you with an AWS resource inventory, configuration history, and configuration change notifications to enable security and governance. With AWS Config, you can discover existing AWS resources, export a complete inventory of your AWS resources with all configuration details, and determine how a resource was configured at any point in time. These capabilities enable compliance auditing, security analysis, resource change tracking, and troubleshooting.

With AWS Config, you can review changes in configurations and relationships between AWS resources. You can also review detailed resource configuration histories and determine overall compliance against those configurations specified in your internal compliance. This enables you to simplify compliance auditing, security analysis, change management, as well as operational troubleshooting.

Track Changes to Resources with AWS Config

- ❏ Provides AWS resource inventory, configuration history, and configuration change notifications.
- ❏ Provides continuous details on all configuration changes associated with AWS resources.
- ❏ Combines with CloudTrail for full visibility into what contributed to a change.
- ❏ Enables compliance auditing, security analysis, resource change tracking, and troubleshooting.

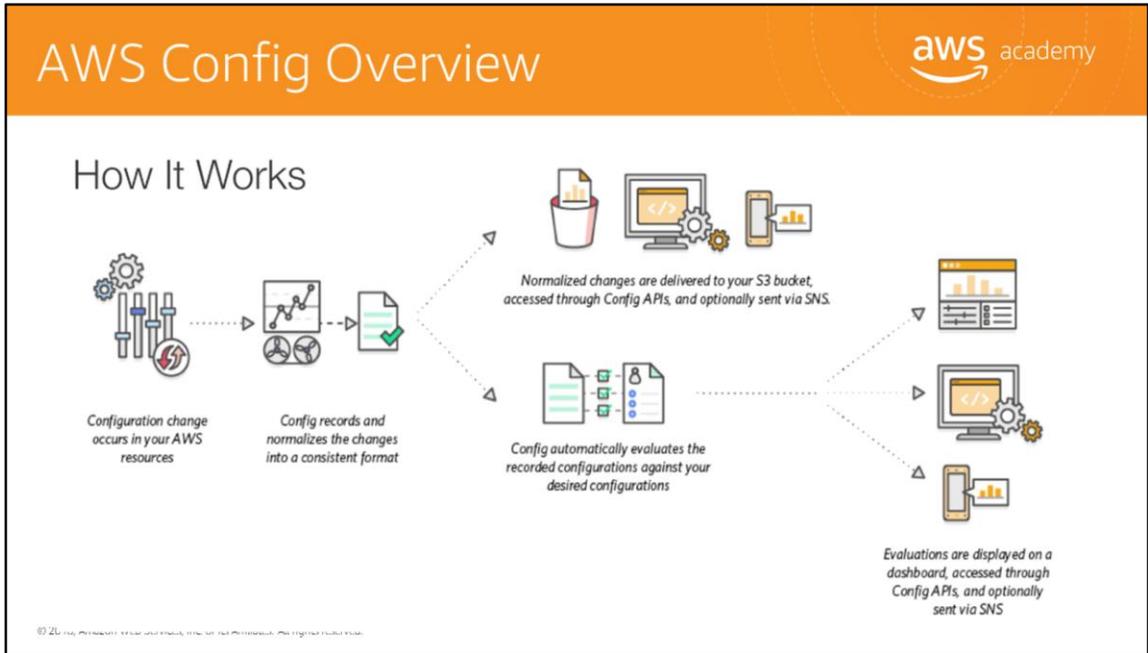
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



AWS Config makes it easy to track your resource's configuration without the need for upfront investments, and you can avoid the complexity of installing and updating agents for data collection or maintaining large databases. After you enable AWS Config, you can view continuously updated details of all configuration attributes associated with AWS resources. You are notified via Amazon Simple Notification Service (Amazon SNS) of every configuration change.

AWS Config gives you access to resource configuration history. You can relate configuration changes with CloudTrail events that may have contributed to the change in configuration. This information provides you full visibility from details, such as "Who made the change?" and "From what IP address?", to the effect of this change on AWS resources and related resources. You can use this information to generate reports to aid auditing and assessing compliance over a period of time.

If you want to track changes to resources configuration, answer questions about resource configurations, demonstrate compliance, troubleshoot, or perform security analysis, use AWS Config.



So, how does this work?

The first thing that happens is a change occurs in one of your AWS resources. Next, the AWS Config engine records and normalizes that change in a consistent format. Then those changes are delivered to an S3 bucket, they are assessed through the AWS Config APIs and, optionally, they can be sent out via a notification service like Amazon SNS.

AWS Config will automatically evaluate the recorded configuration against your desired configuration. Those evaluations will be displayed on the dashboard or they are accessible via the AWS Config APIs. They can also be sent out via Amazon SNS.

AWS Config Summary



Simple setup



Customize rules



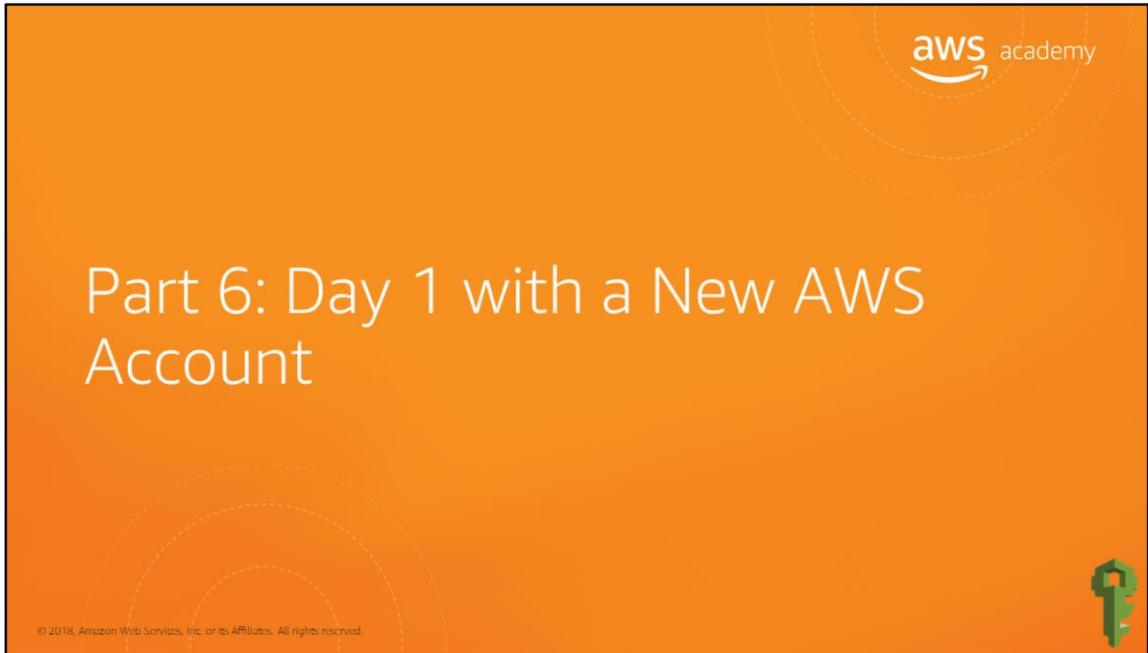
Continuous compliance

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

In summary, AWS Config is a simple service to set up, but it is a very powerful tool that allows you to take advantage of custom rules to automatically discover your AWS resources.

With AWS Config, you can:

- Evaluate your AWS resource configurations for desired settings.
- Get a snapshot of the current configurations of the supported resources that are associated with your AWS account.
- Retrieve configurations of one or more resources that exist in your account.
- Retrieve historical configurations of one or more resources.
- Receive a notification whenever a resource is created, modified, or deleted.
- View relationships between resources. For example, you might want to find all resources that use a particular security group.



Let's take a look at how we put these services into action by starting from the beginning.

This is Day 1 with a new AWS account. What are the best practices for setting up the account?

Day One with AWS



1 Stop using the root account as soon as possible.

The root account has completely unrestricted access to your resources.

To stop using the root account, take the following steps:

- 1) With the root account, create an IAM user for yourself.
- 2) Create an IAM group, give it full administrator permissions, and add the IAM user to the group.
- 3) Sign in with your IAM user credentials.
- 4) Store your root account credentials in a very secure place. Disable and remove your root account access keys, if you have them.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



As we begin, AWS recommends that if you have access keys for your root account, you remove them once you've established that they are not being used anywhere in your applications.

To stop using the root account, take the following steps:

- First, with the root account, create an IAM user for yourself.
- Next, create an IAM group, give it full administrator permissions, and add the IAM user to the group.
- Then sign in with your IAM user credentials.
- Finally, store your root account credentials in a very secure place. Disable and remove your root account access keys, if you have them.

For instructions for setting up your first IAM user and administrators group, select the link.

http://docs.aws.amazon.com/IAM/latest/UserGuide/getting-started_create-admin-group.html.

Day One with AWS



- 2 Require **Multi-Factor Authentication (MFA)** for access.
 - a. Require MFA for your root account and all IAM users.
 - b. You can also use MFA to control access to AWS service APIs.

Software MFA options: AWS Virtual MFA, Google Authenticator, Authy Authenticator (Windows phone app), or SMS notification.

Hardware MFA options: Key fob or display card offered by Gemalto:
<https://safenet.gemalto.com/multi-factor-authentication/>

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



SMS-based Multi-Factor Authentication is currently in preview release only. Select the links to request access and learn more.

<https://aws.amazon.com/iam/details/mfa/smsmfa/>.

<https://aws.amazon.com/iam/details/mfa/>.

Day One with AWS



3 Enable **CloudTrail**.

CloudTrail logs all API requests to resources in your account:

- 1) Via the CloudTrail console: Create a trail, give it a name, apply it to all regions, and enter a name for the new Amazon S3 bucket that the logs will be stored in.
- 2) Ensure that the Amazon S3 bucket you use for CloudTrail has its access restricted to only those who should have access, such as admins.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Let's reveal how to enable CloudTrail.

CloudTrail logs all API requests to resources in your account via the CloudTrail console. Create a trail, give it a name, apply it to all regions, and enter a name for the new Amazon S3 bucket that the logs will be stored in.

Ensure that the Amazon S3 bucket you use for CloudTrail has its access restricted to only those who should have access, such as admins.

CloudTrail is now enabled by default for all customers. It will provide visibility into the past seven days of account activity without the need for you to configure a trail in the service to get started. This new always-on capability provides the ability to view, search, and download the aforementioned account activity through the CloudTrail Event History.

For step-by-step instructions for creating a trail in CloudTrail, select the link.

<http://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-create-a-trail-using-the-console-first-time.html>.

Day One With AWS



- ④ Enable a **billing report**, such as the AWS Cost and Usage Report:
 - a) Billing reports provide information about your usage of AWS resources and estimated costs for that usage.
 - b) AWS delivers the reports to an Amazon S3 bucket that you specify and updates the reports at least once a day.
 - c) The AWS Cost and Usage Report tracks your AWS usage and provides estimated charges associated with your AWS account, either by the hour or by the day.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Now, let's discover how to enable a **billing report**, such as the AWS Cost and Usage Report.

- Billing reports provide information about your usage of AWS resources and estimated costs for that usage.
- AWS delivers the reports to an Amazon S3 bucket that you specify and updates the reports at least once a day.
- The AWS Cost and Usage Report tracks your AWS usage and provides estimated charges associated with your AWS account, either by the hour or by the day.

To create an AWS Cost and Usage report, select the link.

<http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/detailed-billing-reports.html#turnonreports>.

IAM Best Practices Summary



- **Delete** AWS account (root) access keys.
- Create **individual** IAM users.
- **Use groups** to assign permissions to IAM users.
- Grant **least privilege**.
- Configure a **strong password policy**.
- Enable **MFA** for privileged users.
- Use **roles for applications** that run on Amazon EC2 instances.
- Delegate by **using roles** instead of by sharing credentials.
- Use **policy conditions** for extra security.
- **Rotate credentials** regularly.
- **Remove unnecessary** users and credentials.
- **Monitor activity** in your AWS account.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

In summary, displayed are some best practices to follow with IAM.

For more information, select the link.

<http://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html>

In Review



- ❏ AWS can be accessed in three ways:
 - ❏ Via the AWS Management Console
 - ❏ Programmatically (using the CLI)
 - ❏ Using the SDK
- ❏ Root account is the email address used to set up the AWS account and *always* has full administrator access.
 - ❏ These credentials should never be given to anyone.
 - ❏ The AWS Account Root User access keys should be deleted after login.
 - ❏ A user should be created for each individual within the organization.
 - ❏ The root account should always be secured with MFA.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



In summary, AWS can be accessed in three ways:

- Via the AWS Management Console
- Programmatically (using the CLI), and
- Using the SDK

A root account is the email address used to set up the AWS account and *always* has full administrator access.

- These credentials should never be given to anyone.
- The AWS Account Root User access keys should be deleted after login.
- A user should be created for each individual within the organization.
- And the root account should always be secured with multi-factor authentication.

For more information, select the link.

<http://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html>

In Review



- ❏ An IAM **user** is an entity that you create in AWS to represent the person or service that interacts with AWS.
- ❏ An IAM **role** is similar to a user in that it is an AWS identity with permission policies that determine what actions the role can perform and used to delegate access to users.
- ❏ An IAM **group** is a place to store your users:
 - ❏ Identities that represent the user.
 - ❏ Is a simple way to attach policies to multiple users.
- ❏ IAM **policies** are constructed with Java Script Notation (JSON):
 - ❏ Contain key value pairs that contain a name and a value.
 - ❏ Example: {"name": "George Washington"}

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



An IAM **user** is an entity that you create in AWS to represent the person or service that interacts with AWS.

An IAM **role** is similar to a user in that it is an AWS identity with permission policies that determine what actions the role can perform and used to delegate access to users.

An IAM **group** is a place to store your users identities that represent the user. It's a simple way to attach policies to multiple users.

IAM **policies** are constructed with Java Script Notation (or JSON). They contain key value pairs that contain a name and a value, for example, "George Washington."

For more information, select the link.

<http://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html>



Introducing Part 7: AWS Security Compliance Program.

The success of our security and compliance program is primarily measured by our customers' success. Our customers drive our portfolio of compliance reports, attestations, and certifications that support their efforts in running a secure and compliant cloud environment.

You can take advantage of this effort to achieve the savings and security at scale that AWS offers while still maintaining robust security and regulatory compliance.

In this part, we'll be discussing:

- AWS' Compliance Approach, which includes Assurance Programs.
- AWS Risk and Compliance Programs, such as Risk Management, Control Environment, and Information Security.
- AWS Customer Compliance responsibilities.

AWS Service Catalog



-  Ensure compliance with corporate standards
-  Help employees Quickly find and deploy approved IT services
-  Centrally manage IT service lifecycle

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Service Catalog can be integrated with AWS CloudFormation for stack developments to ensure compliance with corporate standards.

To help employees quickly find and deploy approved IT services, customers can use the AWS Service Catalog to create and manage catalogs of IT services that they have approved for use on AWS. This includes virtual machine images, servers, software, and databases to complete multi-tier application architectures.

AWS Service Catalog allows you to centrally manage commonly deployed IT services and helps you achieve consistent governance and meet compliance requirements while enabling users to quickly deploy only the approved IT services they need.

For more information on the AWS Service Catalog, select the link.
<https://aws.amazon.com/servicecatalog/>.

AWS Compliance Approach



- 📦 AWS and customers share control.
- 📦 Responsibility of AWS:
 - 📦 Provide highly secure and controlled environment.
 - 📦 Provide wide array of security features.
- 📦 Responsibility of the customer to configure IT.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

As we discussed in the Shared Security Responsibility Model, AWS and its customers share control over the IT environment, which means both parties have responsibility for managing the IT environment. The responsibility of AWS in this model includes providing its services on a highly secure and controlled environment and providing a wide array of security features for customers to use.

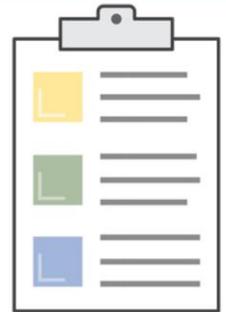
The customers' responsibility includes configuring their IT environments in a secure and controlled manner for their purposes.

AWS Security Information



AWS shares security information by:

- 📁 Obtaining industry certifications.
- 📁 Publishing security and control practices.
- 📁 Providing documentation directly under Non-Disclosure Agreements (NDAs).



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

While customers don't communicate their use and configurations to AWS, AWS does communicate its security and control environment relevant to customers.

AWS does this by:

- Obtaining industry certifications and independent third-party attestations.
- Publishing information about the AWS security and control practices in whitepapers and web site content.
- Providing certificates, reports, and other documentation directly to AWS customers under Non-Disclosure Agreements (NDAs), as required.

AWS Assurance Programs



AWS, certifying bodies, and independent auditors provide:

- 📄 Certifications/attestations
- 📄 Laws, regulations, and privacy
- 📄 Alignments/frameworks



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS engages with external certifying bodies and independent auditors to provide customers with considerable information regarding the policies, processes, and controls established and operated by AWS.

Certifications/Attestations: Compliance certifications and attestations are assessed by a third-party, independent auditor and result in a certification, audit report, or attestation of compliance.

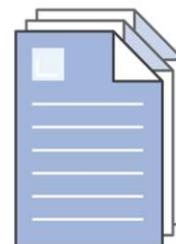
Laws, Regulation, and Privacy: AWS customers remain responsible for complying with applicable compliance laws and regulations. In some cases, AWS offers functionality, such as security features, enablers, and legal agreements, such as the AWS Data Processing Agreement and Business Associate Addendum, to support customer compliance.

Alignments/Frameworks: Compliance alignments and frameworks include published security or compliance requirements for a specific purpose, such as a specific industry or function. AWS provides functionality, such as security features, and that include compliance playbooks, mapping documents, and whitepapers for these types of programs.

AWS Risk and Compliance Programs

AWS Risk and Compliance Programs:

- 📁 Provide information about AWS controls
- 📁 Assist customers in documenting their framework



Components of AWS Risk and Compliance Programs:

- 📁 Risk management
- 📁 Control environment
- 📁 Information Security (IS)

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS provides information about its Risk and Compliance Program to enable customers to incorporate AWS controls into their governance framework. This information can assist customers in documenting a complete control and governance framework with AWS included as an important part of that framework.

The AWS Risk and Compliance Program is made up of three components:

- Risk Management
- Control Environment
- Information Security

Let's take a look at each of the AWS Risk and Compliance Programs in more detail.

AWS Risk Management



- 📁 **Business plan:**
 - 📁 Includes risk management
 - 📁 Plan re-evaluated at least biannually
- 📁 **Responsibilities:**
 - 📁 Identifies risks
 - 📁 Implements appropriate measures to address risks
 - 📁 Assesses various internal/external risks
- 📁 **Information security framework and policies based on:**
 - 📁 Control Objectives for Information and related Technology (COBIT)
 - 📁 American Institute of Certified Public Accountants (AICPA)
 - 📁 National Institute of Standards and Technology (NIST)

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS management has developed a strategic business plan that includes risk identification and the implementation of controls to mitigate or manage risks. AWS management re-evaluates the strategic business plan at least biannually. This process requires management to identify risks within its areas of responsibility and to implement appropriate measures designed to address those risks.

In addition, the AWS control environment is subject to various internal and external risk assessments.

The AWS Compliance and Security teams have established an information security framework and policies that are based on the following governing bodies:

- Control Objectives for Information and related Technology (COBIT)
- American Institute of Certified Public Accountants (AICPA)
- National Institute of Standards and Technology (NIST)

AWS Risk Management



- 📦 **AWS takes care of:**
 - 📦 Maintaining the security policy
 - 📦 Providing security training to employees
 - 📦 Performing application security reviews to assess:
 - 📦 Data confidentiality, integrity, availability
 - 📦 Conformance to IS policy
- 📦 **AWS security**
 - 📦 Scans service endpoints for vulnerabilities
 - 📦 Notifies for remediation of vulnerabilities
- 📦 **Independent security firms**
 - 📦 Scans are not a replacement for customer scans
 - 📦 Customers can ask to scan cloud infrastructure

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS maintains the security policy, provides security training to employees, and performs application security reviews. These reviews assess the confidentiality, integrity, and availability of data, as well as conformance to the information security policy.

AWS Security regularly scans all Internet facing service endpoint IP addresses for vulnerabilities. However, scans are not performed on customer EC2 instance interfaces. AWS Security notifies the appropriate parties to remediate any identified vulnerabilities.

In addition, external vulnerability threat assessments are performed regularly by independent security firms. Findings and recommendations resulting from these assessments are categorized and delivered to AWS leadership. These scans are done in a manner for the health and viability of the underlying AWS infrastructure and are not meant to replace the customer's own vulnerability scans required to meet their specific compliance requirements. Customers can request permission to conduct scans of their cloud infrastructure as long as they are limited to the customer's instances and do not violate the AWS Acceptable Use Policy.

AWS Control Environment



- 📦 Includes policies, processes, and control activities.
- 📦 Secure delivery of AWS service offerings.
- 📦 Control environment encompasses:
 - 📦 People
 - 📦 Processes
 - 📦 Technology
- 📦 Supports the operating effectiveness of the AWS control framework.
- 📦 Integrates controls identified by industry-leading cloud bodies.
- 📦 AWS monitors for leading practice ideas to manage control environment.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS manages a comprehensive control environment that includes policies, processes, and control activities that leverage various aspects of Amazon's overall control environment. This control environment is in place for the secure delivery of AWS service offerings. The collective control environment encompasses the people, processes, and technology necessary to establish and maintain an environment that supports the operating effectiveness of the AWS control framework.

AWS has integrated applicable cloud-specific controls identified by leading cloud computing industry bodies into the AWS control framework. AWS continues to monitor these industry groups for ideas on which leading practices can be implemented to better assist customers with managing their control environment.

Information Security



- 📦 Designed to protect:
 - 📦 Confidentiality
 - 📦 Integrity
 - 📦 Availability
- 📦 Publishes security whitepaper



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS has implemented a formal information security program designed to protect the confidentiality, integrity, and availability of customers' systems and data.

AWS publishes a security whitepaper that is available on the public website that addresses how AWS can help customers secure their data.

To learn more about compliance and find additional resources for this topic, select the link. <https://aws.amazon.com/compliance/>.

Customer Compliance Requirements



- 📦 Maintain governance over the entire IT control environment.
- 📦 Customers should understand:
 - 📦 Required compliance objectives
 - 📦 Validation-based risk tolerance
- 📦 Establish control environment.
- 📦 Verify effectiveness of control environment.
- 📦 Customer compliance basic approach:
 - 📦 Review
 - 📦 Design
 - 📦 Identify
 - 📦 Verify



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS customers are required to continue to maintain adequate governance over the entire IT control environment regardless of how IT is deployed. Leading practices include an understanding of required compliance objectives and requirements (from relevant sources), establishment of a control environment that meets those objectives and requirements, an understanding of the validation required based on the organization's risk tolerance, and verification of the operating effectiveness of their control environment. Deployment in the AWS cloud gives enterprises different options to apply various types of controls and various verification methods.

Strong customer compliance and governance might include the following basic approach:

- **Review** information available from AWS together with other information to understand as much of the entire IT environment as possible, and then document all compliance requirements.
- **Design** and implement control objectives to meet the enterprise compliance requirements.
- **Identify** and document controls owned by outside parties.
- **Verify** that all control objectives are met and all key controls are designed and operating effectively.

By staying engaged in the compliance and governance process with AWS, customers can

ensure compliance requirements are being met.

In Review



AWS security compliance programs

- 📁 Enables customers to understand robust controls to maintain security and data protection
- 📁 Shared compliance responsibilities

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Cloud Compliance enables customers to understand the robust controls in place at AWS to maintain security and data protection in the cloud. As systems are built on top of AWS cloud infrastructure, compliance responsibilities will be shared.

By tying together governance-focused, audit-friendly service features with applicable security compliance regulations or audit standards, AWS Compliance enablers build on traditional programs; helping customers to establish and operate in an AWS security controlled environment.



Introducing Part 8: AWS Security Resources.

As we mentioned before, AWS communicates its security and control environment relevant to customers by doing the following:

- Industry certifications and independent third-party attestations.
- Information about AWS security and control practices in whitepapers and web content.
- Certificates, reports, and other documentation provided directly to AWS customers under NDA.

Let's take a closer look at how AWS provides customers with guidance and expertise through online tools, resources, support, and professional services to secure their data in the cloud.

AWS Account Teams



- 📦 First point of contact
- 📦 Guide deployment
- 📦 Point toward the right resources to resolve security issues



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Account Teams provide a first point of contact that can guide you through your deployment and implementation and point you toward the right resources to resolve security issues that you may encounter.

AWS Enterprise Support*



- 📦 15-minute response time
- 📦 24/7, by phone, chat, or email
- 📦 Dedicated Technical Account Manager (TAM)



*For details, see:

<https://aws.amazon.com/premiumsupport/enterprise-support/>

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Enterprise Support provides 15-minute response time and is available 24x7 by phone, chat, or email; along with a dedicated Technical Account Manager (TAM). This concierge service ensures that customers' issues are addressed as swiftly as possible.

Select the link to learn more.

<https://aws.amazon.com/premiumsupport/enterprise-support/>

AWS Professional Services and AWS Partner Network



AWS Partner Network (APN) is a group of cloud software and service vendors that has hundreds of certified AWS Consulting Partners worldwide.



- 📦 APN have earned endorsement from AWS.
- 📦 Two groups:
 - 📦 APN Consulting Partners:
 - 📦 Help customers implement and manage an AWS cloud deployment.
 - 📦 Help develop security policies.
 - 📦 Help meet compliance requirements.
 - 📦 Include system integrators and managed services providers.
 - 📦 APN Technology Partners:
 - 📦 Provide software tools and services hosted on or integrated with AWS.
 - 📦 Include independent software vendors and providers of Software as a Service (SaaS).



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

The AWS Partner Network is a group of cloud software and service vendors that has hundreds of certified AWS Consulting Partners worldwide to assist customers with their security and compliance needs.

The AWS Partner Network has earned endorsement from AWS.

AWS Professional Services and AWS Partner Network both help customers develop security policies and procedures based on well-proven designs, and help to ensure that customers' security design meets internal and external compliance requirements.

AWS Advisories and Bulletins



- 📦 Advisories/bulletins provided on current vulnerabilities and threats.
- 📦 Customers work with experts to address:
 - 📦 Reporting abuse
 - 📦 Vulnerabilities
 - 📦 Penetration testing



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

With AWS Advisories and Bulletins, AWS provides advisories around current vulnerabilities and threats and enables customers to work with AWS security experts to address concerns like reporting abuse, vulnerabilities, and penetration testing.

AWS Auditor Learning Path



- 📦 Understand how internal operations gain compliance on AWS.
- 📦 Visit the compliance website:
 - 📦 Recommended training
 - 📦 Self-paced labs
 - 📦 Auditing resources



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

If you are in an auditing, compliance, or legal role, check out AWS Auditor Learning Path to obtain a better understanding of how your internal operations can demonstrate compliance using AWS. You can access Recommended Training, self-paced labs, and auditing resources from the Compliance website.

AWS Compliance Solutions Guide



- 📁 Understand the Shared Responsibility Model
- 📁 Request a compliance report
- 📁 Complete a security questionnaire
- 📁 Services in scope
- 📁 AWS Security Blog
- 📁 Case studies
- 📁 FAQs



For additional compliance information see:
<https://aws.amazon.com/compliance/resources/>

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

If you do not know where to start with compliance or need to access frequently used resources and processes, check out the **AWS Compliance Solutions Guide**. Learn about the available compliance solutions such as:

- Understanding the Shared Responsibility Model
- Requesting a Compliance Report
- Completing a Security Questionnaire

More AWS Compliance Resources include:

- **Services in Scope** – Details which services are currently in scope and which are in progress.
- **AWS Security Blog** – The blog is a great way to track all the newest updates to AWS security programs.
- **Case Studies** – Provide insightful information on some of the AWS current customer experiences with security.

You can also get answers to frequently asked questions for specific compliance types, such as:

- Certifications and attestations:
 - Payment Card Industry (PCI)
 - System & Organization Control (SOC)
 - Federal Risk and Authorization Management Program (FedRAMP)
- Laws and regulations, such as the U.S. Health Insurance Portability and Accountability Act (HIPAA)

Select the link to learn more.

<https://aws.amazon.com/compliance/resources/>

Module 4 Review:



- 📁 Reviewed the AWS Shared Responsibility Model
- 📁 Discussed IAM
- 📁 Reviewed AWS Trusted Advisor, AWS CloudTrail, and AWS Config
- 📁 Explored the AWS security and compliance programs
- 📁 Explored additional AWS security resources
- 📁 Demonstrated and discussed Day One best practices

To finish this module:

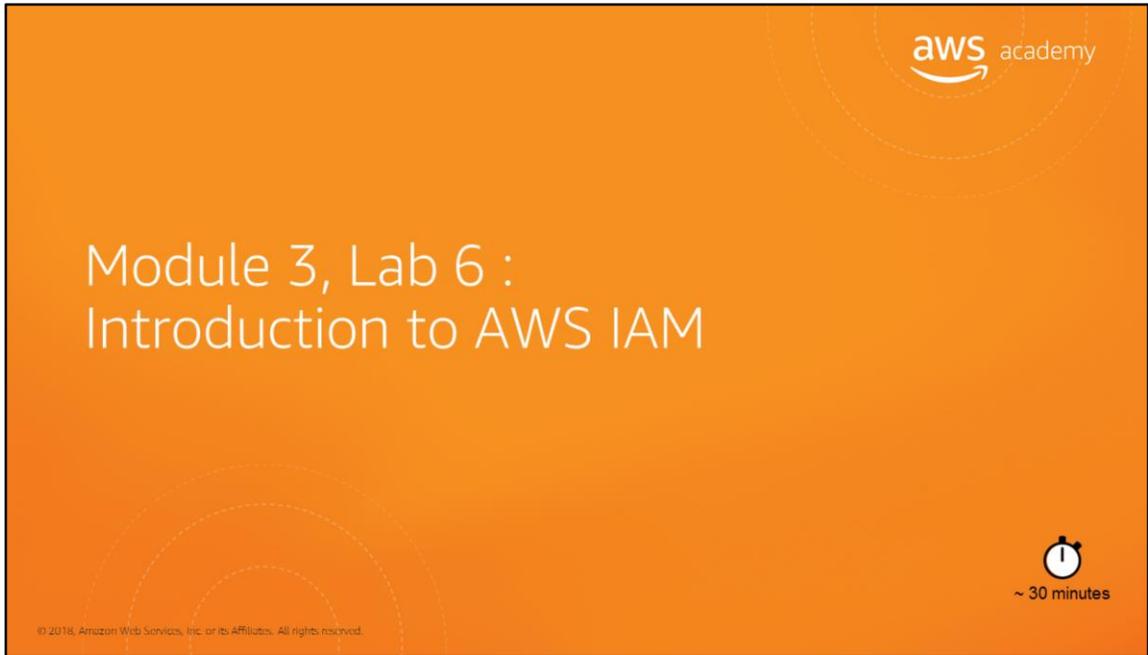
- 📁 Complete: **Knowledge Assessment**

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

In summary, we:

- Reviewed the AWS Shared Responsibility Model
- Discussed IAM
- Reviewed AWS Trusted Advisor, AWS CloudTrail, and AWS Config
- Explored the AWS security and compliance programs
- Explored additional AWS security resources
- And demonstrated and discussed Day One best practices

To finish this module, complete the knowledge assessment.



Introducing Module 3, Lab 6: Introduction to AWS IAM.

Lab 6: Scenario



In this lab, you will explore pre-created IAM users and groups and add users to groups with specific capabilities enabled.



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

IAM is a web service that enables AWS customers to manage users and user permissions in AWS. With IAM, you can centrally manage users, security credentials, such as access keys, and permissions that control which AWS resources users can access.

In this lab, you will:

- Explore pre-created IAM users and groups.
- Inspect IAM policies as applied to the pre-created groups.
- Follow a real-world scenario, adding users to groups with specific capabilities enabled.
- Locate and use the IAM sign-in URL.
- Experiment with the effects of policies on service access.

Lab 6: Tasks



Explore the **Users and Groups**.



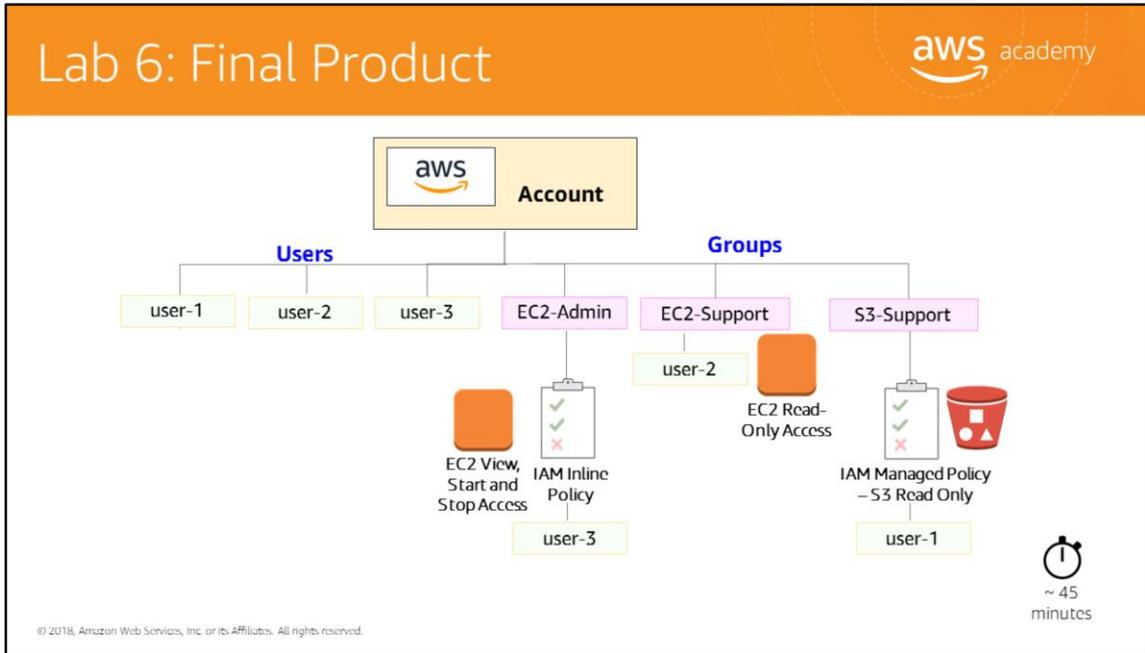
Add **Users** to **Groups**.



Sign in and **test Users**.

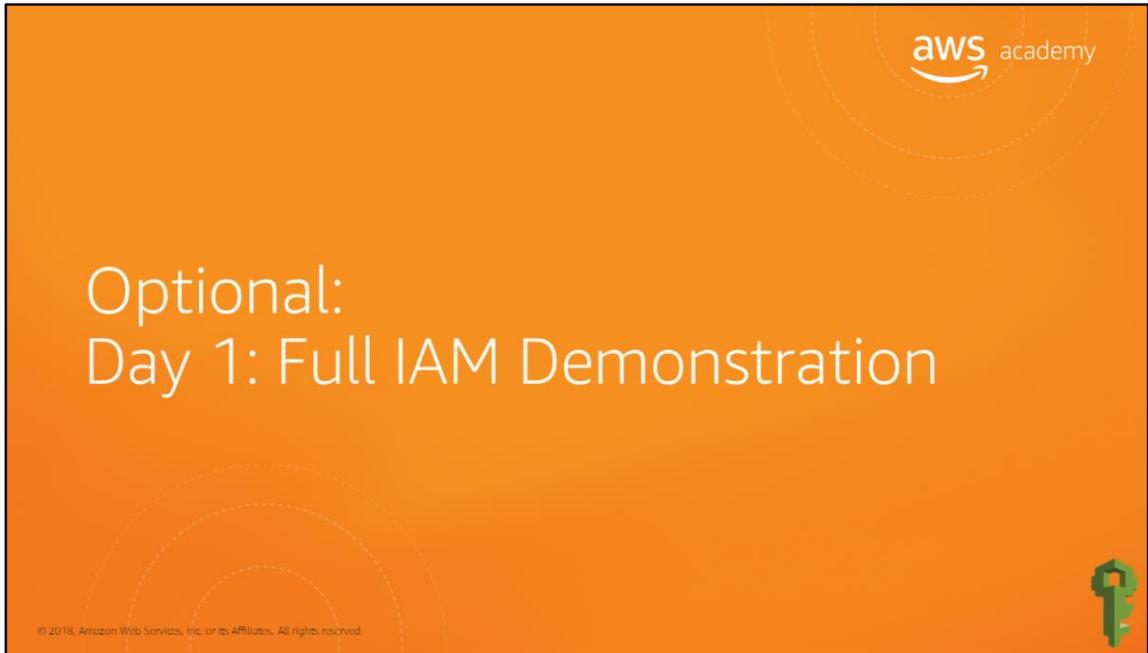
© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

In this lab, you'll explore the users and groups, add users to groups and sign in and test users.



In this lab, you:

- Explored pre-created IAM Users and Groups
- Inspected IAM policies as applied to the pre-created Groups
- Followed a real-world scenario, adding Users to Groups with specific capabilities enabled
- Located and use the IAM sign-in URL
- Experimented with the effects of policies on service access



Introducing the Day 1 IAM Demonstration:

1. Go to IAM.
2. Review Security Status by taking the following steps.:
 - Activate MFA on your root account.
 - Create individual IAM users.
 - Use groups to assign permissions.
 - Apply an IAM password policy.

The goal is to receive green checks by each security status item.

The screenshot displays the AWS IAM console's Security Status page. The 'Custom Sign In Link' is highlighted with a red box and a red arrow pointing to the 'Customize' button. The 'Security Status' section shows a progress bar at 1 out of 5 complete, with a list of tasks: 'Delete your root access keys' (checked), 'Activate MFA on your root account' (warning), 'Create individual IAM users' (warning), 'Use groups to assign permissions' (warning), and 'Apply an IAM password policy' (warning).

A review of the current Security Status indicates that:

- MFA has not been activated on the root account.
- No individual IAM users have been created.
- No permissions have been assigned to groups.
- No IAM password policy has been applied.

There is a custom sign-in link for the account. Note that the account number has been hidden for security purposes. Use the **Customize** button to change the name of the account so that the account number is not displayed. This link is used to sign in to the account and can be sent to users as they are set up.

Activate MFA on Root Account

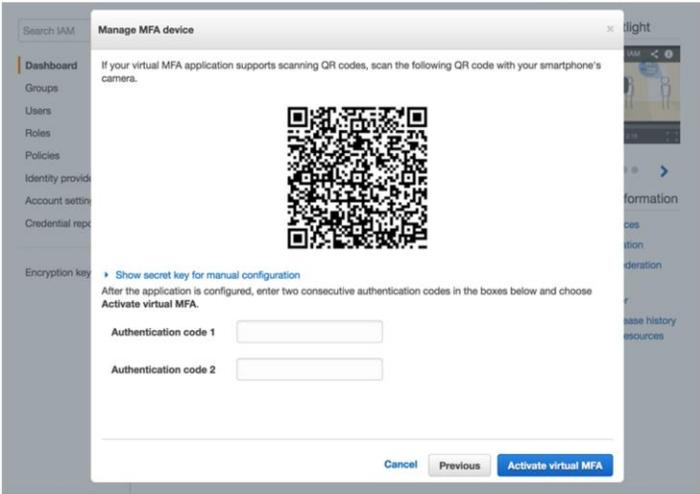
The screenshot displays the AWS IAM console interface. On the left, there is a navigation menu with 'Custom Sign In Link' and 'MFA Activation' highlighted. The main content area shows the 'Welcome to Identity and Access Management' dashboard. A red box highlights the sign-in link: `https://console.signin.aws.amazon.com`. Below this, the 'Security Status' section shows a progress bar at '1 out of 5 complete'. A red box highlights the 'Activate MFA on your root account' option, which is marked with a warning icon. Other options include 'Delete your root access keys', 'Create individual IAM users', 'Use groups to assign permissions', and 'Apply an IAM password policy'. The right sidebar contains a 'Feature Spotlight' video and 'Additional Information' links.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Before creating the users, activate MFA on the root account. The root account is the email address that you used to sign up for the AWS account. The root account has access to everything; thus, why it's important to secure this account with restrictions.

To set up MFA, click **Activate MFA on your root account** and then click **Manage MFA**. Two options will be presented: Virtual and Hardware. A hardware device is an actual hardware device. For purposes of this demonstration, select **Virtual** and then click **Next Step**. A new dialog box appears and asks us to configure a virtual MFA device. An app must be downloaded for this task. After that is complete, click **Next**.

Activate MFA on Root Account



The screenshot displays the AWS IAM console interface. A modal window titled "Manage MFA device" is open. It contains a QR code for scanning. Below the QR code, there are two input fields labeled "Authentication code 1" and "Authentication code 2". At the bottom of the modal, there are three buttons: "Cancel", "Previous", and "Activate virtual MFA". The "Activate virtual MFA" button is highlighted in blue. The AWS Academy logo is visible in the top right corner of the slide.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

In the authenticator application, press the **plus sign**. Scan the barcode, enter the two authentication codes, and then click the **Activate Virtual MFA** button.

MFA on Root Account is Activated

The screenshot displays the AWS IAM console interface. At the top, a banner reads "MFA on Root Account is Activated" with the AWS Academy logo. The main content area is titled "Welcome to Identity and Access Management" and includes a sign-in link, IAM resource counts (Users: 0, Roles: 0, Groups: 0, Identity Providers: 0), and a "Security Status" section. The "Security Status" section shows a progress bar at "2 out of 5 complete" and a list of tasks: "Delete your root access keys" (checked), "Activate MFA on your root account" (checked and highlighted with a red box), "Create individual IAM users" (warning), "Use groups to assign permissions" (warning), and "Apply an IAM password policy" (warning). On the left sidebar, the "MFA Activated" status is visible. The right sidebar contains a "Feature Spotlight" video and "Additional Information" links. A green key icon is located in the bottom right corner of the screenshot.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Click **Finish** and refresh your browser. The MFA should now show that it is set up.

Create an Individual IAM User

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Most AWS accounts are set up as company accounts with multiple users. Each user is set up with individual permissions or included as part of a group with specific permissions. A best practice is to have each user have their own account so they are not logging in a root with global privileges.

Click **Create individual IAM users** and then select **Manage Users**.

Create an Individual IAM User

aws academy

Add user

1 Details 2 Permissions 3 Review 4 Complete

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name* Mickey_Mouse

[Add another user](#)

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type* Programmatic access
Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.

AWS Management Console access
Enables a password that allows users to sign-in to the AWS Management Console.

Console password* Autogenerated password
 Custom password

Require password reset User must create a new password at next sign-in
Users automatically get the IAMUserChangePassword policy to allow them to change their own password.

* Required

Cancel Next: Permissions

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Select **Add another user**. Add a username. Note that usernames cannot have spaces.

Select the **Access types**. There are two access types:

- **Programmatic access** enables the user to have command line access to provision resources. This option will generate an access key one time. This access key must be saved as it will be used for all future access.
- **AWS Management Console access** enables user to log in to the AWS console.

Select a password type.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Next, you will assign permissions. There are three options:

- Add user to group
- Copy permissions from an existing user
- Attach existing policies directly

We want to add the user to a group, so select **Add user to group** and then select the **Create group** button. A group is where you put users to inherit the policies assigned to the group.

Create an Individual IAM User



Create group
✕

Create a group and select the policies to be attached to the group. Using groups is a best-practice way to manage users' permissions by job functions, AWS service access, or your custom permissions. [Learn more](#)

Group name

Create policy

Filter: Policy type Showing 313 results

	Policy name	Type	Attachments	Description
<input checked="" type="checkbox"/>	AdministratorAccess	Job function	0	Provides full access to AWS services and resources.
<input type="checkbox"/>	AlexaForBusinessDeviceSetup	AWS managed	0	Provide device setup access to AlexaForBusiness services
<input type="checkbox"/>	AlexaForBusinessFullAccess	AWS managed	0	Grants full access to AlexaForBusiness resources and access to relat...
<input type="checkbox"/>	AlexaForBusinessGatewayEx...	AWS managed	0	Provide gateway execution access to AlexaForBusiness services
<input type="checkbox"/>	AlexaForBusinessReadOnlyA...	AWS managed	0	Provide read only access to AlexaForBusiness services
<input type="checkbox"/>	AmazonAPIGatewayAdminist...	AWS managed	0	Provides full access to create/edit/delete APIs in Amazon API Gatew...
<input type="checkbox"/>	AmazonAPIGatewayInvokeFu...	AWS managed	0	Provides full access to invoke APIs in Amazon API Gateway.
<input type="checkbox"/>	AmazonAPIGatewayPushToC...	AWS managed	0	Allows API Gateway to push logs to user's account.
<input type="checkbox"/>	AmazonAppStreamFullAccess	AWS managed	0	Provides full access to Amazon AppStream via the AWS Managemen...
<input type="checkbox"/>	AmazonAppStreamReadOnly...	AWS managed	0	Provides read only access to Amazon AppStream via the AWS Mana...

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Give the group a name. For this example, give the lead developer administrative access. Select the **Create group** button.

Create an Individual IAM User

aws academy

Add user

1 Details 2 Permissions 3 Review 4 Complete

Set permissions for Mickey_Mouse

Add user to group Copy permissions from existing user Attach existing policies directly

Add user to an existing group or create a new one. Using groups is a best practice way to manage user's permissions by job functions. [Learn more](#)

Create group Refresh

Search Showing 1 result

Group	Attached policies
<input checked="" type="checkbox"/> Administrators	AdministratorAccess

Cancel Previous Next Review

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Select **Next Review** to review what is being created, and then select **Create user**.

IAM User Creation Successful

aws academy

Add user

1 Details 2 Permissions 3 Review 4 Complete

Success
 You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.
 Users with AWS Management Console access can sign-in at: <https://raysinut.signin.aws.amazon.com/console>

Download .csv

User	Access key ID	Secret access key	Password	Email login instructions
Mickey_Mouse		***** Show	***** Show	Send email ↗

Close

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

When a user is created, several things are generated:

1. **Access key ID** that is used to access AWS at the command line to programmatically access AWS, or blocked out.
2. **Secret access key** that is used to access AWS at the command line to programmatically access AWS.
3. **Password** that is used to log in to the console.

Press **Show** to display the values in each of the fields. All of the information can be downloaded by selecting the **Download csv button**.

Note: Never put this information in a public place. This information can be used to access your account.

IAM Dashboard Update

The screenshot shows the AWS IAM Dashboard with the following sections:

- Search IAM**: Search bar.
- Navigation Menu**: Dashboard, Groups, Users, Roles, Policies, Identity providers, Account settings, Credential report, Encryption keys, Password Policy Creation.
- Welcome to Identity and Access Management**: IAM users sign-in link: <https://signin.aws.amazon.com/console>. Buttons: Customize | Copy Link.
- IAM Resources**: Users: 1, Roles: 0, Groups: 1, Identity Providers: 0, Customer Managed Policies: 0.
- Security Status**: 4 out of 5 complete. Items: Delete your root access keys (checked), Activate MFA on your root account (checked), Create individual IAM users (checked), Use groups to assign permissions (checked), Apply an IAM password policy (warning icon, highlighted).
- Feature Spotlight**: Introduction to AWS IAM video player.
- Additional Information**: IAM best practices, IAM documentation, Web Identity Federation Playground, Policy Simulator, Videos, IAM release history and additional resources.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

When you return to the dashboard, the individual IAM user and group security status items have been addressed.

The last thing to do is apply an IAM password policy.

Set IAM Password Policy

You have unapplied changes to your password policy.

A password policy is a set of rules that defines the type of password an IAM user can set. For more information about password policies, go to [Temporary Passwords in Using IAM](#).

Currently, this AWS account does not have a password policy. Specify a password policy below.

Minimum password length:

- Require at least one uppercase letter
- Require at least one lowercase letter
- Require at least one number
- Require at least one non-alphanumeric character
- Allow users to change their own passwords
- Enable password expiration

Password expiration period (in days):

Passwords must be unique

Number of passwords to remember:

Password expiration requires administrator reset

[Apply password policy](#) [Delete password policy](#)

Security Token Service Regions

You can enable additional regions from which you can request temporary credentials. Activate only the regions you intend to use. [Learn More.](#)

Region	Status	Actions
US East (N. Virginia)	Always Active	
US East (Ohio)	Active	Deactivate
US West (N. California)	Active	Deactivate
US West (Oregon)	Active	Deactivate
Canada (Central)	Active	Deactivate
EU (Ireland)	Active	Deactivate
EU (Frankfurt)	Active	Deactivate
EU (London)	Active	Deactivate
EU (Paris)	Active	Deactivate
Asia Pacific (Singapore)	Active	Deactivate
Asia Pacific (Sydney)	Active	Deactivate
Asia Pacific (Tokyo)	Active	Deactivate
Asia Pacific (Chennai - Local)	Active	Deactivate
Asia Pacific (Bahrain)	Active	Deactivate
South America (Sao Paulo)	Active	Deactivate

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

The IAM password policy is a set of rules that defines the type of password that an IAM user can set.

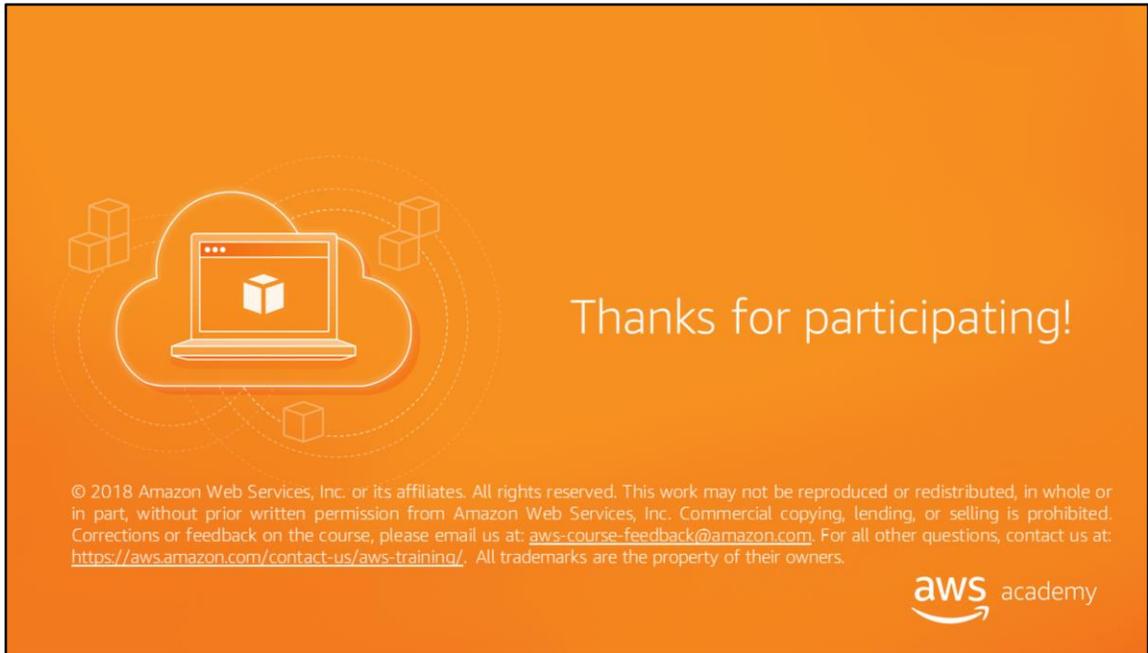
Select the rules that the passwords should follow and click the **Apply password policy** button.



Up Next: Module 4 – Cloud Architecting

- Part 1: Introduction to the Well-Architected Framework
- Part 2: Well-Architected Design Principles
- Part 3: Understanding Reliability and High Availability
- Part 4: Scaling
- Part 5: Example - Transitioning a Data Center to the Cloud

In the next module we will look at principles for architecting a cloud solution.



Thanks for participating!