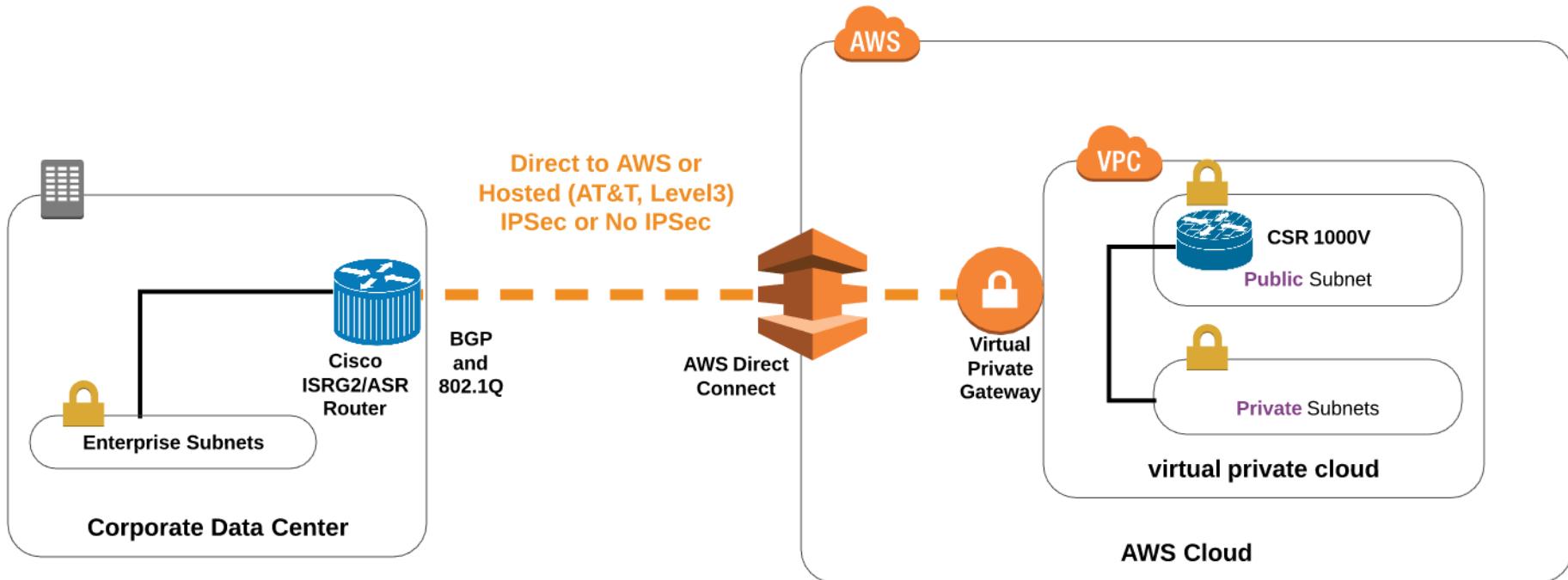


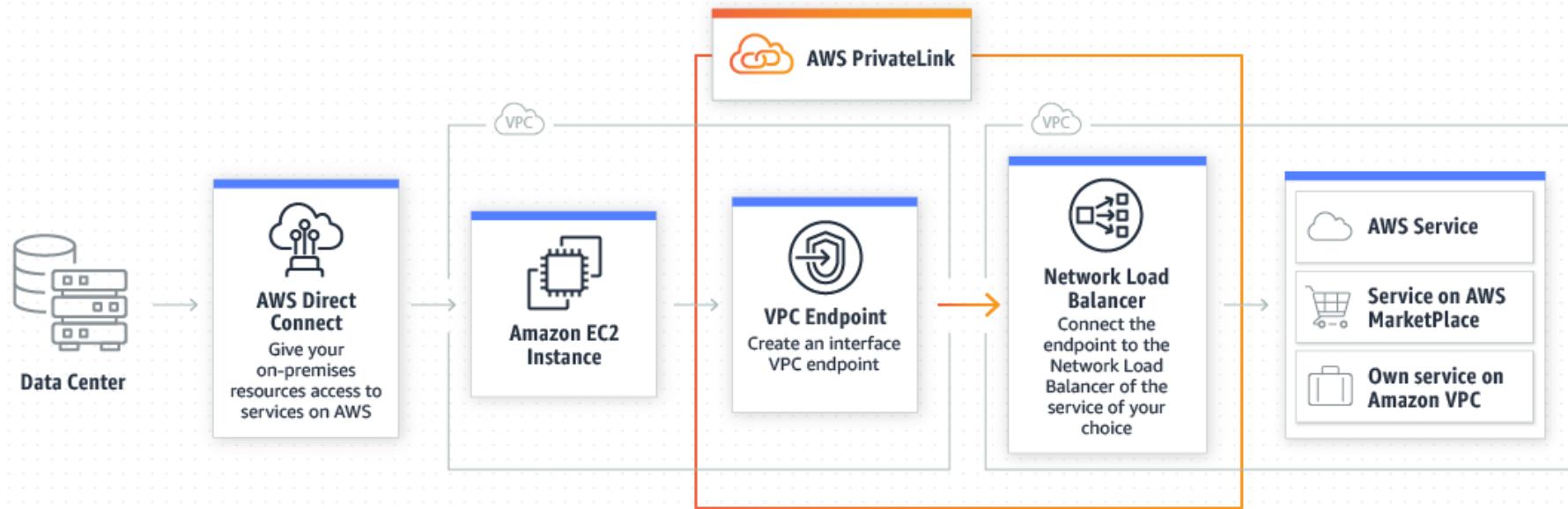
# AWS Direct Connect



# AWS PrivateLink

- AWS PrivateLink simplifies the security of data shared with cloud-based applications by removing the exposure of the public Internet
- PrivateLink offers isolated connectivity between VPCs, AWS services, and on-premises applications, securely on the Amazon network infrastructure
- PrivateLink makes it easy to connect services across different accounts and VPCs to considerably streamline the network architecture

# AWS PrivateLink



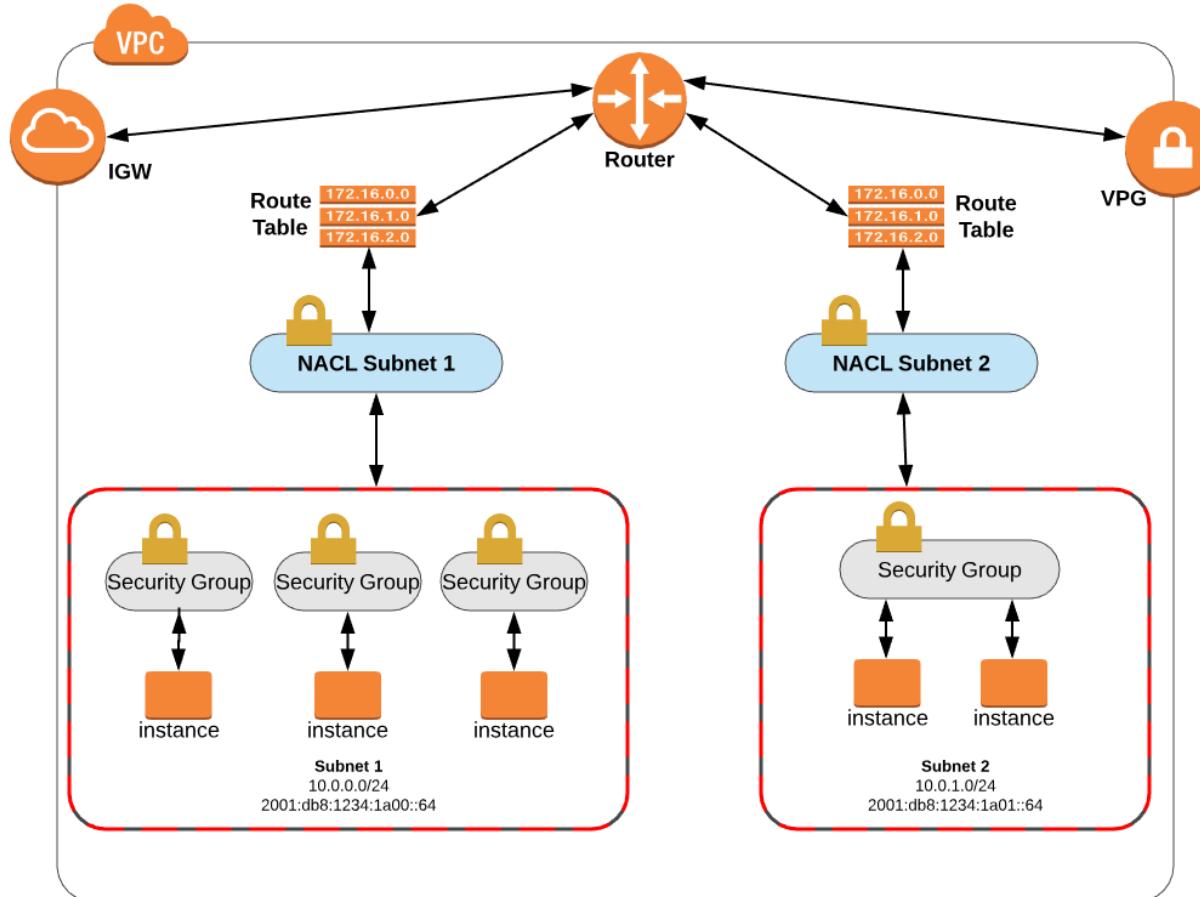
# AWS Transit Gateway

- **AWS Transit Gateway** allows customers to connect their VPCs and their on-premises networks to a single gateway
- You can easily scale your networks across multiple accounts and Amazon VPCs to keep up with growth
- You only need to create and manage a single connection (hub) from the central gateway to each VPC, on-premises data center, or remote office across your network
- Any new VPC is simply connected to the Transit Gateway and is then automatically available to every other network that is connected to the Gateway

# Network ACLs

- NACLs allow stateless traffic filtering and management of IPv4 and IPv6 traffic
- Applies to all inbound OR outbound traffic from a subnet within a VPC
- Can contain ordered rules (ACE's) to permit or deny based on IP protocol (for example GRE, IPSec ESP, ICMP), service port, and source/destination IP address
- NACLs are agnostic of TCP and UDP sessions
- NACLs work in conjunction with security groups and can permit or deny traffic before it reaches the security group

# NACLs and Security Groups



# NAACLs

The screenshot shows the AWS VPC Management Console with the Subnets page open. The left sidebar lists various VPC components, with 'Subnets' selected. The main area displays a table of subnets, and the 'Public subnet' row is highlighted with a red box. Below the table, the 'Edit' tab is selected for the Network ACL of the Public subnet, and its configuration is shown in a red-bordered box.

**Subnets | VPC Management Con X**

Services ▾ Resource Groups ▾

shankhantoo ▾ Ohio ▾ Support ▾

VPC Dashboard

Filter by VPC: Select a VPC

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

Endpoint Services

NAT Gateways

Peering Connections

Security

Create Subnet Subnet Actions

Search Subnets and their prop X

1 to 5 of 5 Subnets

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR	Available IPv6
Private subnet	subnet-f5ff558e	available	vpc-63864f0b   MY-VPC	10.0.1.0/24	251		us-east-2
	subnet-0e6d6575	available	vpc-1f30fc77	172.31.16.0/20	4090		us-east-2
	subnet-e71758aa	available	vpc-1f30fc77	172.31.32.0/20	4091		us-east-2
Public subnet	subnet-dc5852a7	available	vpc-63864f0b   MY-VPC	10.0.0.0/24	250		us-east-2

**Edit**

Network ACL: acl-c37eddb

Inbound:

Rule #	Type	Protocol	Port Range / ICMP Type	Source	Allow / Deny
100	ALL Traffic	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

Outbound:

Rule #	Type	Protocol	Port Range / ICMP Type	Destination	Allow / Deny
100	ALL Traffic	ALL	ALL	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

Feedback English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use



# NAACLs

The screenshot shows the AWS VPC Management Console with the Network ACLs page open. A search bar at the top has 'acl-c37eddb' entered. On the left, a sidebar lists various VPC components like Subnets, Route Tables, and Security Groups. The main area shows a 'Create Network ACL' dialog for 'acl-c37eddb'. The dialog includes fields for Name (set to 'acl-c37eddb'), a dropdown menu of ports (e.g., DNS (TCP) (53), HTTP (80)), and two input fields for Rule # (100 and 101). Below these are tabs for Summary, Inbound, and Outbound. The Outbound tab is selected, showing a table with one row:

Protocol	Port Range	Source	Allow / Deny	Remove
ALL	ALL	0.0.0.0/0	ALLOW	X

The right side of the screen displays the details for the 'acl-c37eddb' Network ACL, which is associated with two subnets and is set to 'Default'. It also shows tabs for Rules, Subnet Associations, and Tags.

Feedback

English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy Terms of Use



Pearson

# NACL Recommendations

- AWS Documentation » **Amazon Virtual Private Cloud** » **User Guide** » Security » Recommended Network ACL Rules for Your VPC

VPC with a Single Public Subnet
<b>VPC with Public and Private Subnets</b>
VPC with Public and Private Subnets and Hardware VPN Access
VPC with a Private Subnet Only and Hardware VPN Access

ACL Rules for the Public Subnet

Inbound					
Rule #	Source IP	Protocol	Port	Allow/Deny	Comments
100	0.0.0.0/0	TCP	80	ALLOW	Allows inbound HTTP traffic from any IPv4 address.
110	0.0.0.0/0	TCP	443	ALLOW	Allows inbound HTTPS traffic from any IPv4 address.
120	Public IP address range of your home network	TCP	22	ALLOW	Allows inbound SSH traffic from your home network (over the Internet gateway).
130	Public IP address range of your home network	TCP	3389	ALLOW	Allows inbound RDP traffic from your home network (over the Internet gateway).
140	0.0.0.0/0	TCP	1024-65535	ALLOW	Allows inbound return traffic from hosts on the Internet that are responding to requests originating in the subnet.  This range is an example only. For information about choosing the correct ephemeral ports for your configuration, see <a href="#">Ephemeral Ports</a> .
*	0.0.0.0/0	all	all	DENY	Denies all inbound IPv4 traffic not already handled by a preceding rule (not modifiable).

# Security Groups

- A security group is a virtual layer 3/4 **stateful** firewall that controls the (whitelisted only) traffic flow for its associated instances
- SGs operate at the hypervisor level for all EC2 instances and other VPC objects
- All EC2 instances are launched with the default SG unless a user-defined SG is specified when spun up
- An unchanged default SG will **permit** communication between all resources within the security group AND allows all outbound traffic
- All other traffic is implicitly denied

# Security Groups

- Security groups are stateful—if you send a request from your instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules
- IOW, Responses to allowed inbound traffic are allowed to flow out, regardless of outbound rules
- You add the inbound rules to control incoming traffic to the instance and outbound rules to control the outgoing traffic from your instance
- Remember: You can specify allow rules, but not deny rules

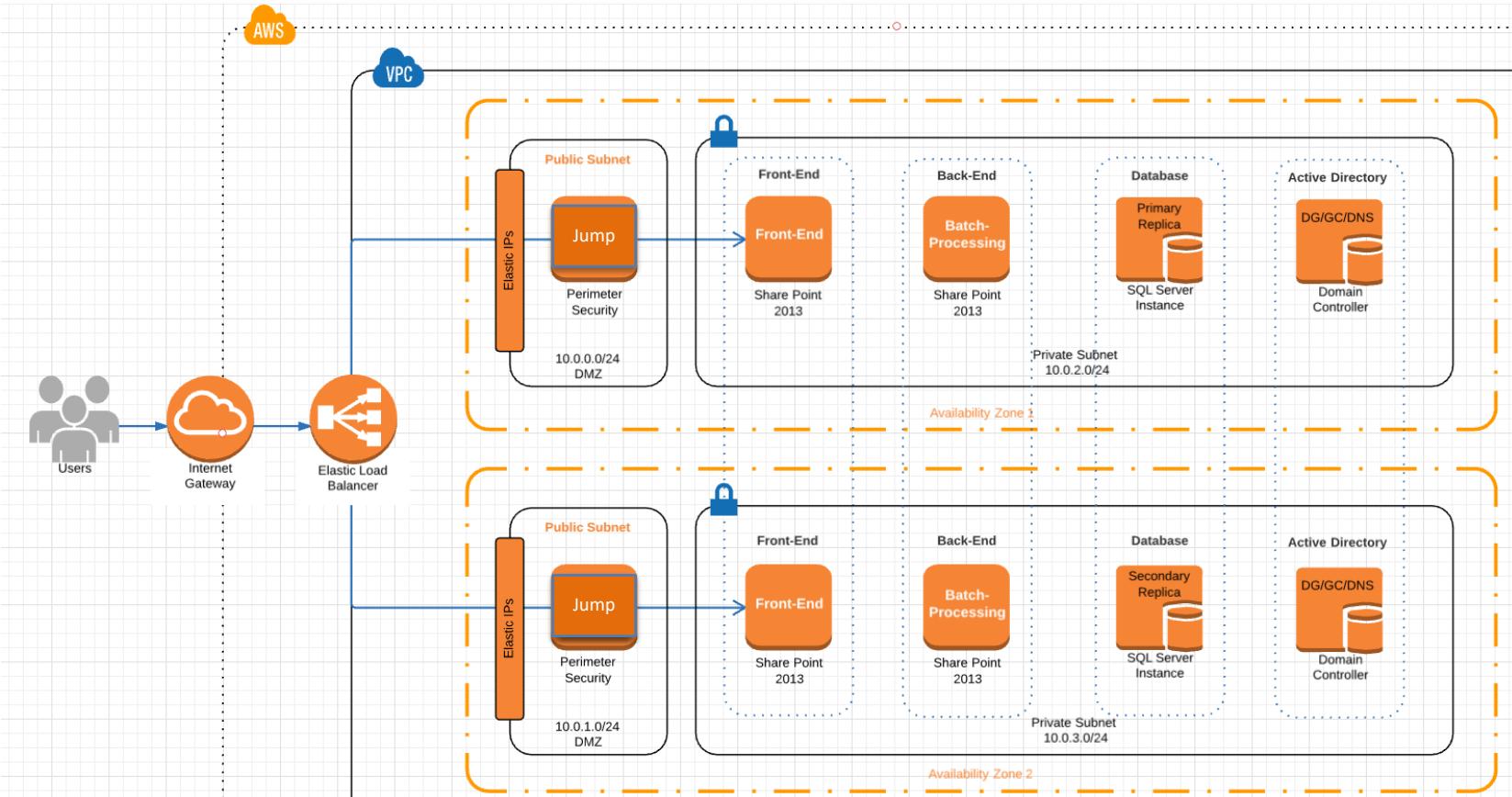
# Security Groups

- To associate a security group with an instance, it is best practice to specify the security group when you launch the instance
- When **you** create a security group, it has no inbound rules. Therefore, no inbound traffic originating from another host to your instance is allowed until you add inbound rules to the security group
- By default, an SG includes an outbound rule that allows all outbound traffic but no inbound traffic is allowed until you add inbound rules to the security group

# Comparing Security Groups and NACLs

Network ACL	Security Group
Functions at the network level	Functions at the instance level
Supports allow and deny rules	Supports allow rules only (whitelisting)
Stateless so return traffic must be explicitly allowed	Stateful so that return traffic is automatically allowed
Rules are processed in a numbered order	All rules are evaluated before deciding to allow traffic
Applies automatically to all of the instances in the associated subnet	Applies to the instance only

# SG Defense-in-Depth



# Assign a Security Group

The screenshot shows the AWS EC2 Management Console interface. The top navigation bar includes the EC2 Management Console logo, a search bar with the URL https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#LaunchInstance, and user account information (shankantoo, Ohio, Support). Below the navigation bar, a breadcrumb trail indicates the current step: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group (which is highlighted in orange), and 7. Review.

## Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more about Amazon EC2 security groups.](#)

**Assign a security group:**

- Create a **new** security group
- Select an **existing** security group

Security Group ID	Name	Description	Actions
<input type="checkbox"/> sg-0e998166	default	default VPC security group	<a href="#">Copy to new</a>

Select a security group above to view its inbound rules.

Cancel Previous **Review and Launch**

# Default Security Group

The screenshot shows the AWS VPC Manager interface. The left sidebar is collapsed, and the main area displays the 'Security Groups' page. A red box highlights the 'Security Groups' link in the sidebar. The main content area shows a table of security groups. One row, 'sg-0e998166', is selected and highlighted with a red box. This row corresponds to the 'default' VPC security group. Below the table, tabs for 'Summary', 'Inbound Rules', 'Outbound Rules', and 'Tags' are visible, with 'Outbound Rules' being the active tab. Another red box highlights the 'Edit' button above the outbound rules table. The outbound rules table has columns: Type, Protocol, Port Range, Destination, and Description. A single rule is listed: ALL Traffic, ALL, ALL, 0.0.0.0/0.

Type	Protocol	Port Range	Destination	Description
ALL Traffic	ALL	ALL	0.0.0.0/0	



# Default Security Group

Inbound			
Source	Protocol	Port Range	Comments
The security group ID (sg-xxxxxxxx)	All	All	Allow inbound traffic from instances assigned to the same security group.
Outbound			
Destination	Protocol	Port Range	Comments
0.0.0.0/0	All	All	Allow all outbound IPv4 traffic.
::/0	All	All	Allow all outbound IPv6 traffic. This rule is added by default if you create a VPC with an IPv6 CIDR block or if you associate an IPv6 CIDR block with your existing VPC.

# Inbound Rules to Web Servers

The screenshot shows the AWS VPC Manager interface. The left sidebar has a red box around the 'Security Groups' link under the 'Security' section. The main content area shows a list of security groups with one selected, also highlighted by a red box. The selected security group is 'sg-ea4cab81'. Below it, the 'Inbound Rules' tab is selected (also highlighted by a red box), showing a table of rules:

Type	Protocol	Port Range	Source	Description	Remove
HTTP (80)	TCP (6)	80	0.0.0.0/0	From all IPv4 addresses	X
HTTP (80)	TCP (6)	80	::/0	From all IPv6 addresses	X
HTTPS (443)	TCP (6)	443	0.0.0.0/0	From all IPv4 addresses	X
HTTPS (443)	TCP (6)	443	::/0	From all IPv6 addresses	X
SSH (22)	TCP (6)	22	50. 235/32	(From the Internet gateway)	X
RDP (3389)	TCP (6)	3389	50. 235/32	(From the Internet gateway)	X

At the bottom, there is a button labeled 'Add another rule'.

# Outbound Rules to Web Servers

The screenshot shows the AWS VPC Manager interface. The left sidebar navigation bar includes links for Route Tables, Internet Gateways, Egress Only Internet Gateways, DHCP Options Sets, Elastic IPs, Endpoints, Endpoint Services, NAT Gateways, Peering Connections, Security, Network ACLs, and Security Groups. The Security Groups link is highlighted with an orange border. The main content area displays a list of security groups under the heading "Create Security Group" and "Security Group Actions". A search bar at the top right allows filtering by "All security groups" or "Search Security Groups and th". The list shows two security groups: "sg-0e998166" (default VPC security group) and "sg-ea4cab81" (default VPC security group). The "sg-ea4cab81" group is selected, and its details are shown in the main pane. The "Outbound Rules" tab is active, showing two rules:

Type	Protocol	Port Range	Destination	Description	Remove
MS SQL (1433)	TCP (6)	1433	pl-4ca54025	(info)	X
MySQL/Aurora (3306)	TCP (6)	3306	pl-4ca54025	(info)	X

An "Add another rule" button is available at the bottom of the rule list.





## Segment 4: AWS WAF, Shield Advanced, and GuardDuty

# AWS Web Application Firewall (WAF)

- AWS WAF is a web application firewall that lets you monitor the HTTP and HTTPS requests forwarded to Amazon CloudFront or an ELB Application Load Balancer
- At a basic level WAF can:
  - Allow all requests except for ones you designate (permissive)
  - Block all requests except for ones you designate (restrictive)
  - Count the requests that match the properties that you specify (monitor mode before deployment)

# Using Managed Rule Groups

The screenshot shows the AWS WAF 'Create web ACL' interface. On the left, a vertical sidebar lists five steps: Step 1 (Describe web ACL and associate it to AWS resources), Step 2 (Add rules and rule groups: Add managed rule groups), Step 3 (Set rule priority), Step 4 (Configure metrics), and Step 5 (Review and create web ACL). The current step, Step 2, is highlighted. The main content area is titled 'Add managed rule groups' with an 'Info' link and a 'Close' button. It contains four sections: 'AWS managed rule groups', 'Cyber Security Cloud Inc. managed rule groups', 'Fortinet managed rule groups', and 'GeoGuard managed rule groups'. At the bottom right are 'Cancel' and 'Add rules' buttons.

AWS WAF > Web ACLs > Create web ACL

Step 1  
Describe web ACL and associate it to AWS resources

Step 2  
Add rules and rule groups: Add managed rule groups

Step 3  
Set rule priority

Step 4  
Configure metrics

Step 5  
Review and create web ACL

Add managed rule groups [Info](#) [Close](#)

Managed rule groups are created and maintained for you by AWS and AWS Marketplace sellers.

▶ AWS managed rule groups

▶ Cyber Security Cloud Inc. managed rule groups

▶ Fortinet managed rule groups

▶ GeoGuard managed rule groups

[Cancel](#) [Add rules](#)

# Using Managed Rule Groups

Name	Capacity	Action
<b>Admin protection</b> Contains rules that allow you to block external access to exposed admin pages. This may be useful if you are running third-party software or would like to reduce the risk of a malicious actor gaining administrative access to your application.	100	<input type="checkbox"/> Add to web ACL
<b>Amazon IP reputation list</b> This group contains rules that are based on Amazon threat intelligence. This is useful if you would like to block sources associated with bots or other threats.	25	<input type="checkbox"/> Add to web ACL
<b>Core rule set</b> Contains rules that are generally applicable to web applications. This provides protection against exploitation of a wide range of vulnerabilities, including those described in OWASP publications and common Common Vulnerabilities and Exposures (CVE).	700	<input type="checkbox"/> Add to web ACL
<b>Known bad inputs</b> Contains rules that allow you to block request patterns that are known to be invalid and are associated with exploitation or discovery of vulnerabilities. This can help reduce the risk of a malicious actor discovering a vulnerable application.	200	<input type="checkbox"/> Add to web ACL
<b>Linux operating system</b> Contains rules that block request patterns associated with exploitation of vulnerabilities specific to Linux, including LFI attacks. This can help prevent attacks that expose file contents or execute code for which the attacker should not have had access.	200	<input type="checkbox"/> Add to web ACL
<b>PHP application</b>		

# WAF Matching Attributes

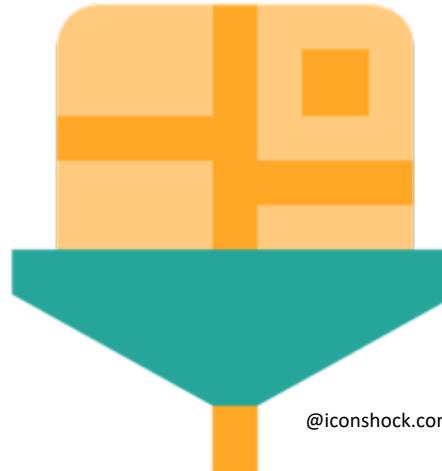
- IP addresses of originating requests
- Country that requests originate from
- Values in request headers  
(e.g. User-Agent, Content-Type)
- Literal or regex string patterns that appear in requests (e.g. [cC][mM][dD].[eE][xX][eE])
- Length of requests (buffer overflows)
- Presence of SQL injection code that is likely to be malicious
- Presence of a malicious cross-site scripting attack



@iconshock.com

# WAF Matching Attributes

- Rules that can allow, block, or count web requests that meet the specified conditions
- Rules can block or count web requests that meet or exceed a specified number of requests in any 5-minute period
- Rules that you can reuse for multiple web applications
- Real-time metrics and sampled web requests
- Automated administration using



@iconshock.com

# WAF Rules

- **Regular rules** use only conditions to target specific requests
  - The requests come from 192.168.2.55
  - They contain the value EvilBot in the User-Agent header
  - They appear to include SQL-like code in the query string
- **Rate-based Rules** are similar to regular rules, with one addition: a rate limit
  - They count the requests that arrive from a specified IP address every five minutes
  - The rule can trigger an action if the number of requests exceed the rate limit

# Web ACLs

- After you combine your conditions into rules, you combine the rules into a web ACL
- This is where you define an action for each rule—allow, block, or count—**and a default action**



@iconshock.com

# AWF WAF Configuration

## Web ACL details

**Name**  
  
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and \_ (underscore).

**Description - optional**  
  
The description can have 1-256 characters.

**CloudWatch metric name**  
  
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and \_ (underscore).

**Resource type**  
Choose the type of resource to associate with this web ACL.  
 CloudFront distributions  
 Regional resources (Application Load Balancer and API Gateway)

**Region**  
Choose the AWS region to create this web ACL in.  
 ▾

# Web ACL

## Rate-based rule

(combines conditions with an AND and adds a rate limit)



### Condition

Example: Cross-site scripting threat

AND



### Condition

Example: Specific IP addresses

AND



**Rate limit: 15,000**

If rule is met: do this action (Example: block)

## Rule

(combines conditions with an AND)



### Condition

Example: SQL injection threat

AND



### Condition

Example: Specific string in header

If rule is met: do this action (Example: count)

OR

# WAF Configuration Basics

## Conditions

### Suspicious IPs

192.0.2.0/24

192.51.100.0/24

2001:db8:a0b:12f0:ac34:1:1:1/128

2001:db8:a0b:12f0:0:0:0/64

### String match condition example

#### Bad bots

User-Agent header matches  
listbot

User-Agent header matches  
shopbot

### SQL injection match condition example

#### SQLi checks

URI contains SQL injection

Query string contains SQL  
injection

## Rules contain conditions

### Bad User-Agents

#### IP match condition

Suspicious IPs

and

#### String match condition

Bad bots

### Detect SQLi

#### SQL injection match condition

SQLi checks

## Web ACLs contain rules

Rule 1, Bad User-Agents, then  
block

#### IP match condition

Suspicious IPs

and

#### String match condition

Bad bots

or if requests match

Rule 2, Detect SQLi, then block

#### SQL injection match condition

SQLi checks

otherwise, perform the default action

### Default action

Allow requests that don't match  
any rules

# AWF WAF Configuration

## Create conditions

Conditions specify the filters that you want to use to allow or block requests that are forwarded to AWS resources such as Amazon CloudFront distributions.

### Cross-site scripting match conditions

Name

Create condition

You don't have any cross-site scripting match conditions. Choose **Create XSS match condition** to get started.

A cross-site scripting match condition specifies the parts of a web request (such as a User-Agent header) that you want AWS WAF to inspect for cross-site scripting threats. [Learn more](#)

### Geo match conditions

Name

Create condition

You don't have any geo match conditions. Choose **Create condition** to get started.

A geo match condition lets you allow, block, or count web requests based on the geographic origin of the request. [Learn more](#)

## Concepts overview

### Web ACL example

if requests match

Rule 1, Bad User-Agents, then block

#### IP match condition

Suspicious IPs

and

#### String match condition

Bad bots

or if requests match

Rule 2, Detect SQLi, then block

#### SQL injection match condition

SQLi checks

otherwise, perform the default action

# Geo Match Condition

The screenshot shows the AWS WAF & Shield console with a green header bar. The URL in the address bar is <https://console.aws.amazon.com/waf/home?region=global#/wizard//newgms/global>. A sidebar on the left lists 'Geo match conditions', 'IP match conditions', and 'Size constraint'. The main area is titled 'Create geo match condition'. It has a 'Region\*' dropdown set to 'Global (CloudFront)'. Below it is a descriptive text: 'Choose Global (CloudFront) to create AWS WAF resources to use with CloudFront distributions in all AWS Regions. Choose a specific AWS Region to create AWS WAF resources to use with an Application Load Balancer in that region.' A 'Filter settings' section allows adding one or more locations, with 'Location type\*' set to 'Country' and 'Location\*' set to 'Christmas Island - CX'. An 'Add location' button is available. A 'Filters in this geo match condition' section contains a 'Geographic origin of the request to filter on' input field containing 'Christmas Island - CX'. At the bottom are 'Required' and 'Create' buttons.

AWS WAF & Shield

https://console.aws.amazon.com/waf/home?region=global#/wizard//newgms/global

Most Visited

Create geo match condition

Region\*

Global (CloudFront)

Choose Global (CloudFront) to create AWS WAF resources to use with CloudFront distributions in all AWS Regions. Choose a specific AWS Region to create AWS WAF resources to use with an Application Load Balancer in that region.

Filter settings

Add one or more locations.

Location type\*

Country

Location\*

Christmas Island - CX

Add location

Filters in this geo match condition

Geographic origin of the request to filter on

Christmas Island - CX

\* Required

Create

# AWF WAF Configuration

## IP match conditions

Name	Create condition
You don't have any IP match conditions. Choose <b>Create IP match condition</b> to get started.	

An IP match condition specifies the IP addresses and/or IP address ranges that you want to use to control access to your content. Put IP addresses that you want to allow and IP addresses that you want to block into separate IP match conditions. [Learn more](#)

## Default action

Allow requests that don't match any rules

## Size constraint conditions

Name	Create condition
You don't have any size constraint conditions. Choose <b>Create size constraint condition</b> to get started.	

A size constraint condition specifies the parts of a web request (such as a User-Agent header) that you want AWS WAF to compare to a set size. [Learn more](#)

# IP Match Condition

The screenshot shows the AWS WAF & Shield console with the URL <https://console.aws.amazon.com/waf/home?region=global#/wizard//newips/global>. The main navigation bar has tabs for Step 1: Name web ACL, Step 2: Create conditions (which is active), Step 3: Create rules, and Step 4: Review and create. On the left, there are sections for Cross-site scripting match conditions, Geo match conditions, and IP match conditions. The IP match conditions section is expanded, showing sub-sections for IP address matching and IP range matching.

**Create IP match condition**

IP Version\*  IPv4  IPv6

Address\*

AWS WAF supports /8, /16, /24, and /32 CIDR blocks for IPv4 Examples:  
For a single IP address, please specify like 192.0.2.44/32  
For an IP range from 192.0.2.0 to 192.0.2.255, please use 192.0.2.0/24

Add IP address or range

**Filters in IP match condition**

IP address of the request to filter on

\* Required

Cancel Create

# AWF WAF Configuration

## SQL injection match conditions

Name

Create condition

You don't have any SQL injection match conditions.  
Choose **Create SQL injection match condition** to get started.

A SQL injection match condition specifies the parts of a web request (such as a User-Agent header) that you want AWS WAF to inspect for SQL queries. Create separate conditions for parts that you want to allow SQL queries in and parts that you don't. [Learn more](#)

## String and regex match conditions

Name

Create condition

You don't have any string or regex match conditions.  
Choose **Create condition** to get started.

A string match condition, or a regex match condition, specifies the part of a web request (such as a User-Agent header) and the text (the value of the header) that you want to use to control access to your content. Create separate conditions for strings or regex patterns that you want to allow or block. [Learn more](#)

# Regex Match Condition

The screenshot shows the AWS WAF & Shield console with a modal window titled "Create string match condition". The modal provides instructions for creating a string match condition, which contains a list of strings to allow or block. It includes fields for "Name\*" (set to "MATCH-BADREQUEST"), "Region\*" (set to "Global (CloudFront)"), and "Type\*" (set to "String match"). A dropdown menu for "Type\*" is open, showing "String match" (selected) and "Regex match". The "String match" option is highlighted with a red box. Below the dropdown, there is a note about choosing String match for CloudFront distributions and Regex match for Application Load Balancers. The "Filter settings" section is partially visible at the bottom.

AWS WAF & Shield

https://console.aws.amazon.com/waf/home#/wizard//newbms/global?isInWizard=true

Most Visited

conditions

String and regex match conditions

Create string match condition

A string match condition contains a list of the strings that appear in web requests that you want to allow or block.  
[Learn more](#)

Name\* MATCH-BADREQUEST

Region\* Global (CloudFront)

Choose Global (CloudFront) to create AWS WAF resources to use with CloudFront distributions in all AWS Regions. Choose a specific AWS Region to create AWS WAF resources to use with an Application Load Balancer in that region.

Type\* String match

String match

Regex match

Specify the settings that you want to use to allow or block web requests. If you add more than one filter to a string match condition, a web request needs to match only one of the filters for the request to match the string match condition. (The filters are ORed together.)

Part of the request to filter on

Please select

\* Required

Cancel Create



# Regex Match Condition

The screenshot shows the AWS WAF & Shield console with a modal dialog titled "Create regex match condition". The dialog is used to define a filter setting for a regex match condition. The "Part of the request to filter on" dropdown is set to "Body", which is highlighted with a blue border. Below this, there is a "Transformation" section and a "Regex patterns to match to request\*" section containing two radio button options: "Create regex pattern set" (selected) and "Use saved regex pattern set". At the bottom, there is a "New pattern set name\*" input field with the placeholder "Type a pattern set name" and a "Type a regular expression" input field with a "+" icon. A note at the bottom left says "\* Required". At the bottom right, there are "Cancel" and "Create" buttons.

AWS WAF & Shield

https://console.aws.amazon.com/waf/home#/wizard//newrm/global?isInWizard=true

Most Visited

conditions

Create regex match condition

Filter settings

Specify the settings that you want to use to allow or block web requests. You can only have one filter in a regex match condition, but you can have up to 10 regex patterns in the regex pattern set used in the filter. All patterns within a pattern set will be used in request matching together without priority.

String and regex match conditions

Part of the request to filter on

- Body
- Header
- HTTP method
- Query string
- URI
- Body

Transformation

Regex patterns to match to request\*

- Create regex pattern set
- Use saved regex pattern set

New pattern set name\* Type a pattern set name

Type a regular expression +

\* Required

Cancel Create



# Regex Match Condition

The screenshot shows the AWS WAF & Shield console with a modal dialog titled "Create regex match condition".

**Transformation:** None

**Regex patterns to match to request\***

- Create regex pattern set
- Use saved regex pattern set

New pattern set name\*:

x

Type a regular expression +

**Create pattern set and add filter**

**Filter in this regex match condition**

**Part of the request to filter on**

This condition has no filters.

\* Required

Cancel Create



# Regex Match Condition

The screenshot shows the AWS WAF & Shield console interface. A modal window titled "Create regex match condition" is open in the foreground, overlaid on a darker background where the "String and regex match conditions" section is visible.

**Create regex match condition**

A regex match condition contains a list of the regex patterns that matches particular part in web requests that you want to allow or block. [Learn more](#)

**Name\*** MATCH-BADREQUEST

**Region\*** Global (CloudFront)

Choose Global (CloudFront) to create AWS WAF resources to use with CloudFront distributions in all AWS Regions. Choose a specific AWS Region to create AWS WAF resources to use with an Application Load Balancer in that region.

**Type\*** Regex match

To create a standard string match condition, choose String match. To create a regular expression match condition, choose Regex match.

**Filter in this regex match condition**

You can only have one filter in a regex match condition.

**Part of the request to filter on**

Body matches following patterns in regex pattern set 'badrquest'. ×

I[a@]mAB[a@]dRequest

\* Required

**Cancel** **Create**



# Create a Rule

The screenshot shows the AWS WAF & Shield console with the URL <https://console.aws.amazon.com/waf/home?region=global#/wizard/>. The page title is "Set up a web access control list (web ACL)". On the left, a sidebar lists steps: "Concepts overview", "Step 1: Name web ACL", "Step 2: Create conditions", "Step 3: Create rules" (which is highlighted with a red box), and "Step 4: Review and create". The main content area is titled "Create rules" and contains instructions about rules and conditions. It features two sections: "Add rules to a web ACL" and "If a request matches all of the conditions in a rule, take the corresponding action". The "Create rule" button in the "Add rules to a web ACL" section is also highlighted with a red box. To the right, there is a "Concepts overview" sidebar with examples of rules and their actions.

Rules contain the conditions that you want to use to filter web requests. You add rules to a web ACL, and then specify whether you want to allow or block requests based on each rule. [Learn more](#)

Add rules to a web ACL

Rules Select a rule Add rule to web ACL Create rule

If a request matches all of the conditions in a rule, take the corresponding action

Order	Rule	Action
Create new rule using IP match or string match conditions created in previous step.		

If a request doesn't match any rules, take the default action

Default action\*  Allow all requests that don't match any rules  Block all requests that don't match any rules

\* Required Cancel Previous Review and create

Concepts overview

Web ACL example if requests match

Rule 1, Bad User-Agents, then block

IP match condition Suspicious IPs

and

String match condition Bad bots

or if requests match

Rule 2, Detect SQLi, then block

SQL injection match condition SQLi checks

otherwise, perform the default action

Default action



# Create a Rule

The screenshot shows the AWS WAF & Shield console with the URL <https://console.aws.amazon.com/waf/home?region=global#/wizard//newrule/global>. The interface is a step-by-step wizard for creating a rule.

**Step 1: Name web ACL**  
**Step 2: Create conditions**  
**Step 3: Create rules** (highlighted)  
**Step 4: Review and create**

**Create rule**

**When a request**

- does ▾
- originate from a geographic location in ▾
- US-Geo-Match ▾

**Filters in US-Geo-Match**

- Christmas Island - CX

**And**

**When a request**

- does ▾
- originate from an IP address in ▾
- BlockIP ▾

**IP Addresses in BlockIP**

- 209.200.63.0/24

\* Required

**Create rule**



# Create a Rule

AWS WAF & Shield x + https://console.aws.amazon.com/waf/home?region=global#/wizard/ ... ☆ Search Search Minimize Maximize Close

Most Visited

Concepts overview

Step 1: Name web ACL

Step 2: Create conditions

**Step 3: Create rules**

Step 4: Review and create

## Create rules

Rules contain the conditions that you want to use to filter web requests. You add rules to a web ACL, and then specify whether you want to allow or block requests based on each rule. [Learn more](#)

### Add rules to a web ACL

Rules MY-RULE Add another rule Create rule

Rule created successfully.

If a request matches all of the conditions in a rule, take the corresponding action

Order	Rule	Action
1	MY-RULE	<input type="radio"/> Allow <input checked="" type="radio"/> Block <input type="radio"/> Count <span>x</span>

If a request doesn't match any rules, take the default action

Default action\*  Allow all requests that don't match any rules  Block all requests that don't match any rules

\* Required Cancel Previous Review and create

## Concepts overview

Web ACL example if requests match

Rule 1, Bad User-Agents, then block

IP match condition Suspicious IPs

and

String match condition Bad bots

or if requests match

Rule 2, Detect SQLi, then block

SQL injection match condition SQLi checks

otherwise, perform the default action

Default action

Allow requests that don't match any rules



# Associate Web ACL with Distribution

The screenshot shows the AWS WAF & Shield console interface. The left sidebar has a navigation menu with 'AWS WAF' selected under 'Web ACLs'. The main content area displays a success message: 'Your web ACL was successfully created.' Below this, the 'Web ACLs' section shows a 'Create web ACL' button and a 'Delete' button. A 'Filter' dropdown is set to 'Global (CloudFront)'. A table lists a single entry: 'Name' (MY-WEB-ACL). To the right, the 'MY-WEB-ACL' configuration page is shown. It includes tabs for 'Requests' and 'Rules', with 'Rules' selected. The 'Edit web ACL' button is visible. A table lists one rule: 'Order' (1), 'Rule' (MY-RULE), 'Type' (Regular), and 'Action' (Block requests). Below this, a section for default actions states: 'If a request doesn't match any rules, take the default action' with 'Default action' set to 'Allow all requests that don't match any rules'. At the bottom, the 'AWS resources using this web ACL' section shows a table with no entries, stating 'No resource is using this web ACL.'

Your web ACL was successfully created.

Web ACLs

Create web ACL Delete

Filter Global (CloudFront)

Name

MY-WEB-ACL

MY-WEB-ACL

Requests Rules

If a request matches all of the conditions in a rule, take the corresponding action

Edit web ACL

Order	Rule	Type	Action
1	MY-RULE	Regular	Block requests

If a request doesn't match any rules, take the default action

Default action Allow all requests that don't match any rules

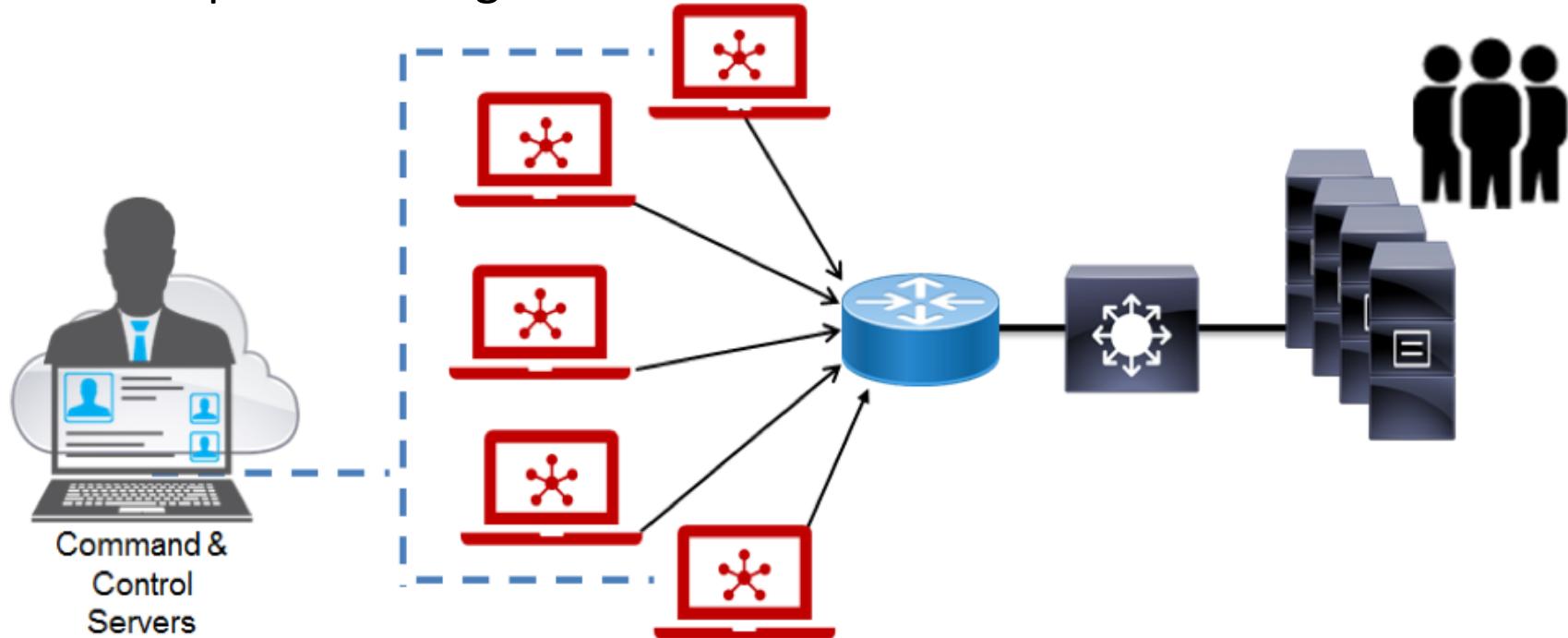
AWS resources using this web ACL

Add association

Resource	Type
No resource is using this web ACL.	

# AWS Shield

- AWS provides AWS Shield Standard and AWS Shield Advanced for protection against DDoS attacks



# AWS Shield Standard

- Included with AWS WAF at no additional cost beyond what you are paying for AWS WAF and your other AWS services
- AWS technologies that are built from the ground up to provide resilience in the face of network and transport layer DDoS attacks
- For web application attacks, you also can use AWS WAF to configure web access control lists (web ACLs) that target network layer DDoS regex request patterns and help to minimize the effects of a DDoS attack

# AWS Shield Advanced

- Provides expanded DDoS attack protection for your Elastic Load Balancing load balancers, CloudFront distributions, and Amazon Route 53 hosted zones
- Includes intelligent DDoS attack detection and mitigation for OSI layers 3 through 7
- You get 24x7 DDoS response team (DRT) assistance during a DDoS attack
- You have exclusive access to advanced, real-time metrics and reports for deep visibility into attacks on your AWS resources

# AWS Shield

The screenshot shows the AWS WAF & Shield console with the AWS Shield page open. The left sidebar lists various AWS services and features under the AWS WAF category. The main content area displays a comparison table between AWS Shield Standard and AWS Shield Advanced.

**AWS Shield**

As an AWS customer, you automatically have basic DDoS protection with the AWS Shield Standard plan, at no additional cost beyond what you already pay for AWS WAF and your other AWS services. For an additional cost, you can get advanced DDoS protection by activating the AWS Shield Advanced plan. The following table shows a comparison of the two plans.

Features	AWS Shield Standard	AWS Shield Advanced
<b>Active monitoring</b>		
Network flow monitoring	✓	✓
Automated application (layer 7) traffic monitoring	-	✓
<b>DDoS mitigations</b>		
Helps protect from common DDoS attacks, such as SYN floods and UDP reflection attacks	✓	✓
Access to additional DDoS mitigation capacity	-	✓
<b>Visibility and reporting</b>		
Layer 3/4 attack notification and attack		

# AWS Shield Threat Landscape Report

- The AWS Shield Threat Landscape Report (TLR) offers a summary of threats detected by AWS Shield
- The report is produced by the AWS Threat Research Team (TRT) that persistently monitors and evaluates the threat landscape to formulate security controls for AWS customers

# AWS Trusted Advisor

- Uses best practices derived from history of serving thousands of AWS customers
- Includes 27 individual checks within 4 categories:



Cost Optimizing



Security



Fault Tolerance



Performance

# AWS Trusted Advisor

Trusted Advisor Manager 

Secure | https://console.aws.amazon.com/trustedadvisor/home?region=us-east-2#/category/security

Dashboard  
Cost Optimization  
Performance  
**Security**  
Fault Tolerance  
Service Limits  
Preferences

## Security

 5  0  1 

### Security Checks

Check	Status	Last Refreshed	Previous Status	Actions
MFA on Root Account		Refreshed: a few seconds ago	Green	 
Amazon EBS Public Snapshots		Refreshed: a few seconds ago	Green	 
Amazon RDS Public Snapshots		Refreshed: a few seconds ago	Green	 
Amazon S3 Bucket Permissions		Refreshed: a few seconds ago	Green	 
IAM Use		Refreshed: a few seconds ago	Green	 
Security Groups - Specific Ports Unrestricted		Refreshed: a few seconds ago	Green	 

 Upgrade your Support plan to unlock all Trusted Advisor recommendations!  
You will have access to technical support from a cloud support engineer, with phone and chat support, support API, Identity and Access Management, Architecture support - use case guidance, and more.

# AWS Trusted Advisor

The screenshot shows three browser tabs for the AWS Trusted Advisor Manager, all displaying the same content. The URL is https://console.aws.amazon.com/trustedadvisor/home?region=us-east-2#/category/security. The content lists various security checks:

- Security Groups - Unrestricted Access**  
Checks security groups for rules that allow unrestricted access to a resource.
- IAM Password Policy**  
Checks the password policy for your account and warns when a password policy is not enabled, or if password content requirements have not been enabled.
- Amazon RDS Security Group Access Risk**  
Checks security group configurations for Amazon Relational Database Service (Amazon RDS) and warns when a security group rule might grant overly permissive access to your database.
- Amazon Route 53 MX Resource Record Sets and Sender Policy Framework**  
For each MX resource record set, checks for a TXT resource record set that contains a corresponding SPF value.
- AWS CloudTrail Logging**  
Checks for your use of AWS CloudTrail.
- ELB Listener Security**  
Checks for load balancers with listeners that do not use recommended security configurations for encrypted communication.
- ELB Security Groups**  
Checks for load balancers configured with a missing security group or a security group that allows access to ports that are not configured for the load balancer.
- CloudFront Custom SSL Certificates in the IAM Certificate Store**  
Checks the SSL certificates for CloudFront alternate domain names in the IAM certificate store and alerts you if the certificate is expired, will soon expire, uses outdated encryption, or is not configured correctly for the distribution.
- CloudFront SSL Certificate on the Origin Server**  
Checks your origin server for SSL certificates that are expired, about to expire, missing, or that use outdated encryption.
- IAM Access Key Rotation**  
Checks for active IAM access keys that have not been rotated in the last 90 days.
- Exposed Access Keys**

# AWS Guard Duty

- Amazon GuardDuty is a managed threat detection service that continuously monitors for malicious or unauthorized behavior
- It monitors for activity such as unusual API calls or potentially unauthorized deployments that indicate a possible account compromise (Zero Days)
- GuardDuty also detects potentially compromised instances or reconnaissance by attackers.
- Uses proprietary ML and AI along with strategic partners

# AWS Guard Duty

- When GuardDuty detects suspicious or unexpected behavior it generates a finding - a notification that has the details about a impending security issue
- The finding details include information about what occurred, what AWS resources were involved in the suspicious activity, when this activity took place, and other data
- The finding type provides a description of the potential security issue: ***Recon:EC2/PortProbeUnprotectedPort***
- [docs.aws.amazon.com](https://docs.aws.amazon.com): “Amazon GuardDuty Finding Types”

# Enabling Security Hub

- Security Hub offers a consolidated view of your security status in AWS
- You can automate security checks, manage security findings, and classify the highest priority security issues across your AWS environment using:
  - Amazon GuardDuty
  - Amazon Inspector
  - S3 bucket policy findings from Amazon Macie
  - Plus integrated partner solutions



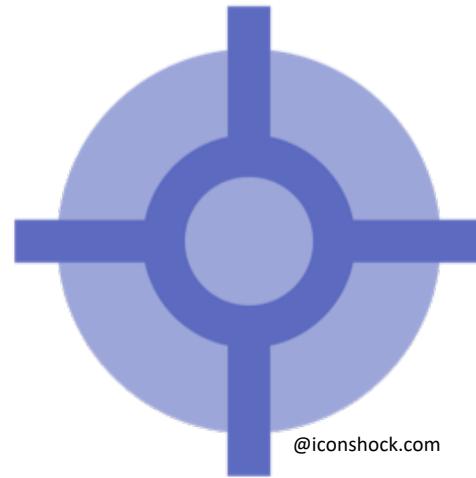
@iconshock.com

# Security Hub Use Cases

- Continuously scan AWS accounts for configuration errors using:
  - **Center for Internet Security (CIS) AWS Foundations benchmarks**
  - **PCI DSS v3.2.1 benchmarks**
  - **AWS Foundational Best Practices Standards**
- Report on security check results at the account and multi-account level to recognize your global security posture
- Use the Hub's summary dashboards and filtering rules to identify and prioritize which findings

# Enabling Security Hub

- Enabling Security Hub grants it permissions to import findings from:
  - Amazon GuardDuty
  - Amazon Inspector
  - Amazon Macie
  - AWS IAM Access Analyzer
  - AWS Firewall Manager
- **AWS offers a 30-day free trial**



@iconshock.com



## Segment 5: Key Management Service (KMS), Service-Specific Security (S3 and EBS), and Organizational Service Control Policies (SCP)

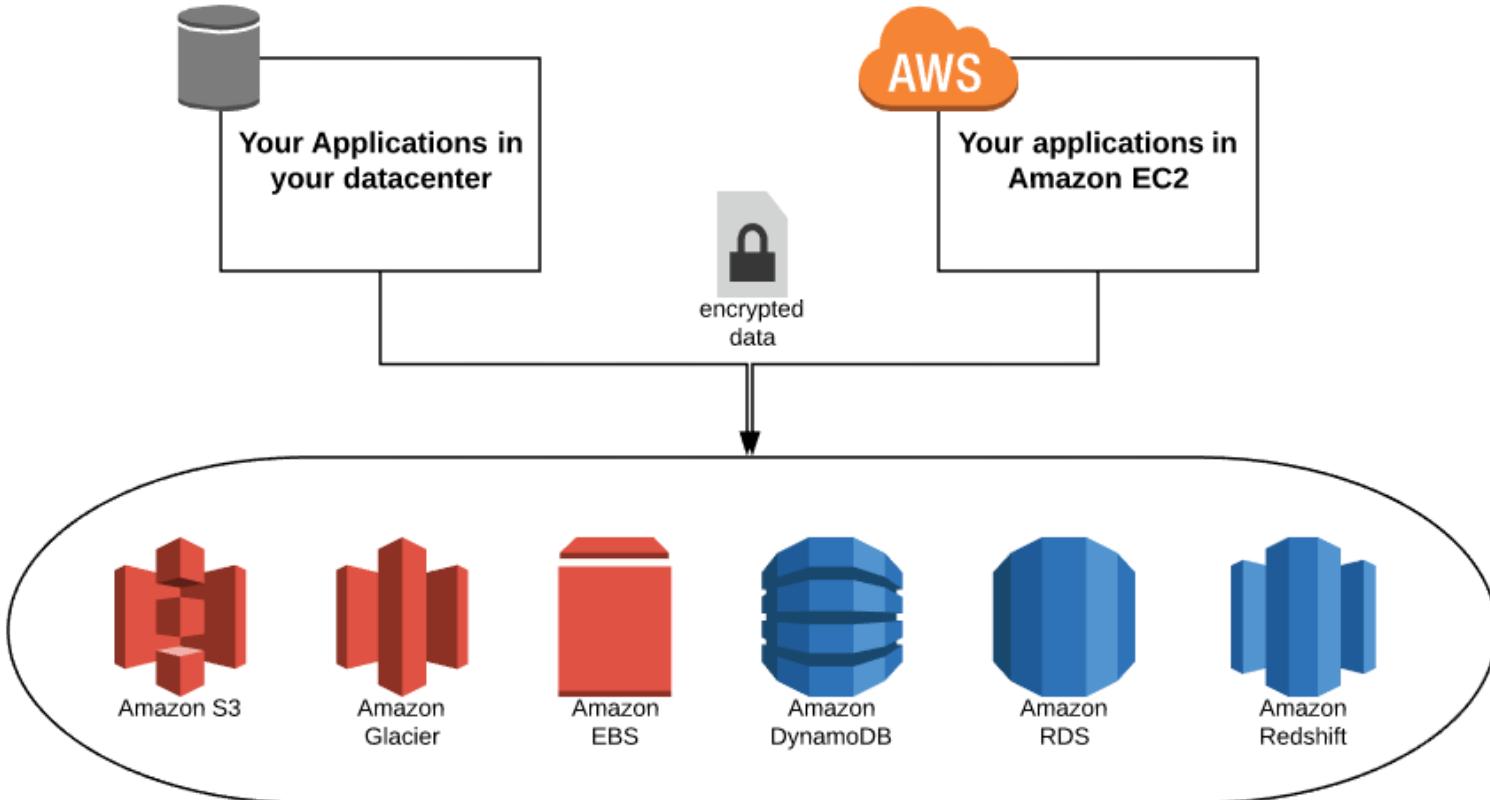
# Encryption and Key Management in AWS

- Client-side encryption: You encrypt your data and manage your own keys
- Server-side encryption: AWS encrypts data and manages the keys for you
- Key Management
  - On your own
  - AWS Management Key Service (KMS)
  - AWS Partner Solutions (Sophos, Trend, etc.)
  - AWS Cloud HSM

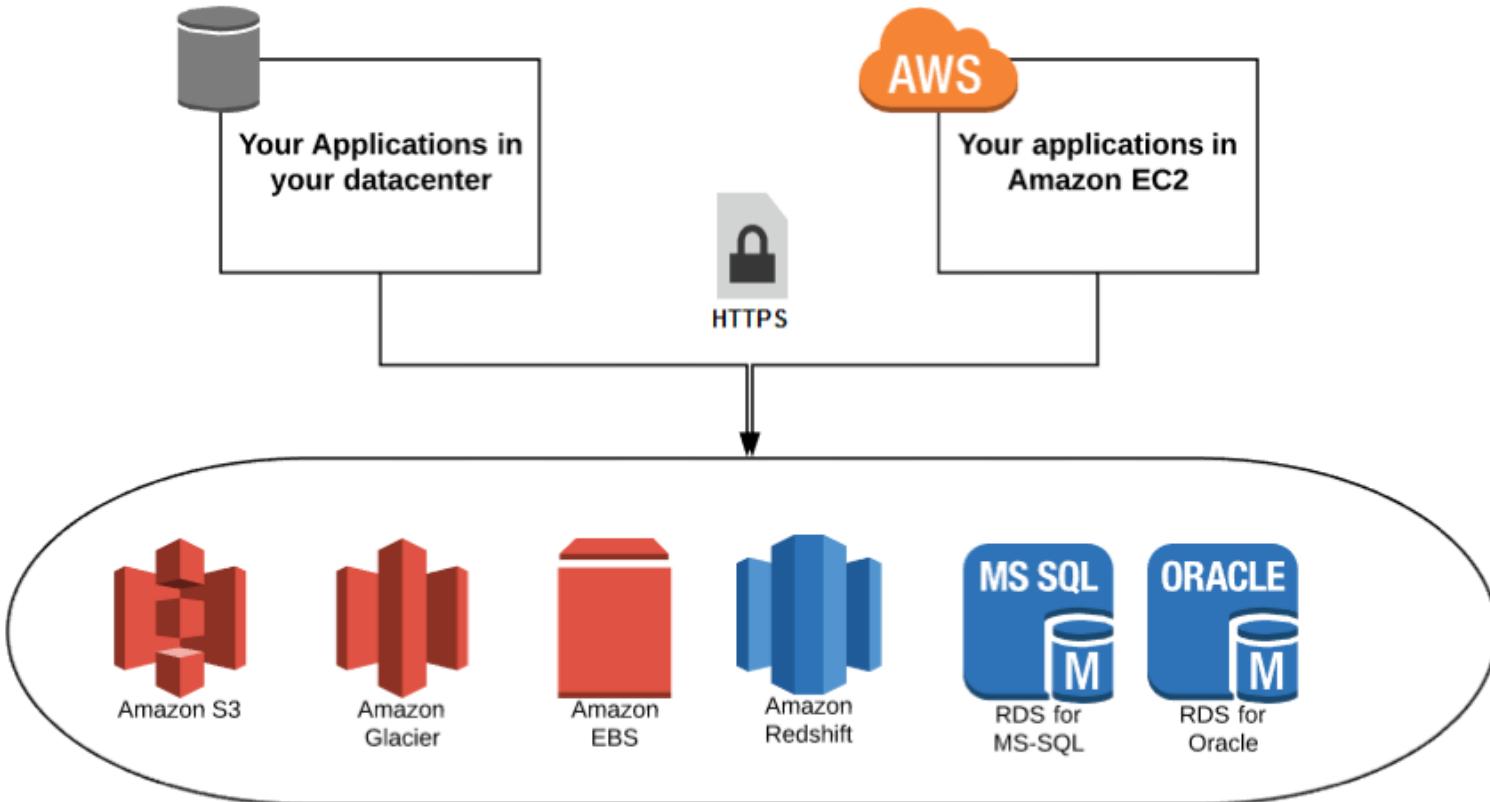


@iconshock.com

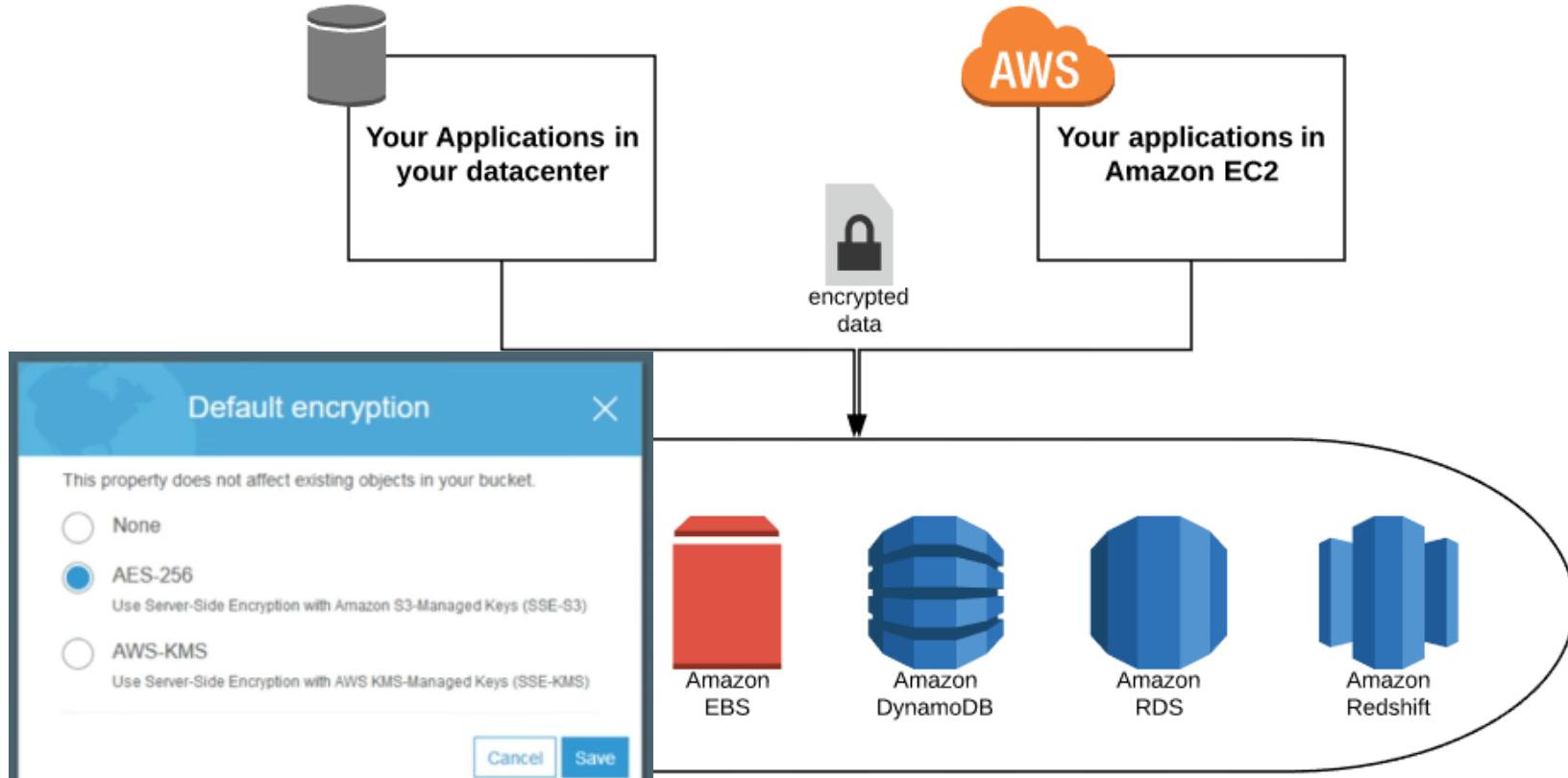
# Client-side Encryption



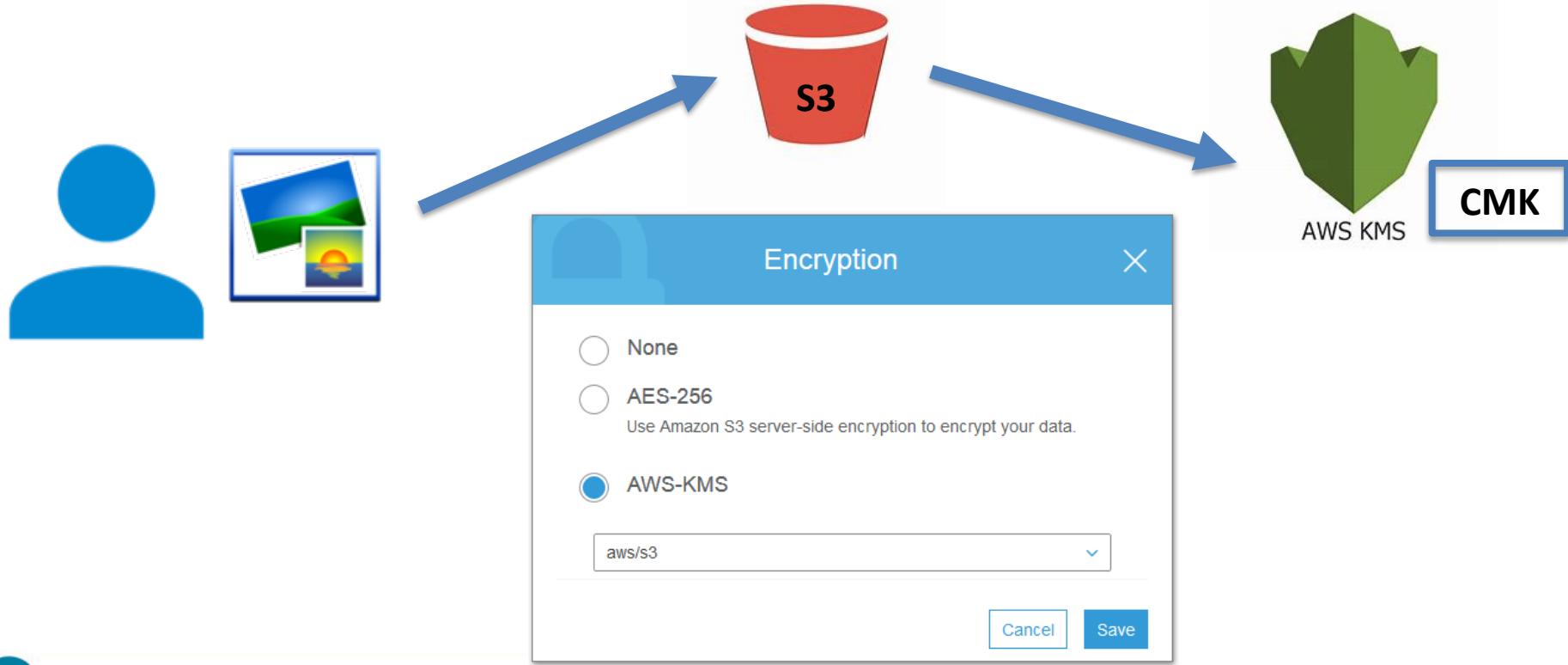
# Server-side Encryption



# AWS S3 Server-side Encryption



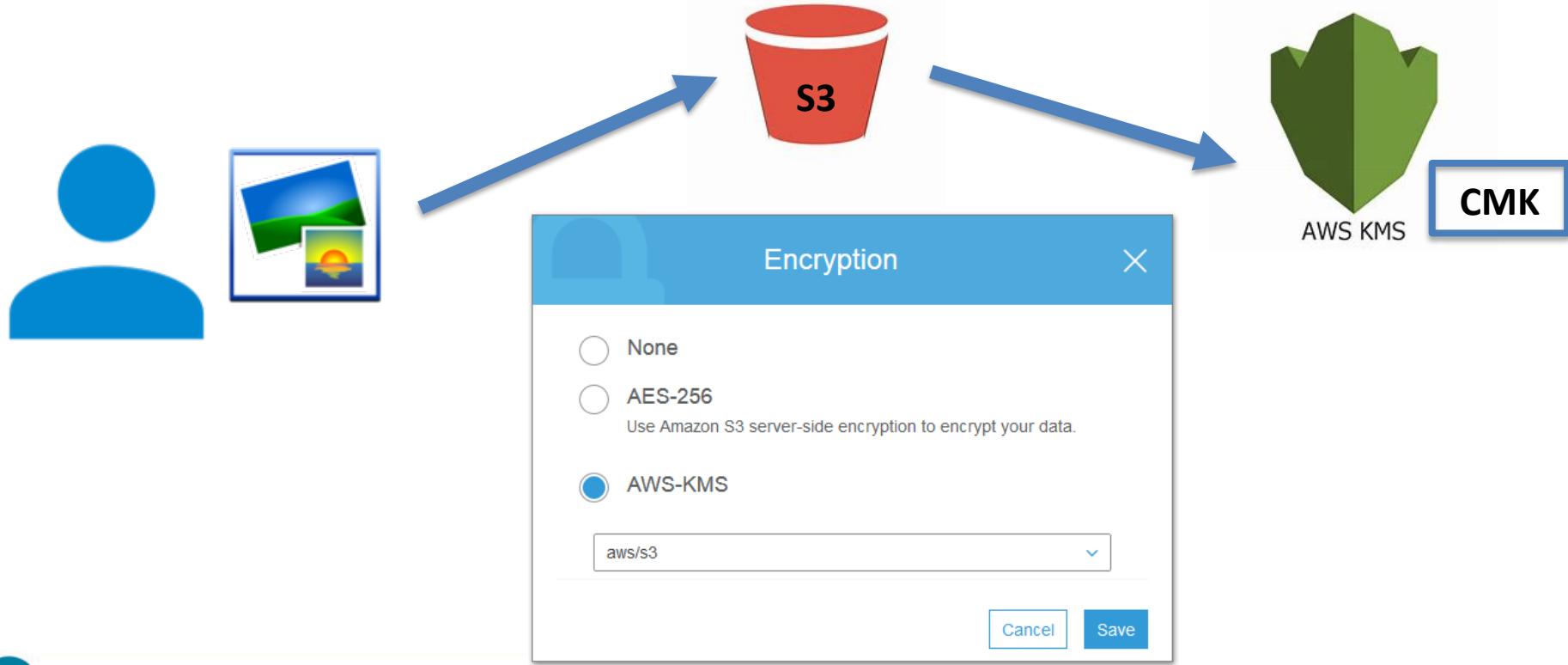
# S3 SSE-KMS Encryption



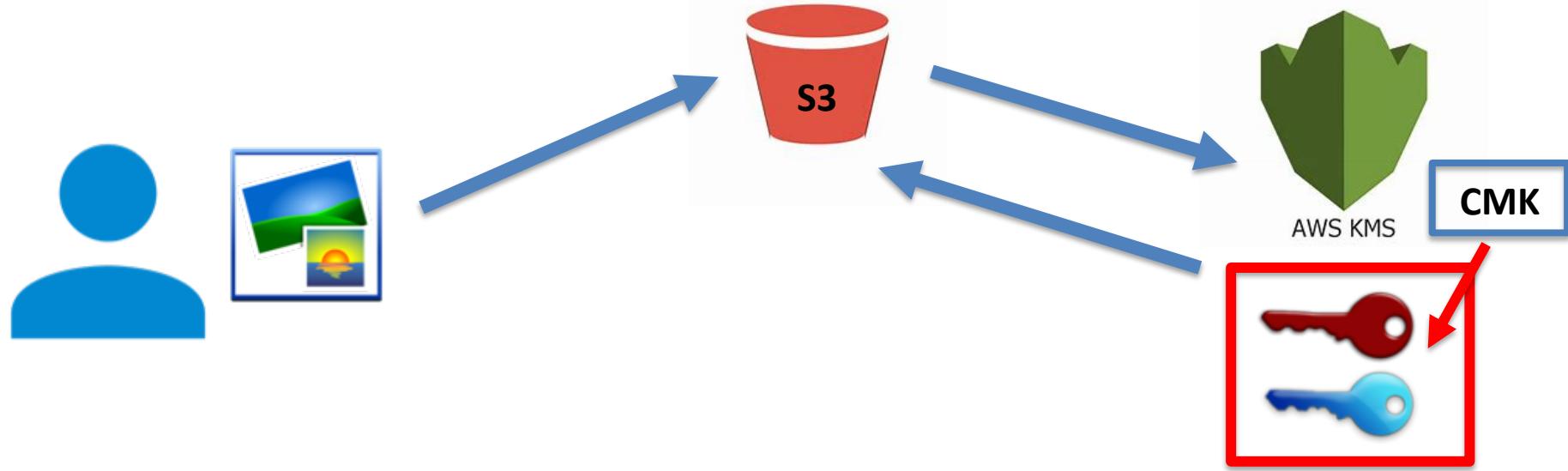
# Types of CMKs

- There are three types of CMKs in AWS accounts: customer managed CMKs, AWS managed CMKs, and AWS owned CMKs
  - Customer managed CMKs are CMKs in your AWS account that you create, own, and manage
  - AWS managed CMKs are CMKs in your account that are created, managed, and used on your behalf by an AWS service that integrates with AWS KMS
  - AWS owned CMKs are not in your AWS account. They are part of a collection of CMKs that AWS owns and manages for use in multiple AWS accounts - AWS services can use AWS owned CMKs to protect your data

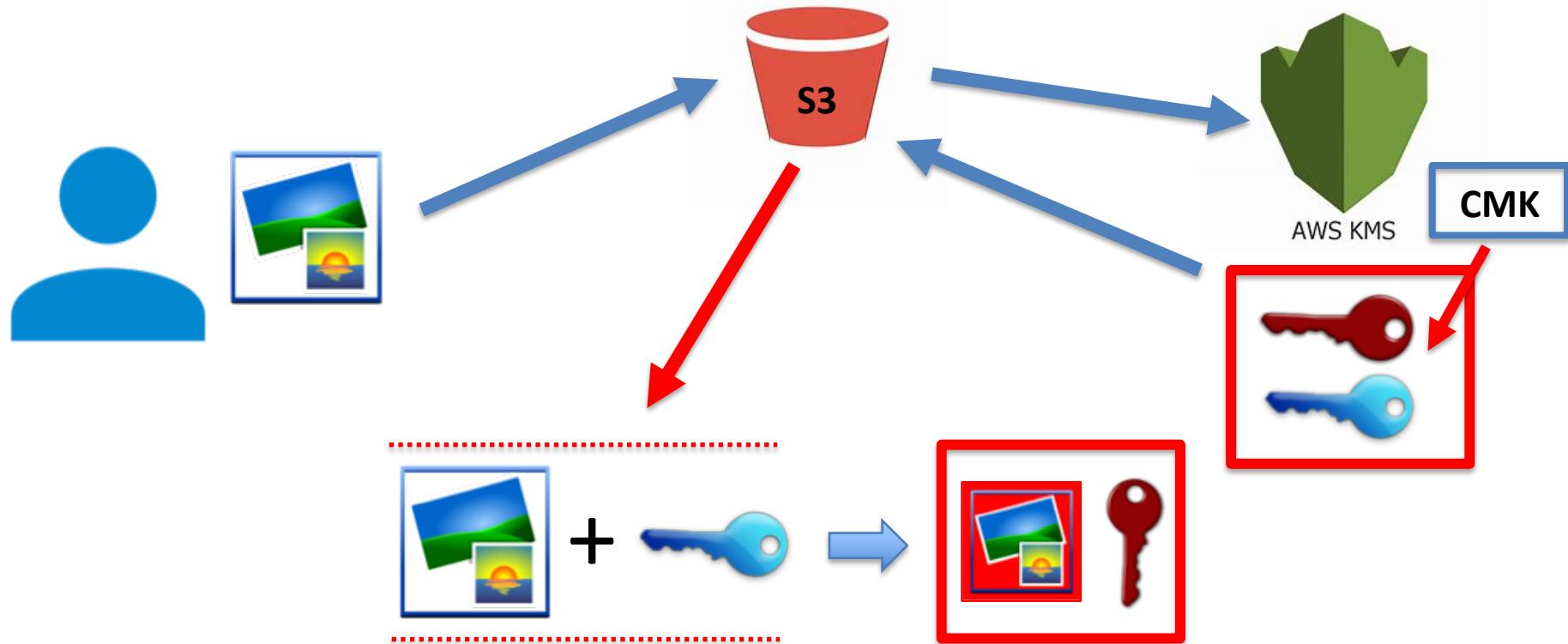
# S3 SSE-KMS Encryption



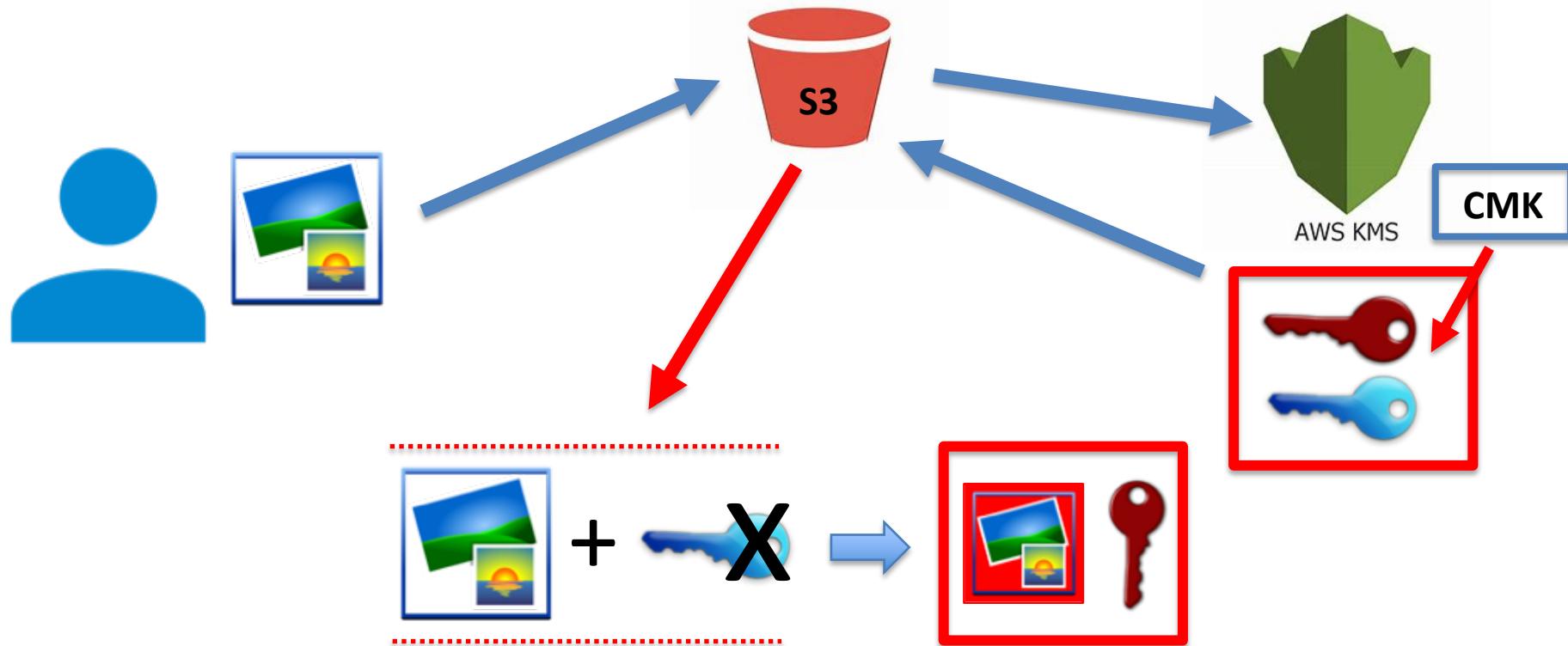
# S3 SSE-KMS Encryption



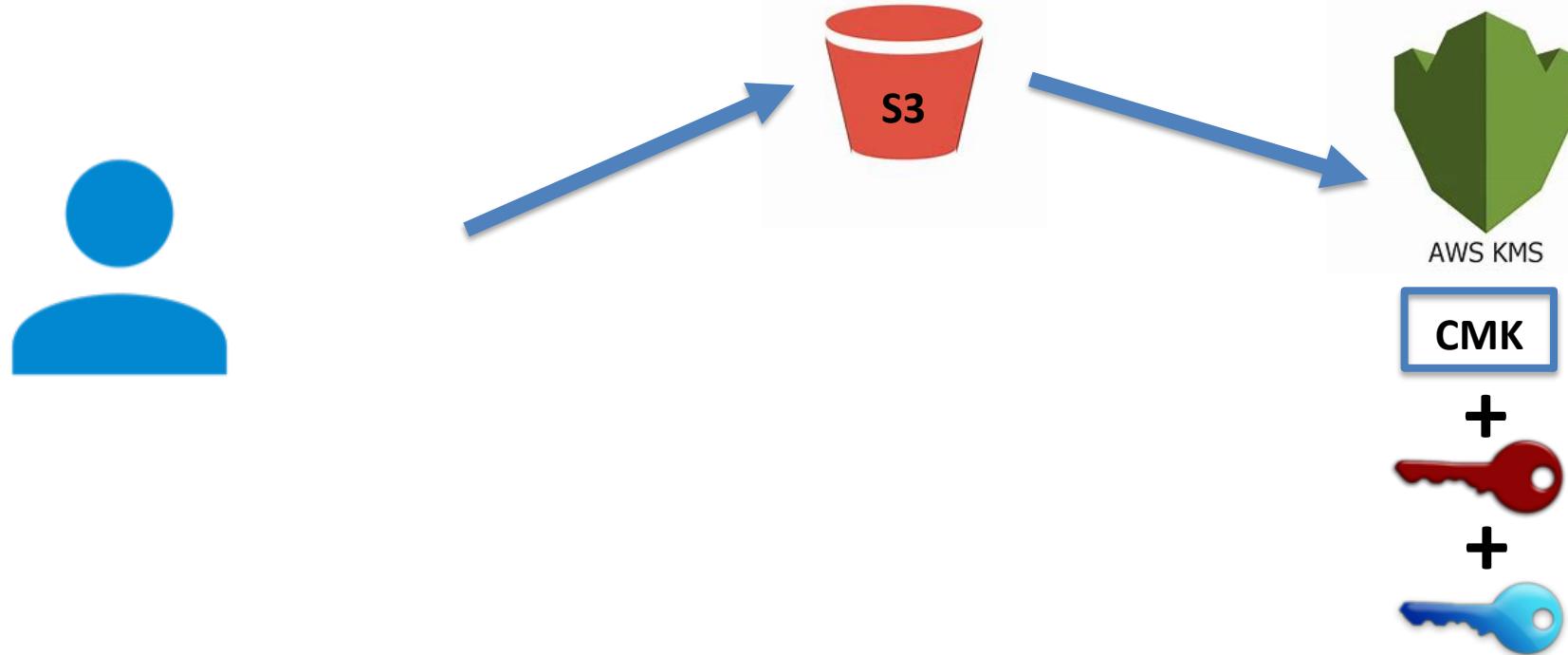
# S3 SSE-KMS Encryption



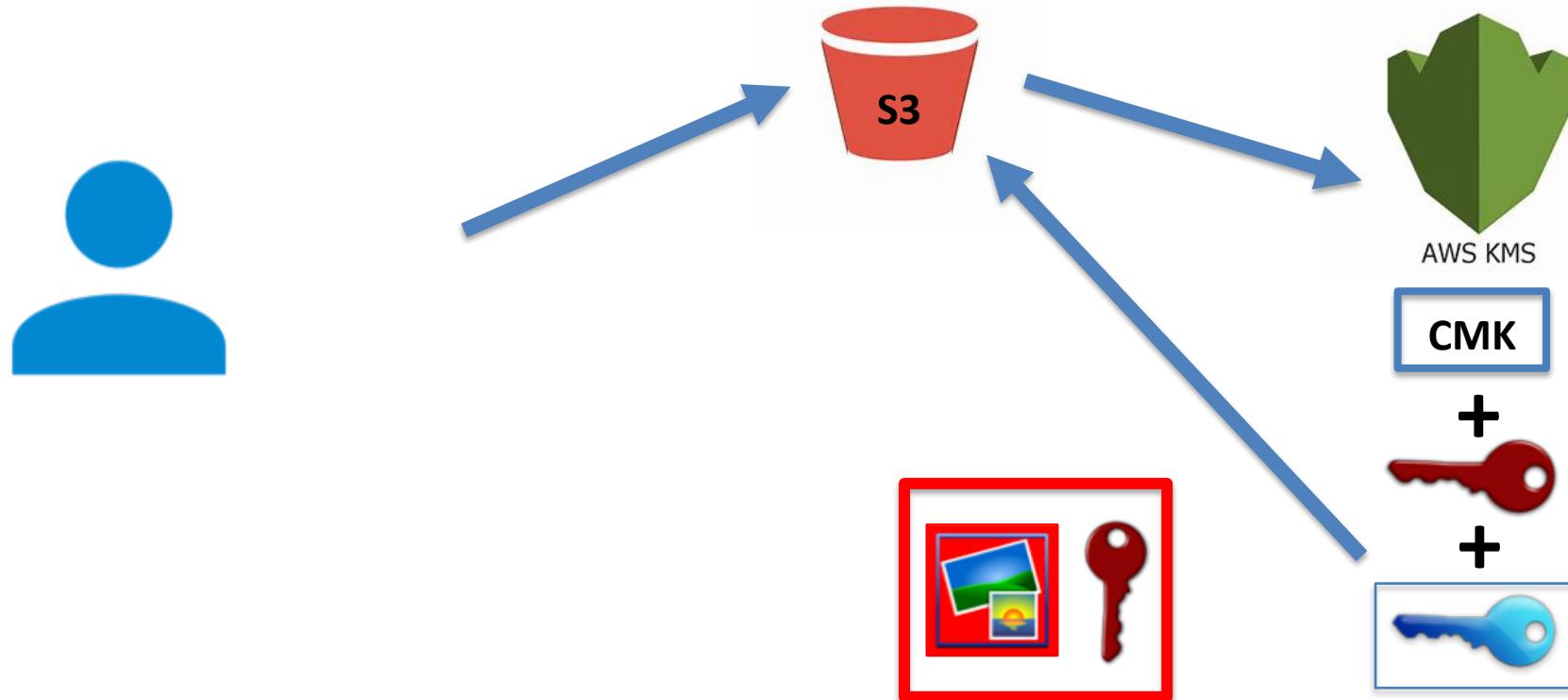
# S3 SSE-KMS Encryption



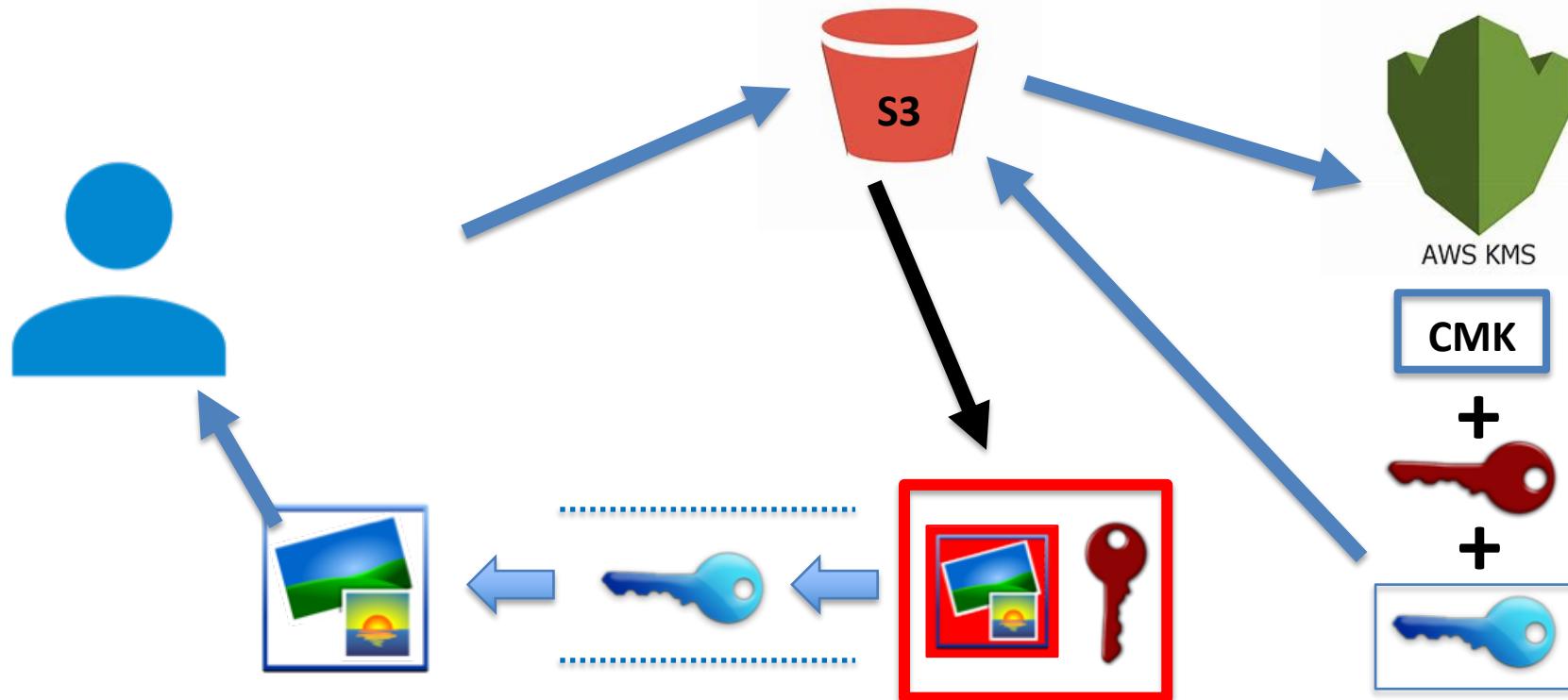
# S3 SSE-KMS Decryption



# S3 SSE-KMS Decryption



# S3 SSE-KMS Decryption



# AWS EBS Encryption

- When you create an encrypted EBS volume and attach it to a supported instance type, the following types of data are encrypted:
  - Data at rest inside the volume
  - All data moving between the volume and the instance
  - All snapshots created from the volume
  - All volumes created from those snapshots
- You can encrypt both the boot and data volumes of an EC2 instance



@iconshock.com

# AWS EBS Encryption

Volume Type <i>(i)</i>	Device <i>(i)</i>	Snapshot <i>(i)</i>	Size (GiB) <i>(i)</i>	Volume Type <i>(i)</i>	IOPS <i>(i)</i>	Throughput (MB/s) <i>(i)</i>	Delete on Termination <i>(i)</i>	Encryption <i>(i)</i>
Root	/dev/xvda	snap-04a92f3aceecdabef	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted
EBS	/dev/sdb	Search (case-insensit	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input type="checkbox"/>	Not Encrypted

Add New Volume

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

Filter by attributes

KMS Key Aliases	KMS Key ID
Not Encrypted	
(default) aws/ebs	alias/aws/ebs

# AWS EBS Encryption by Default

- You can enable the EBS Encryption by Default feature
  - AWS encrypts new EBS volumes on launch
  - AWS encrypts new copies of unencrypted snapshots
- Newly created EBS resources are encrypted to your account's default CMK unless you specify a custom CMK in the EC2 settings or at instance launch

```
aws ec2 enable-ebs-encryption-by-default
```

# Simple Storage Service (Amazon S3)

- **Amazon Simple Storage Service (Amazon S3)** is an object storage service that offers industry-leading scalability, data availability, security, and performance
- Amazon S3 provides easy-to-use management features so you can organize your data and configure finely-tuned access controls to meet your specific business, organizational, and compliance requirements
- S3 is designed for 99.99999999% (11 9's) of durability, and stores data for millions of applications for companies all around the world

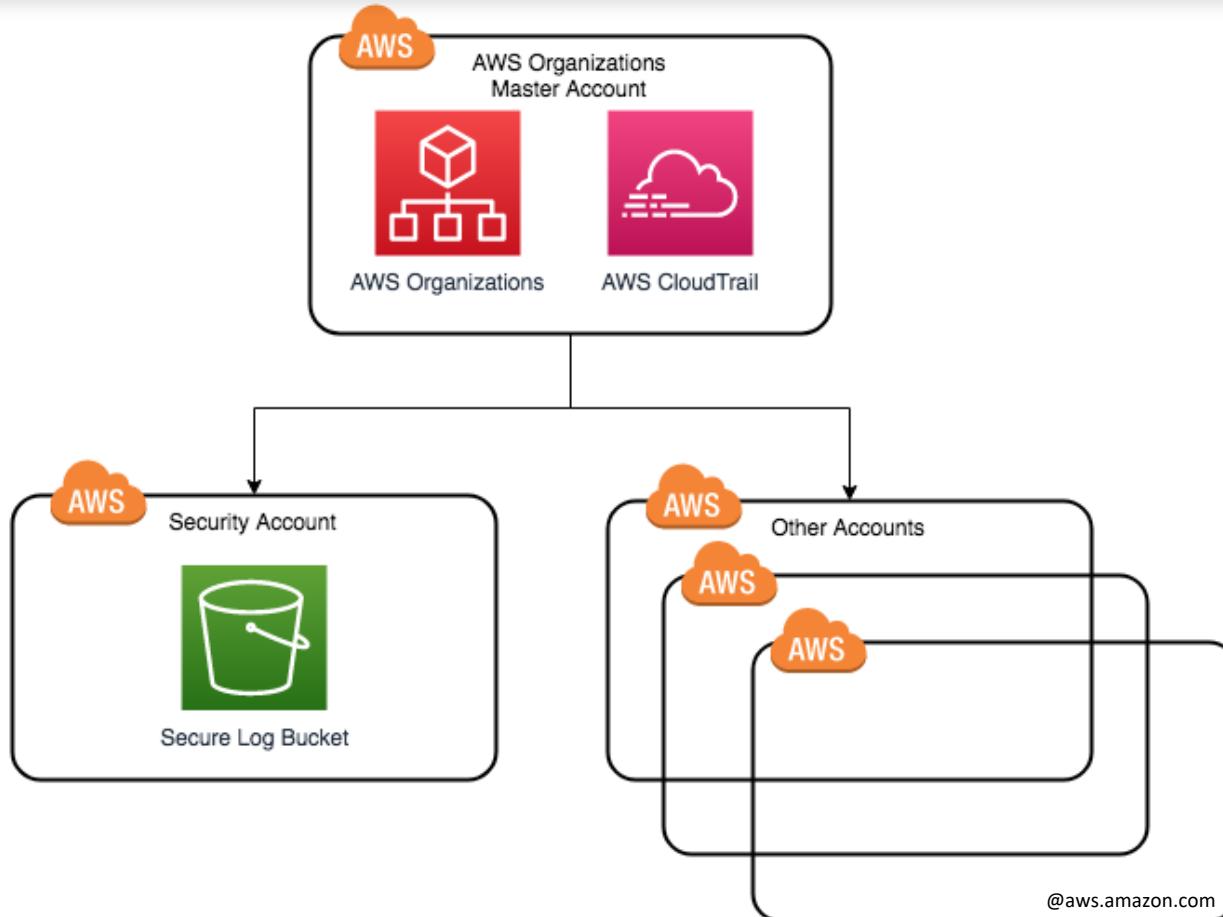
# S3 Security Distinctives

- Amazon S3 standard storage offers the following features:
  - Backed with the Amazon S3 Service Level Agreement
  - Designed to provide 99.99999999% durability and 99.99% availability of objects over a given year
  - Designed to sustain the concurrent loss of data in two facilities
  - Amazon S3 further protects your data using versioning
  - Deploy VPC endpoints for accessing Amazon S3
  - Consider using Amazon Macie with Amazon S3

# Create a Data Bunker

- A data bunker is a secure account which stores critical security data in a secure location
- Only select members of your security team should have access to this account
- Security teams should:
  - Create a new security account in a multi-account organization
  - Create a secure S3 bucket in that account
  - Turn on CloudTrail for the organization and send the logs to this bucket in the secure data account
- You may want to also consider what other data you need to store there (i.e. secure backups)

# Create a Data Bunker



# Amazon Elastic Load Balancing Security

- ELB takes over the encryption and decryption work from the Amazon EC2 instances and manages it centrally on the load balancer
- Offers clients a single point of contact, and can also serve as the first line of defense against attacks on your network



@iconshock.com

# ELB Secure Listener



Services ▾

Resource Groups ▾



Support ▾

1. Define Load Balancer

2. Assign Security Groups

3. Configure Security Settings

4. Configure Health Check

5. Add EC2 Instances

6. Add Tags

7. Review

## Step 3: Configure Security Settings



**Improve your load balancer's security. Your load balancer is not using any secure listener.**

If your traffic to the load balancer needs to be secure, use either the HTTPS or the SSL protocol for your front-end connection. You can go back to the first step to add/configure secure listeners under [Basic Configuration](#) section. You can also continue with current settings.

# Use Certificate Manager with ELB

The screenshot shows the AWS Certificate Manager interface. The title bar says "AWS Certificate Manager". The URL in the address bar is <https://us-east-2.console.aws.amazon.com/acm/home?region=us-east-2#/wizard/?firstrun=true>. The page title is "Request a certificate". On the left, a sidebar lists steps: "Step 1: Add domain names" (highlighted with a blue border), "Step 2: Select validation method", "Step 3: Review", and "Step 4: Validation".

The main content area has two sections:

- A top section with a message: "You can use AWS Certificate Manager certificates with other AWS Services." followed by a link to "AWS Services".
- A bottom section with a message: "Choose **Import a certificate** to import an existing certificate instead of requesting a new one. [Learn more.](#)" followed by a "Import a certificate" button.

Below these sections is a heading "Add domain names" with a question mark icon. A text instruction says: "Type the fully qualified domain name of the site you want to secure with an SSL/TLS certificate (for example, www.example.com). Use an asterisk (\*) to request a wildcard certificate to protect several sites in the same domain. For example: \*.example.com protects www.example.com, site.example.com and images.example.com.".

There is a table-like structure for adding domain names:

Domain name*	Remove
www.trainologie.com	X
trainologie.com	X
*.trainologie.com	X

Below this is a blue "Add another name to this certificate" button. A note below it says: "You can add additional names to this certificate. For example, if you're requesting a certificate for "www.example.com", you might want to add the name "example.com" so that customers can reach your site by either name. [Learn more.](#)".

At the bottom, there is a note: "\*At least one domain name is required". On the right, there are "Cancel" and "Next" buttons.

# Use Certificate Manager with ELB

The screenshot shows the AWS Certificate Manager interface. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and user shankhantoo. The main content area is titled "Request a certificate". On the left, a sidebar lists steps: Step 1: Add domain names, Step 2: Select validation method (which is highlighted with an orange border), Step 3: Review, and Step 4: Validation.

**Select validation method**

Choose how AWS Certificate Manager (ACM) validates your certificate request. Before we issue your certificate, we need to validate that you own or control the domains for which you are requesting the certificate. ACM can validate ownership by using DNS or by sending email to the contact addresses of the domain owner.

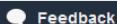
**DNS validation**

Choose this option if you have or can obtain permission to modify the DNS configuration for the domains in your certificate request. [Learn more](#).

**Email validation**

Choose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request. [Learn more](#).

Cancel [Previous](#) [Review](#)



Feedback



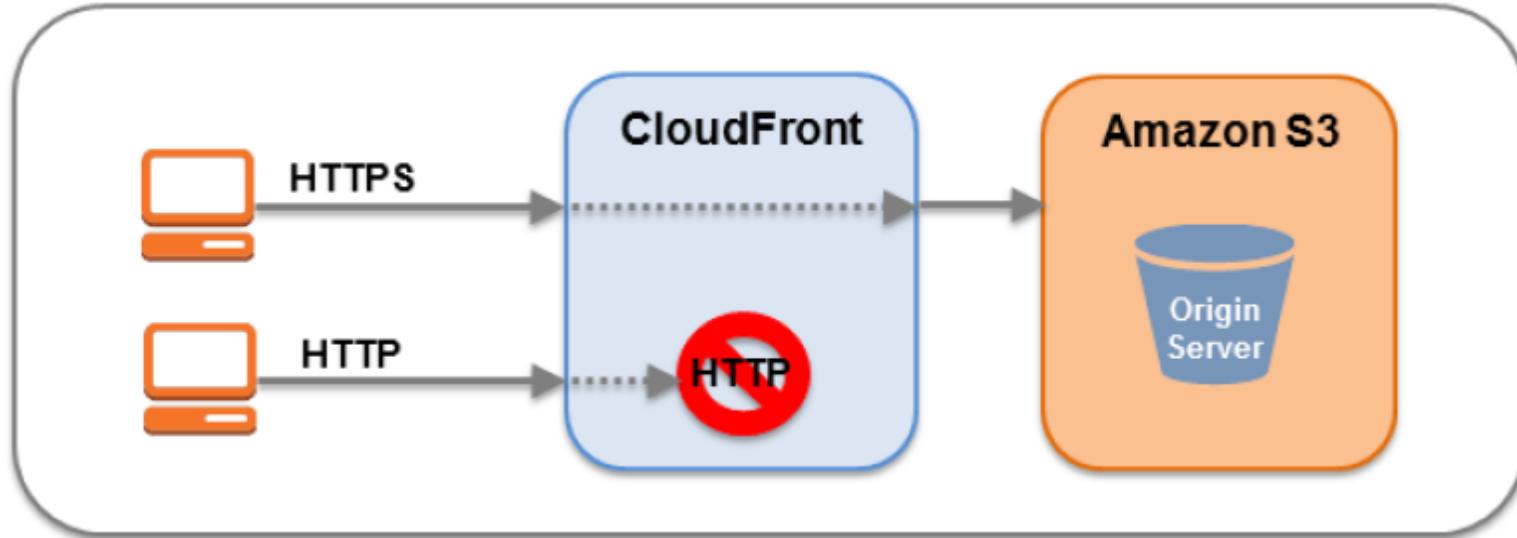
English (US)

© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. [Privacy Policy](#) [Terms of Use](#)



Pearson

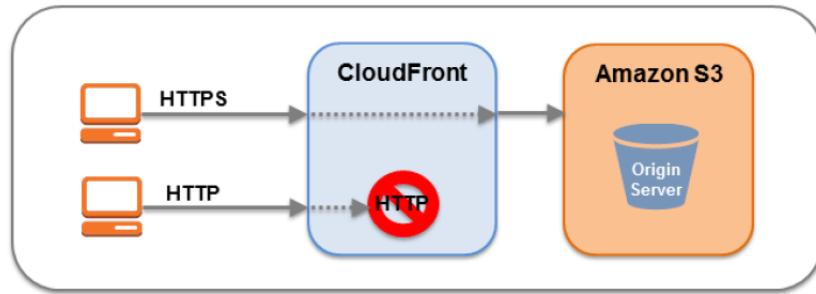
# Amazon CloudFront Security



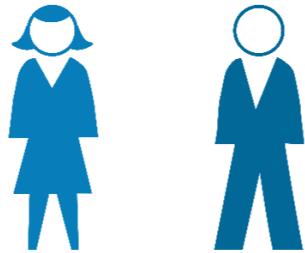
- Every request made to its control API be authenticated – signed with an HMAC-SHA1 signature only accessible through TLS-enabled endpoints
- Private Content Feature controls who can download content from CloudFront
- Origin Access Identities can control access to original copies of objects

# Amazon CloudFront Security

- Amazon CloudFront supports the TLSv1.1 and TLSv1.2 protocols for HTTPS connections between CloudFront and your custom origin webserver
- Selection of cipher suites includes ECDHE protocol on connections to both viewers and the origin



# Default Access to AWS Resources



Employees  
Get AWS IAM  
user accounts  
Assigned to  
Groups or Roles

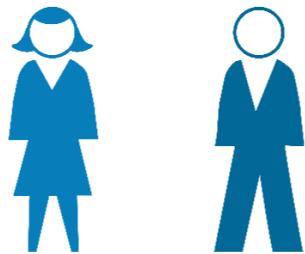


Permissions  
Through  
Managed  
Policies

AWS Account Resources



# Single-Sign-On Access to AWS



If users have a large number of IAM accounts, consider SAML 2.0 federation to enable single sign-on (SSO)



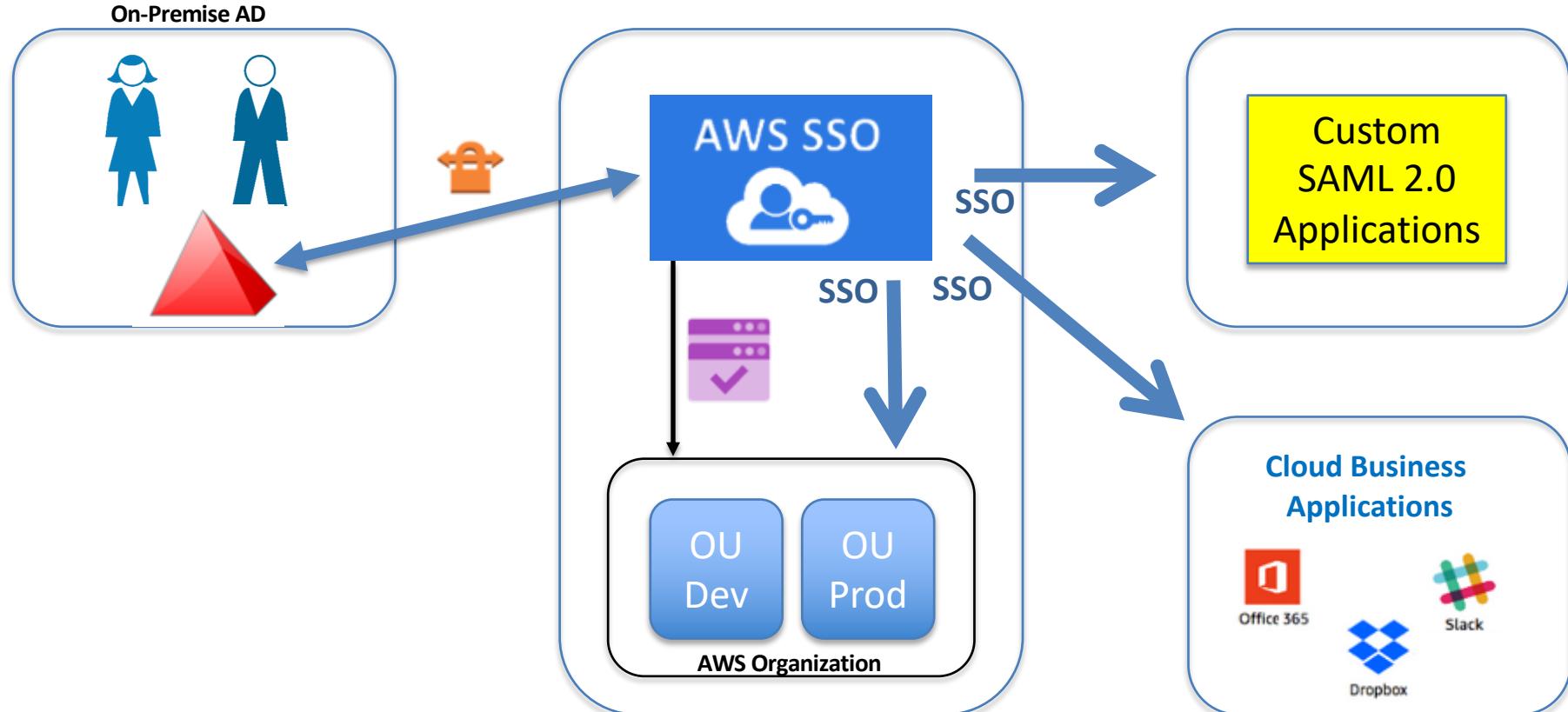
**SAML 2.0**

AWS Account Resources

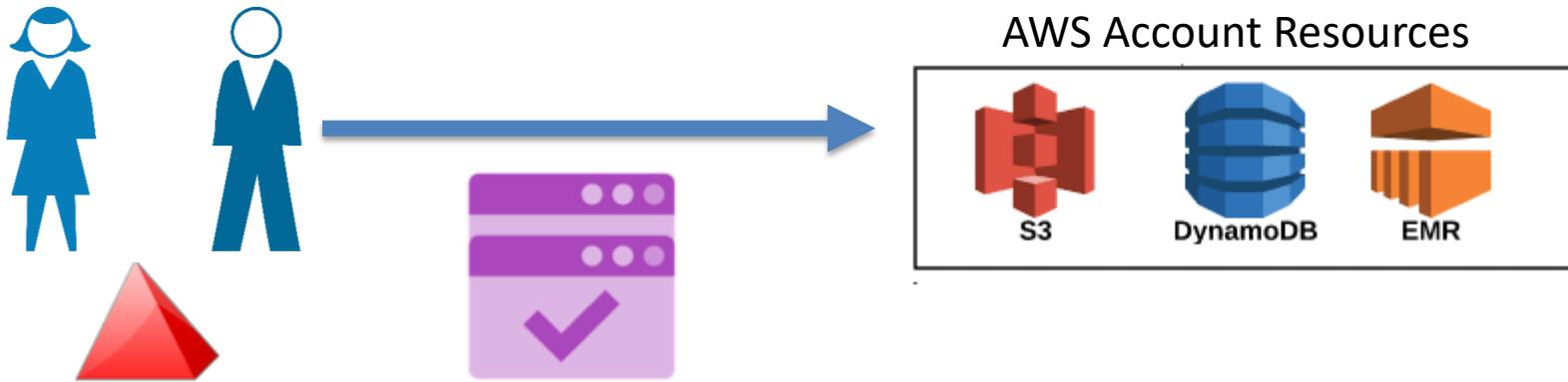


*“Choose Your Own SAML Adventure: A Self-Directed Journey to AWS Identity Federation Mastery” at AWS*

# AWS Single Sign-On (SSO)



# Single-Sign-On Access to AWS



Corporate Microsoft  
Active Directory

AWS SSO also integrates with  
Microsoft Active Directory (AD)  
through AWS Directory Service

# AWS SSO Access to Resources

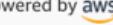
Your applications

Hi John | [Sign out](#)

Search

 AWS Management Console (3)	 Dropbox	 Office365	 Slack
650 ( 650 Account)			>
680 ( 680 Account)			>
903 ( 903 Account)			<
SecurityAudit			

[Terms of Use](#)

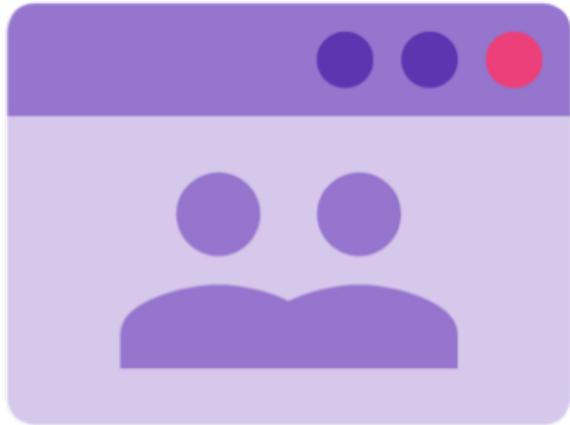
Powered by 

# Amazon Cognito

- Amazon Cognito provides user pools and identity pools
- User pools are custom directories that provide sign-up and sign-in options for your app users
- User Pools let you quickly, easily and securely augment sign-up and sign-in features to mobile and web apps
- It's a fully-managed service that scales to support hundreds of millions of users

# Amazon Cognito

- Identity pools offer AWS credentials to give your users access to other AWS services
- With Cognito Identity Pools, your app can get temporary credentials to access AWS services for anonymous guest users or for users who have signed in



# Cognito Identity Pools Providers

## ▼ Authentication providers ⓘ

Amazon Cognito supports the following authentication methods with Amazon Cognito Sign-In or any public provider. If you allow your users to authenticate using any of these public providers, you can specify your application identifiers here. Warning: Changing the application ID that your identity pool is linked to will prevent existing users from authenticating using Amazon Cognito. [Learn more about public identity providers.](#)

Cognito

Amazon

Apple

Facebook

Google+

Twitter / Digits

OpenID

SAML

Custom

Configure your Cognito Identity Pool to accept users federated with your Cognito User Pool by supplying the User Pool ID and the App Client ID.

User Pool ID

ex: us-east-1\_Ab129faBb



App client id

ex: 7lhkkfbfb4q5kpp90urffao

Add Another Provider

\* Required

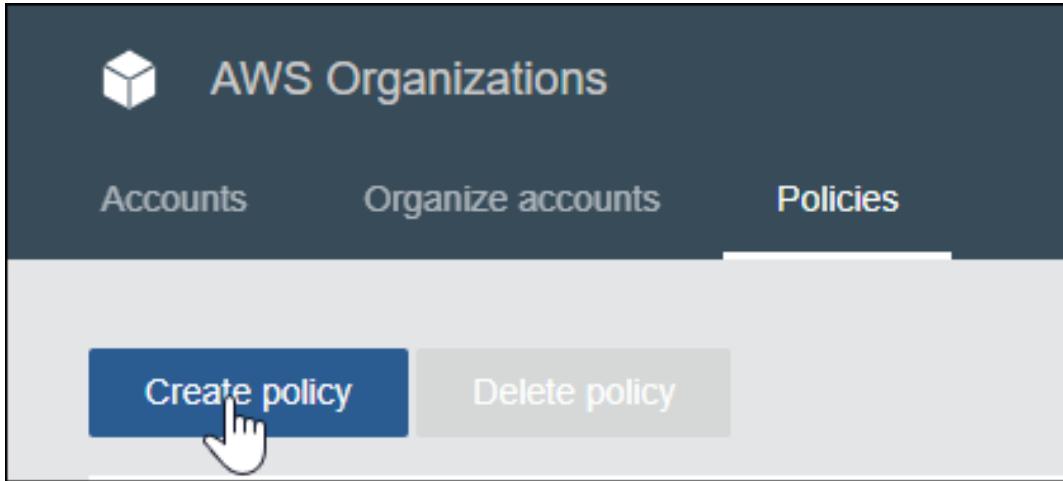
Cancel

Create Pool

# AWS Organizations

- AWS Organizations provide policy-based management for multiple AWS accounts
  - Create groups of accounts
  - Automate account creation
  - Apply and manage policies for account groups
- Can also use Organizations to automate the creation of new accounts through APIs
- Organizations centrally manage Service Control Policies (SCPs) across multiple accounts without using custom scripts or manual processes

# SCP Guardrails



# SCP Guardrails

## Create new policy

A service control policy (SCP) defines the maximum permissions for account users and roles. An SCP doesn't grant permissions. [Learn more](#)

**Policy name \***

DenyChangesToAdminRole

The policy name can have up to 128 characters.

**Description**

Prevents all IAM principals from making changes to AdminRole.

The description can have up to 512 characters. You can't edit the description later.

# SCP Guardrails

The screenshot shows the AWS IAM Policy Editor interface for creating an SCP (Server-side Condition Policy). The policy is titled "'Statement1' statement".

**1. Select service to add actions**: A dropdown menu on the left lists various AWS services. A red box highlights the "Filter Services" input field and the list of services.

**2. Add actions here**: A red arrow points to the JSON code editor area, specifically to the "Action" section of the policy statement. The JSON code is as follows:

```
1 * {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "Statement1",
6       "Effect": "Deny",
7       "Action": [
8         "iam:*"
9       ],
10      "Resource": []
11    }
12  ]
13 }
```

**3. Add resources and conditions here**: A red arrow points to the bottom navigation bar, which includes links for "Add Resource" and "Add Condition".

# SCP Guardrails

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Sid": "DenyChangesToAdminRole",  
6       "Effect": "Deny",  
7       "Action": [],  
8       "Resource": []  
9     }  
10    ]  
11 }
```

# SCP Guardrails

```
1 {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Sid": "Statement1",  
6             "Effect": "Deny",  
7             "NotAction": [  
8                 "iam:GetContextKeysForPrincipalPolicy",  
9                 "iam:GetRole",  
10                "iam:GetRolePolicy",  
11                "iam>ListAttachedRolePolicies",  
12                "iam>ListInstanceProfilesForRole",  
13                "iam>ListRolePolicies",  
14                "iam>ListRoleTags"  
15            ],  
16            "Resource": []  
17        }  
18    ]  
19}
```

# SCP Guardrails

DenyChangesToAdminRole was created. ×

	Create policy	Delete policy
Policy name	FullAWSAcc...	Service control
DenyChang...	DenyChangesToAdminRole	Service control

# AWS Security Crash Course



Michael J.  
Shannon

THANK YOU FOR  
ATTENDING!

