AWS Security Essentials

Course Supplement

Trent Hayes, Course Author

ENTER

Table of Contents

BACK

Secure Global Infrastructure and Compliance

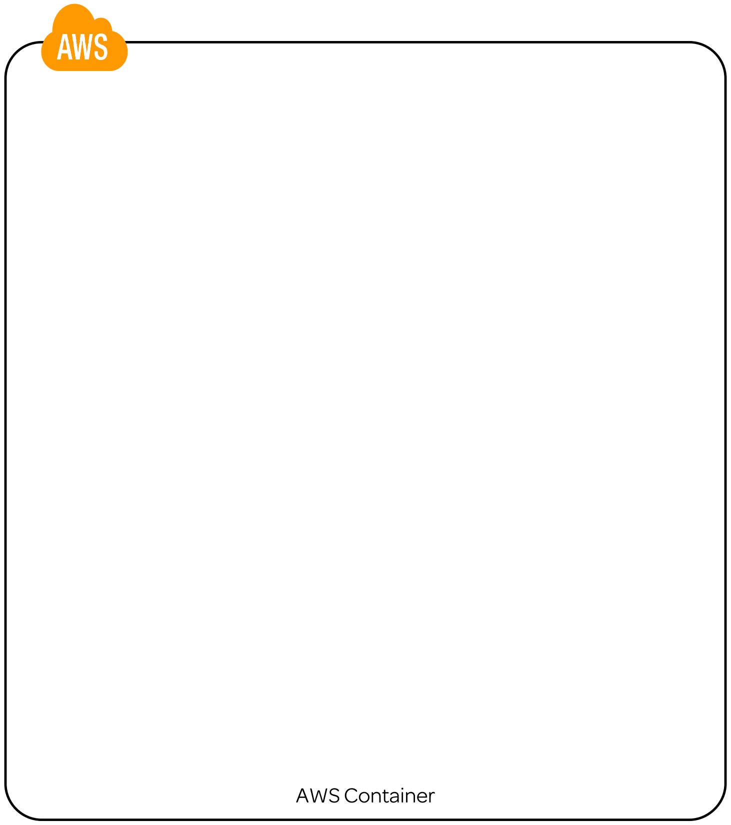
Region

Availability Zones

Endpoints

IAM

Compliance



BACK

Secure Global Infrastructure and Compliance

Region | Availability Zones | Endpoints | IAM | Compliance

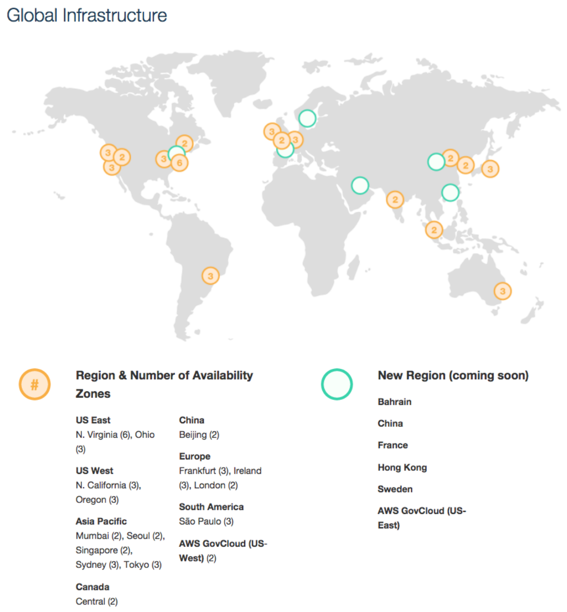
AWS Regions

- The largest organizational unit in AWS
- Represent a geographic area that houses two or more independent AWS data centers

As it relates to security:

- Stored data in a specific region is not replicated to another region by default
- Use regions to:
  - manage compliance with regulations
  - manage network latency

As of October 2017, there are 16 regions with more being released in the near future. Click the diagram below to enlarge.



Global Infrastructure

Region & Number of Availability Zones

US East
N. Virginia (6), Ohio (3)

US West
N. California (3), Oregon (3)

Asia Pacific
Mumbai (2), Seoul (2), Singapore (2), Sydney (3), Tokyo (3)

Canada
Central (2)

China
Beijing (2)

Europe
Frankfurt (3), Ireland (3), London (2)

South America
São Paulo (3)

AWS GovCloud (US-West) (2)

New Region (coming soon)

Bahrain

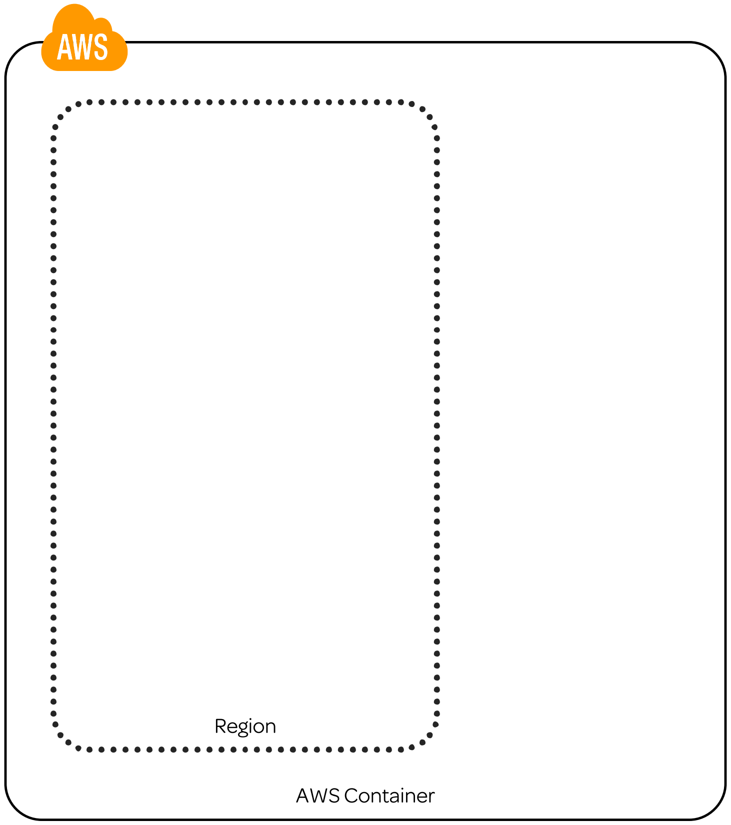
China

France

Hong Kong

Sweden

AWS GovCloud (US-East)



Secure Global Infrastructure and Compliance

BACK

Region | Availability Zones | Endpoints | IAM | Compliance

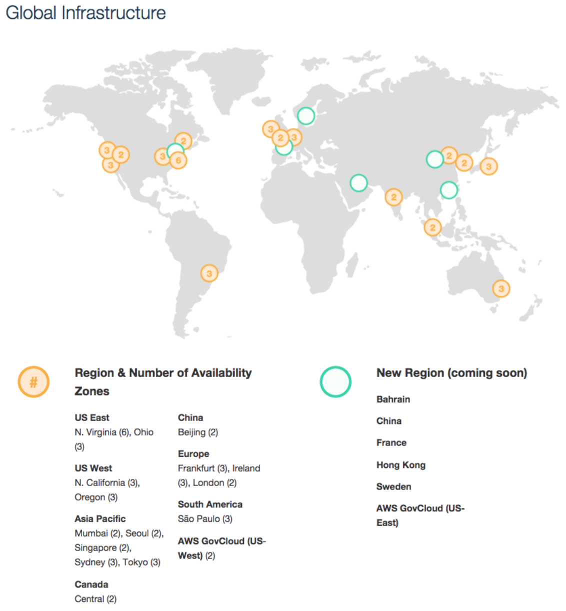
Availability Zones (AZs)

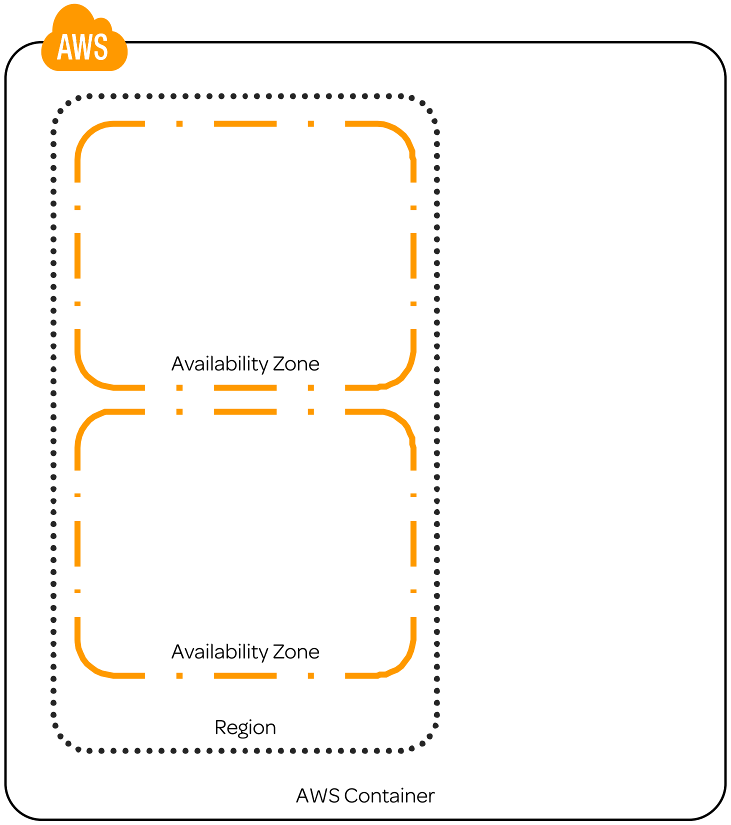
- The independent data centers in an AWS region.
- High-speed links connect them together (LAN-type connectivity)
- There are at least two in every region.

Regarding Security:

- AZs are designed for fault isolation
- Up to the user to configure their systems to take advantage of this fault isolation
    - Surviving a temporary or prolonged failure

Look at the Global Infrastructure chart again below. Take note of the numbers in parentheses. Those numbers are the AZs


Global Infrastructure
#
Region & Number of Availability Zones
US East
N. Virginia (6), Ohio (3)
US West
N. California (3), Oregon (3)
Asia Pacific
Mumbai (2), Seoul (2), Singapore (2), Sydney (3), Tokyo (3)
Canada
Central (2)
China
Beijing (2)
Europe
Frankfurt (3), Ireland (3), London (2)
South America
São Paulo (3)
AWS GovCloud (US-West) (2)
New Region (coming soon)
Bahrain
China
France
Hong Kong
Sweden
AWS GovCloud (US-East)



Secure Global Infrastructure and Compliance

BACK

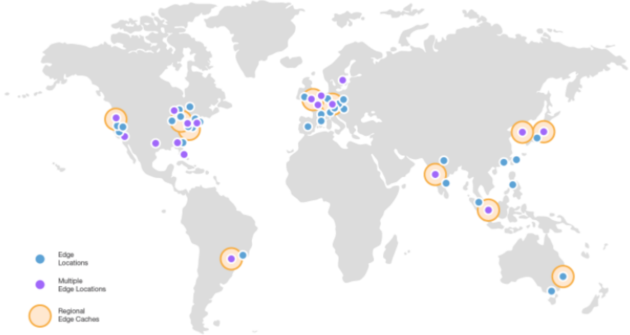
Region | Availability Zones | Endpoints | IAM | Compliance

Endpoints

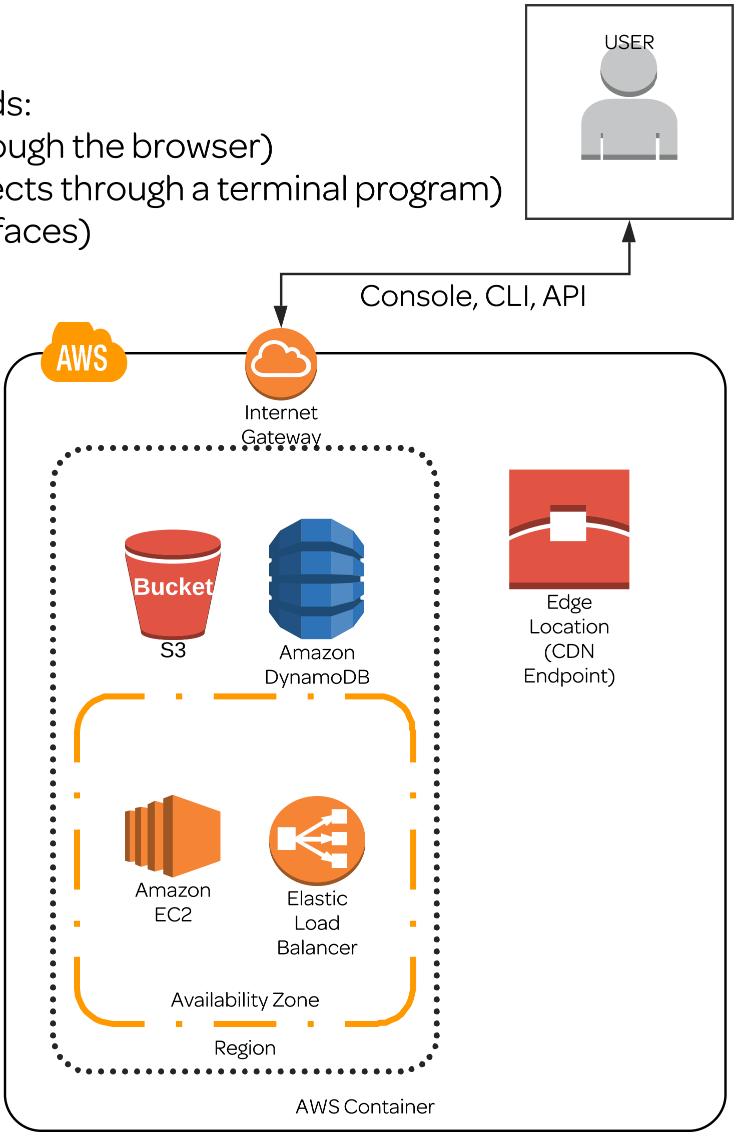
Click Here for **VPC Endpoints**

- AWS provides several different ways to connect to its services
    - Called endpoints
- For access from outside a region:
    - Users can connect using three different methods:
        - AWS Console (Web console launched through the browser)
        - AWS CLI (Command Line Interface, connects through a terminal program)
        - AWS APIs (Application Programming Interfaces)
    - Content Delivery Network (CDN) Endpoints
        -CloudFront Edge Locations

Click the image below to enlarge the edge locations diagram



- Examples of endpoints that exist inside a region and use web URLs:
    - S3
    - DynamoDB
-Examples of services inside Availability Zones that use endpoints
    - EC2 (public subnet)
    - Elastic Load Balancer



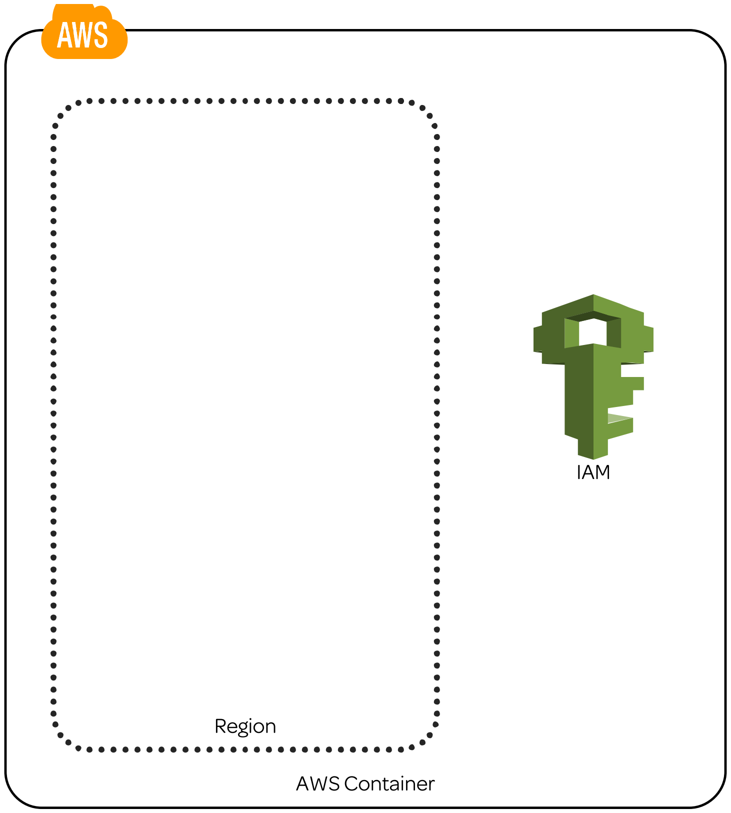
BACK

Secure Global Infrastructure and Compliance

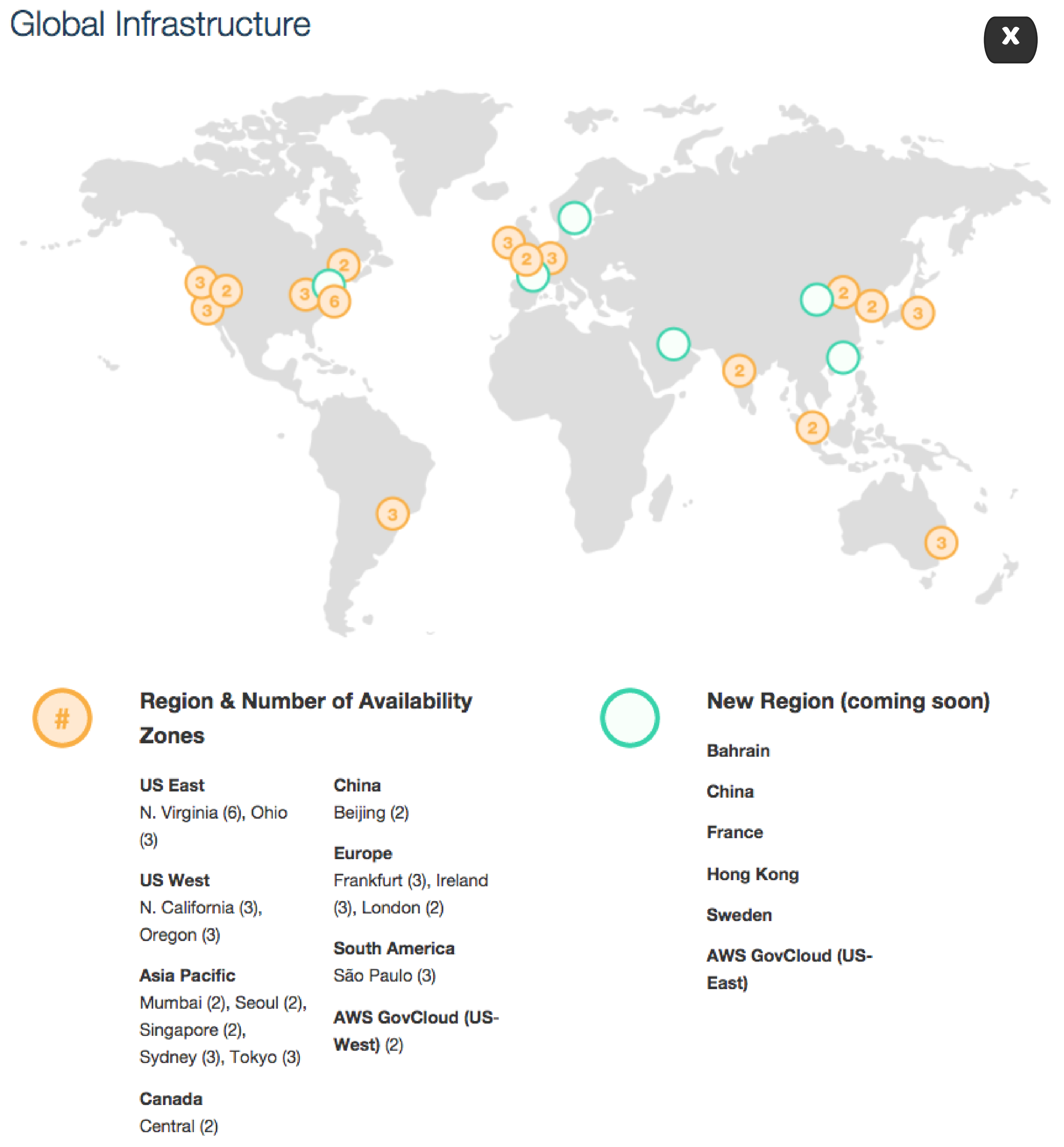
Region | Availability Zones | Endpoints | IAM | Compliance

Identity and Access Management (IAM)

- Global scope across all of AWS (all regions)
    - Allows for large-scale granularity
    - For example, a user can be granted EC2 access in all regions, one region, or multiple regions
- Allows for central management of:
    - Users
    - Passwords
    - Access Keys
    - Permissions
    - Groups
    - Roles
- IAM is one of the main topics in AWS security and will be discussed in greater detail in a later section.



AWS Security Essentials



Global Infrastructure

Region & Number of Availability Zones

US East
N. Virginia (6), Ohio (3)

US West
N. California (3), Oregon (3)

Asia Pacific
Mumbai (2), Seoul (2), Singapore (2), Sydney (3), Tokyo (3)

Canada
Central (2)

China
Beijing (2)

Europe
Frankfurt (3), Ireland (3), London (2)

South America
São Paulo (3)

AWS GovCloud (US-West) (2)

New Region (coming soon)

Bahrain

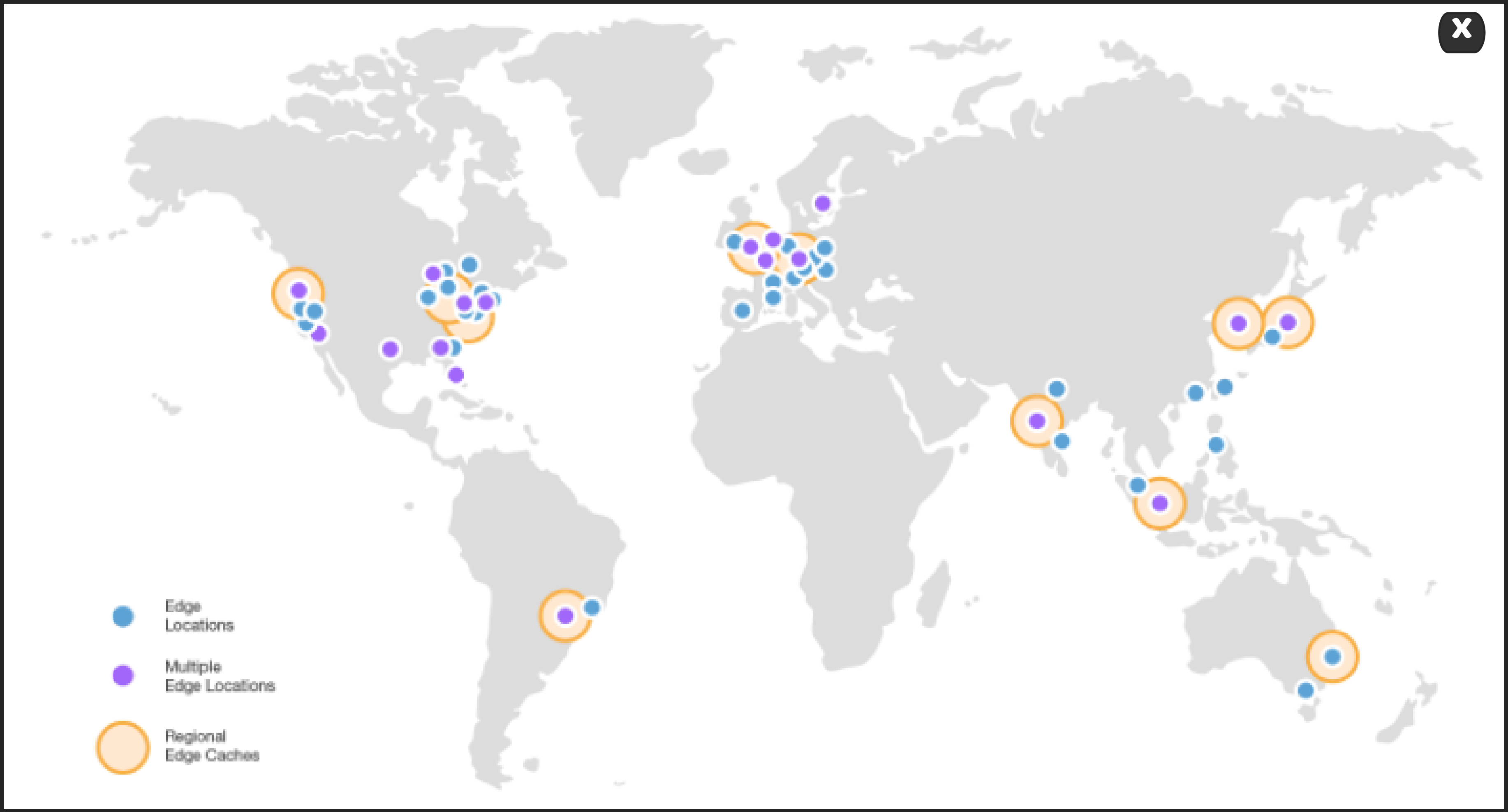
China

France

Hong Kong

Sweden

AWS GovCloud (US-East)


X
Edge Locations
Multiple Edge Locations
Regional Edge Caches

VPC Endpoints

X

- Allows for a private connection to AWS services without going through the internet
- Traffic does not leave the VPC network
- VPC Endpoints are virtual devices and have scalable, redundant and highly available

2 types of VPC Endpoint

Interface (using AWS PrivateLink):

- An Elastic Network Interface (ENI) with a private address serves as the endpoint
- Supported Services
    - Kinesis streams
    - Elastic Load Balancing
    - EC2 API
    - EC2 Systems Manager (centralized management of instances)
    - Service Catalog (central management of IT services in your organization)

Gateway:

- A target for a route table in your environment
- Supported Services
    - DynamoDB
    - S3

Limitations

- Same region only
- IPv4 only
- An **interface** endpoint cannot be accessed through a VPN or VPC Peering connection
    - Only Direct Connect

BACK

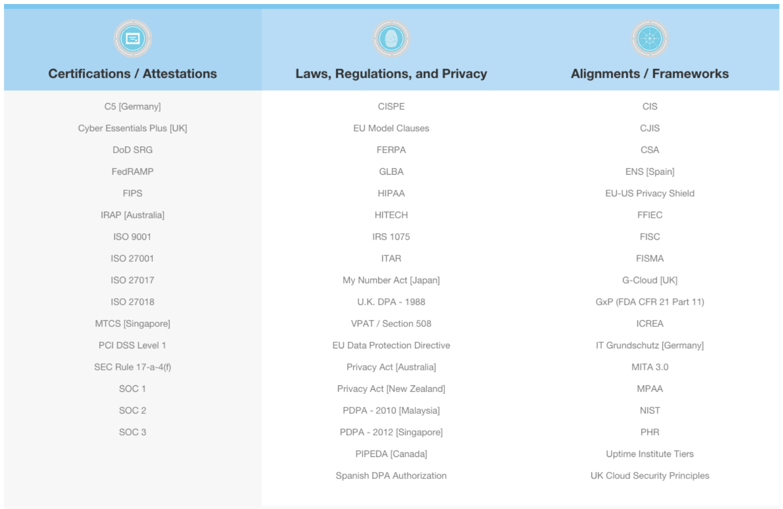
Secure Global Infrastructure and Compliance

Region | Availability Zones | Endpoints | IAM | Compliance

AWS Cloud Compliance

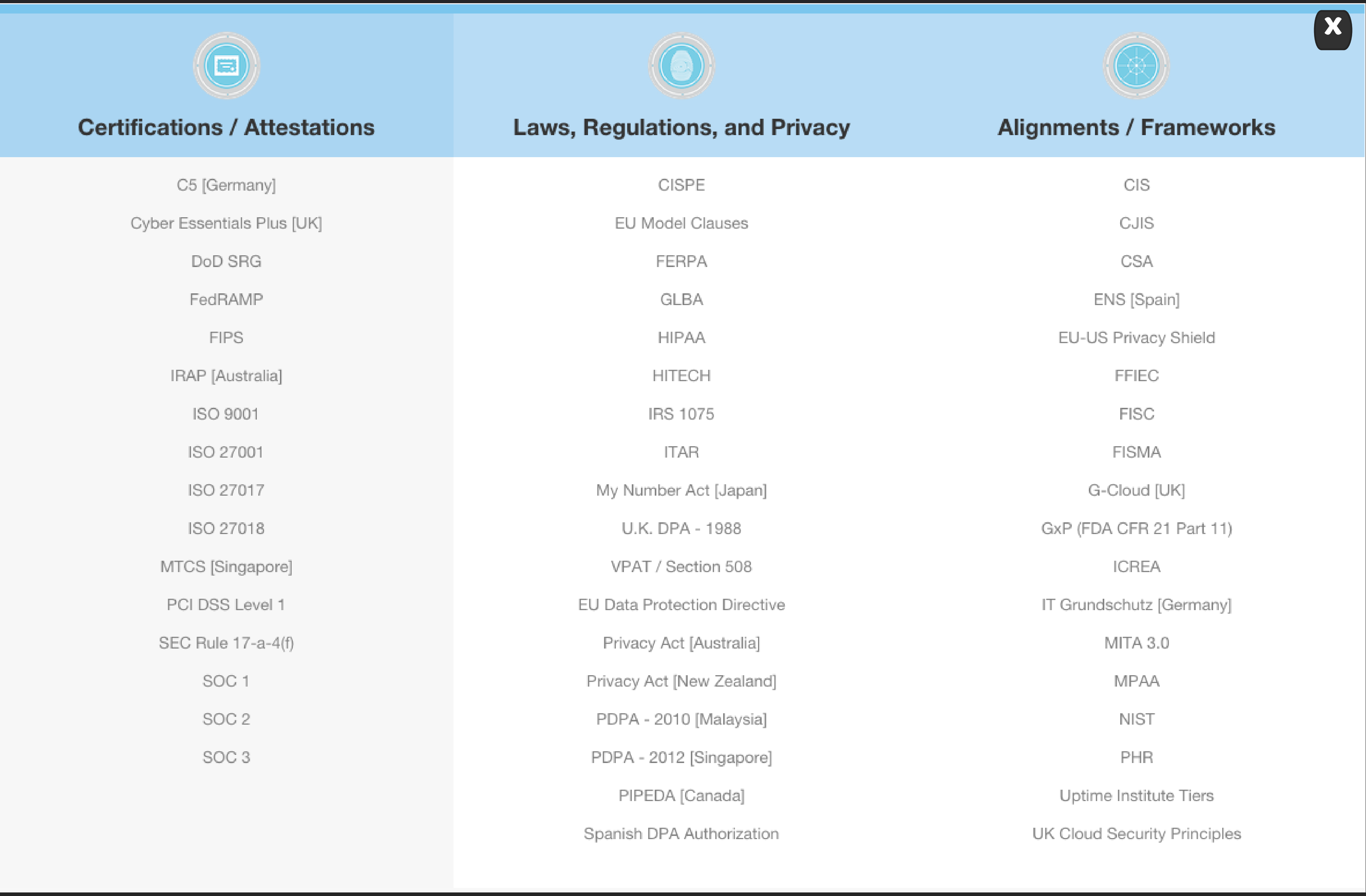
- Programs for finance, healthcare, government, and more
- Documents that AWS meets regulatory, audit, and security standards
    - HIPAA
    - ISO Standards
    - Various regulatory and security agencies around the world
- Only applies to the services and infrastucture that AWS is responsible for
- Does not mean that the applications and data that you deploy in your AWS environment are compliant
    - More on this in the next section
- Documents are available on the AWS website

The image below shows a list of assurance programs AWS is compliant with around the globe. Click the image to enlarge.



Certifications / Attestations	Laws, Regulations, and Privacy	Alignments / Frameworks
C5 [Germany]	CISPE	CIS
Cyber Essentials Plus [UK]	EU Model Clauses	CJIS
DoD SRG	FERPA	CSA
FedRAMP	GLBA	ENS [Spain]
FIPS	HIPAA	EU-US Privacy Shield
IRAP [Australia]	HITECH	FFIEC
ISO 9001	IRS 1075	FISC
ISO 27001	ITAR	FISMA
ISO 27017	My Number Act [Japan]	G-Cloud [UK]
ISO 27018	U.K. DPA - 1988	GxP (FDA CFR 21 Part 11)
MTCS [Singapore]	VPAT / Section 508	ICREA
PCI DSS Level 1	EU Data Protection Directive	IT Grundschutz [Germany]
SEC Rule 17-a-4(f)	Privacy Act [Australia]	MITA 3.0
SOC 1	Privacy Act [New Zealand]	MPAA
SOC 2	PDPA - 2010 [Malaysia]	NIST
SOC 3	PDPA - 2012 [Singapore]	PHR
	PIPEDA [Canada]	Uptime Institute Tiers
	Spanish DPA Authorization	UK Cloud Security Principles

AWS Security Essentials



Certifications / Attestations

C5 [Germany]

Cyber Essentials Plus [UK]

DoD SRG

FedRAMP

FIPS

IRAP [Australia]

ISO 9001

ISO 27001

ISO 27017

ISO 27018

MTCS [Singapore]

PCI DSS Level 1

SEC Rule 17-a-4(f)

SOC 1

SOC 2

SOC 3

Laws, Regulations, and Privacy

CISPE

EU Model Clauses

FERPA

GLBA

HIPAA

HITECH

IRS 1075

ITAR

My Number Act [Japan]

U.K. DPA - 1988

VPAT / Section 508

EU Data Protection Directive

Privacy Act [Australia]

Privacy Act [New Zealand]

PDPA - 2010 [Malaysia]

PDPA - 2012 [Singapore]

PIPEDA [Canada]

Spanish DPA Authorization

Alignments / Frameworks

CIS

CJIS

CSA

ENS [Spain]

EU-US Privacy Shield

FFIEC

FISC

FISMA

G-Cloud [UK]

GxP (FDA CFR 21 Part 11)

ICREA

IT Grundschutz [Germany]

MITA 3.0

MPAA

NIST

PHR

Uptime Institute Tiers

UK Cloud Security Principles

Shared Responsibility Model and Trusted Advisor

BACK

Infrastructure Services

Container Services

Abstracted Services

Trusted Advisor

The Shared Responsibility Model describes what Amazon Web Services is responsible for and what you, the user or customer, is responsible for when it relates to security.

- It's like a line of demarcation.

Shared Responsibility Model and Trusted Advisor

BACK

Infrastructure Services | Container Services | Abstracted Services | Trusted Advisor

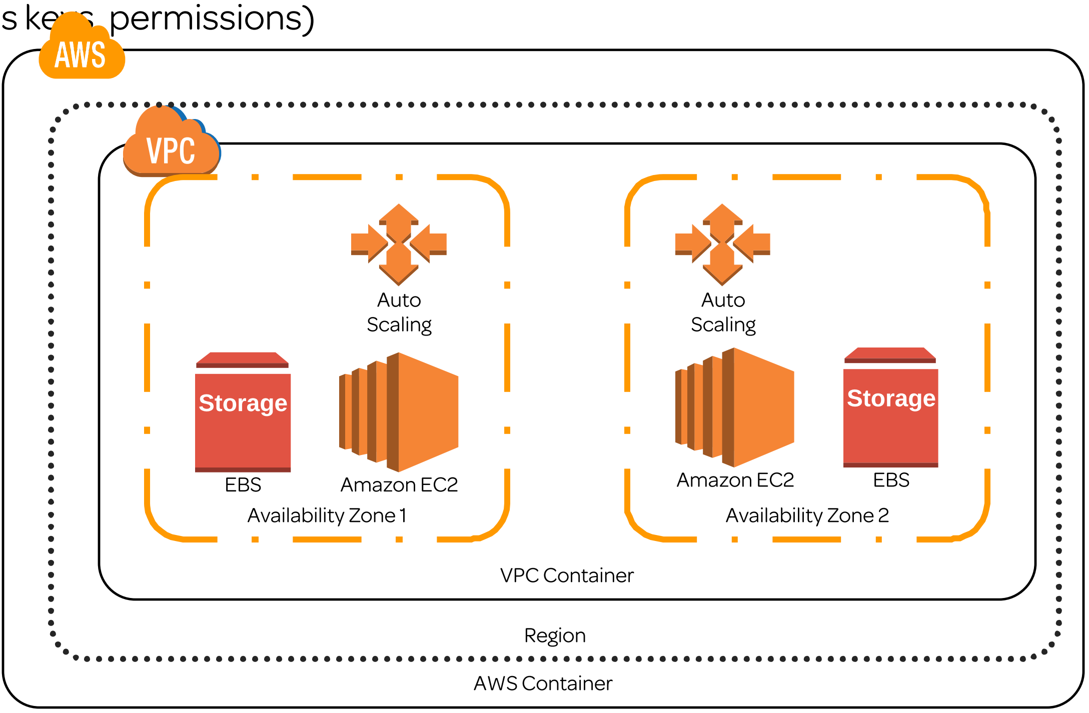
Infrastructure Services

This includes AWS service like VPC, EC2, EBS, and Auto Scaling

Amazon is responsible for "security of the cloud"
- The Global Infrastructure (Regions, AZs, Edge Locations)
- The Foundation Services (Compute, Storage, Database, Networking)

The user/customer is responsible for "security in the cloud"
- Customer Data
- Platforms and Applications
- OS and Network configurations (patching, security groups, network access control)
- Customer IAM (passwords, access keys, permissions)

Additional Concerns:
- Data encryption
- Data integrity



Shared Responsibility Model and Trusted Advisor

BACK

Infrastructure Services | Container Services | Abstracted Services | Trusted Advisor

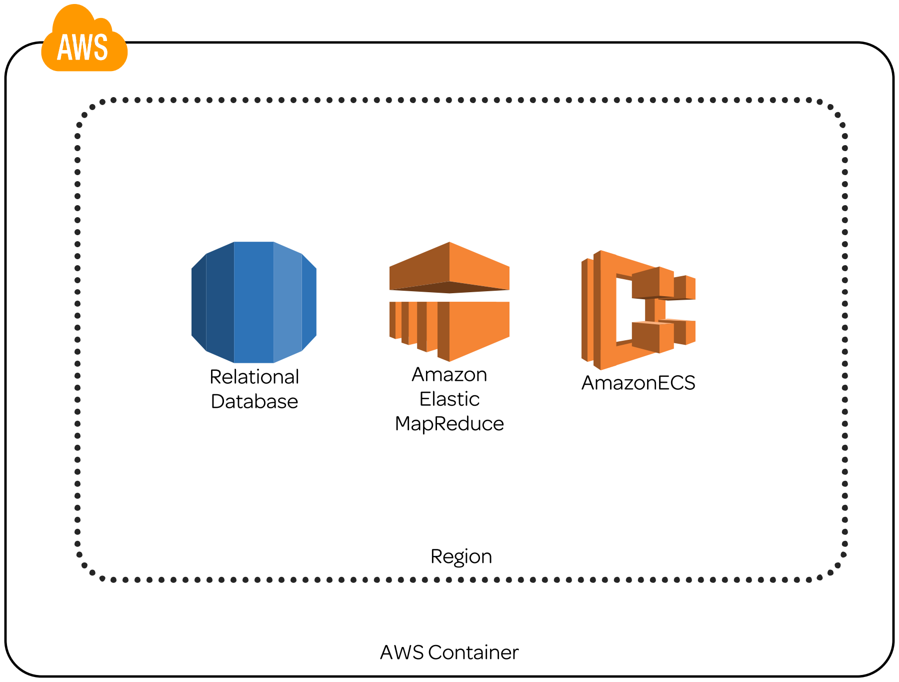
Container Services

- This includes AWS services like RDS, EMR, and ECS
- AWS is responsible for:
  - Platforms and Applications
  - OS and network configurations
  - The Global Infrastructure (Regions, AZs, Edge Locations)
  - The Foundation Services (Compute, Storage, Database, Networking)

The user/customer is responsible for:
- Customer data
- Customer IAM

Additional Concerns

- Data encryption
- Data integrity



Shared Responsibility Model and Trusted Advisor

BACK

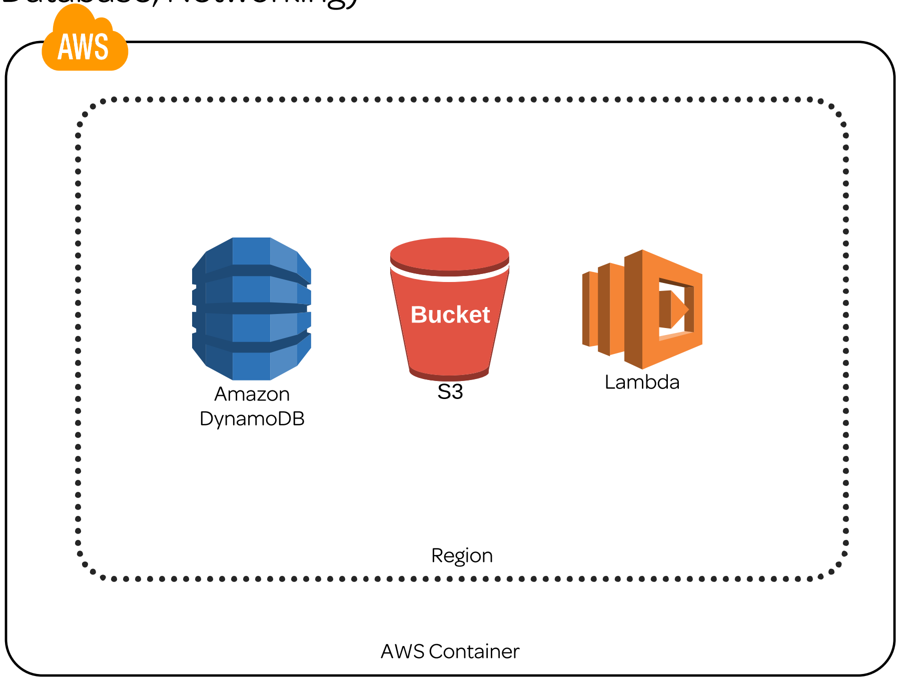
Infrastructure Services | Container Services | Abstracted Services | Trusted Advisor

Abstracted Services

This includes AWS services like DynamoDB, S3, and Lambda

AWS is responsible for:
- Network traffic protection
- Platforms and Applications
- OS and network configurations
- The Global Infrastructure (Regions, AZs, Edge Locations)
- The Foundation Services (Compute, Storage, Database, Networking)

The user/customer is responsible for:
- Customer IAM
- Data in transit and client-side



Additional Concerns

- Data encryption
- Data integrity

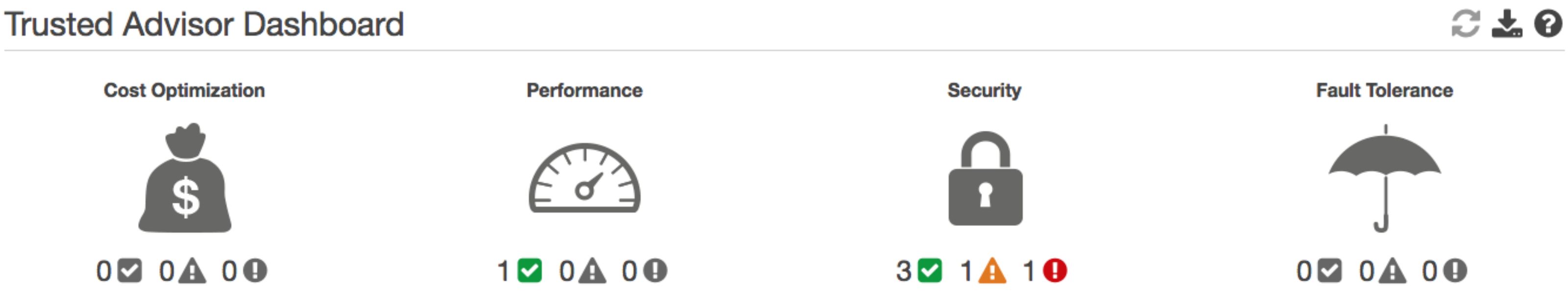
BACK

Shared Responsibility Model and Trusted Advisor

Infrastructure Services | Container Services | Abstracted Services | Trusted Advisor

Trusted Advisor Tool

- Allows an AWS customer to get reports on their environment , including :
    - Cost Optimization
    - Performance
    - Security
    - Fault Tolerance
- Available to all customers:
    - Access to six core checks
        - Security (security groups, IAM, MFA on root account, EBS and RDS public snapshots)
        - Performance (service limits)
- Available to Business and Enterprise support plans:
    - Access to the full set of checks
        - All four catagories above
    - Notifications
        - Weekly updates
    - Programmatic access
        - Retreive results from the AWS Support API



Trusted Advisor Dashboard

Cost Optimization	Performance	Security	Fault Tolerance
0 ✓ 0 ⚠ 0 ❗	1 ✓ 0 ⚠ 0 ❗	3 ✓ 1 ⚠ 1 ❗	0 ✓ 0 ⚠ 0 ❗

BACK

Identity and Access Management (IAM)

Root User

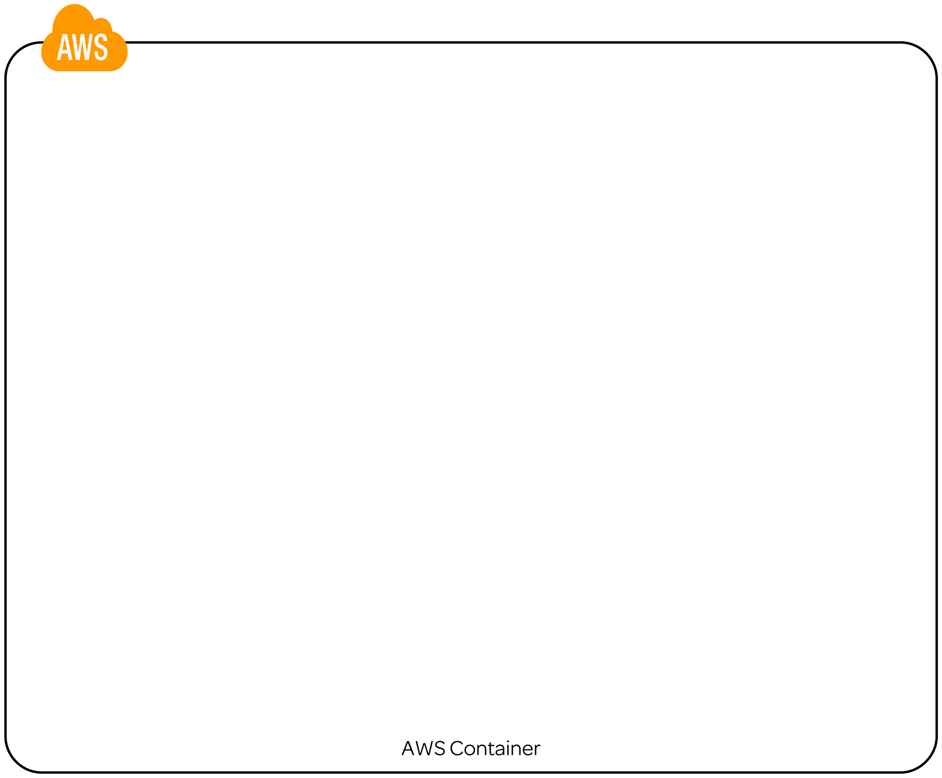
Users

Groups

Roles

Policies

Access Advisor



Identity and Access Management (IAM)

BACK

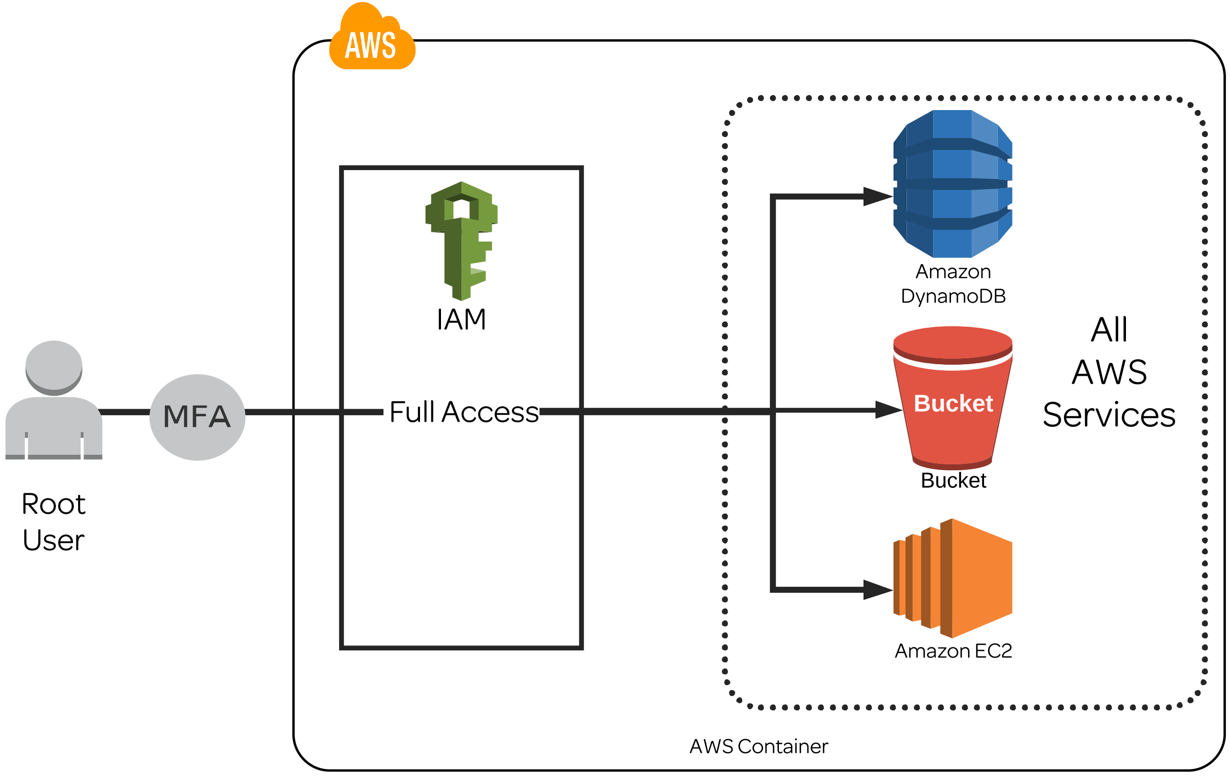
Root User | Users | Groups | Roles | Policies | Access Advisor

Root User

- The user created when an AWS account is created
- The credentials are the email and password used when signing up for an AWS account.
- By default, the root user has *FULL* administrative rights and access to every part of the account

Best practices for root user

- The root user should *not* be used for daily work and administration.
    - Another user should be created for daily work that has admin rights.
- The root user account should not have access keys; delete them if they exist
- The root user should always use *Multi-Factor Authentication (MFA)* like Google Authenticator



AWS Security Essentials

BACK

Identity and Access Management (IAM)

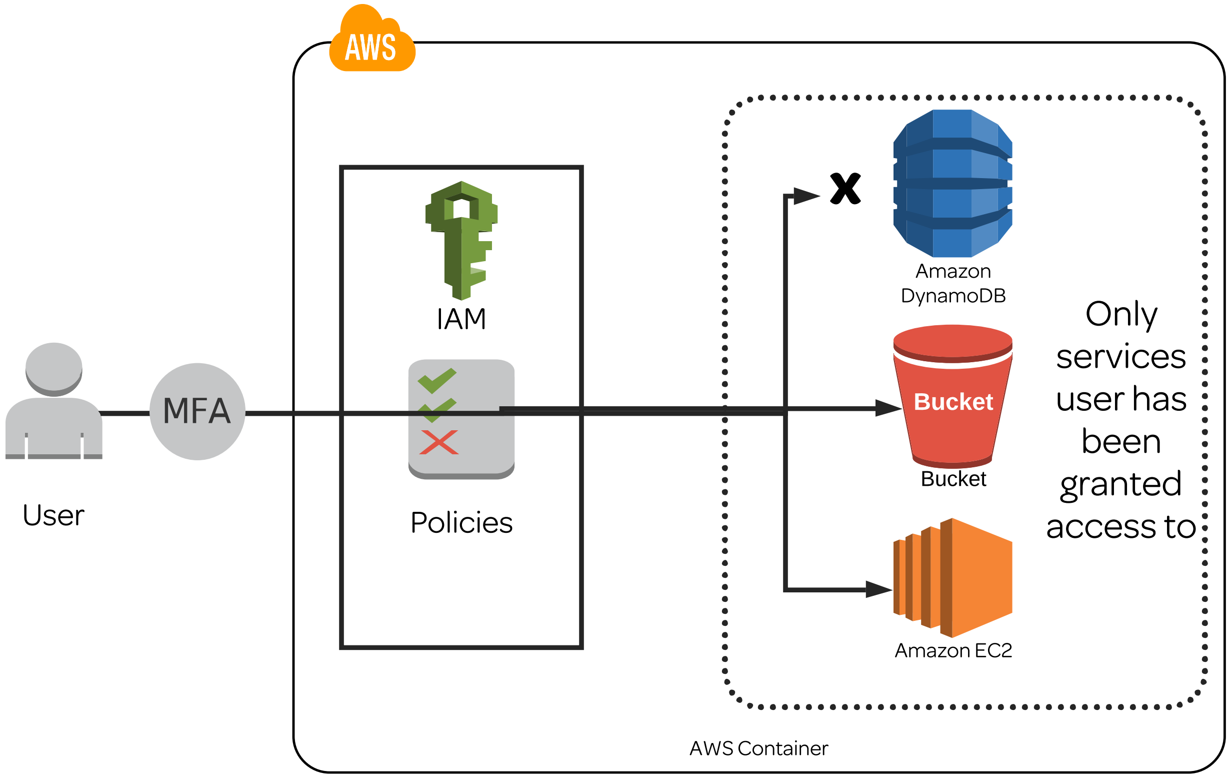
Root User | Users | Groups | Roles | Policies | Access Advisor

IAM Users

- A new user has a *implicit deny* for all AWS services, requiring a policy be added to grant them access.
- Users receive unique credentials (username, password, and possibly access keys)
- Users can have IAM policies appllied directly to them, or they can be a member of a group that has policies attached—more on this later.
- With policies, an explicit deny always overrides an explicit allow from attached policies.
  - For instance, all policies attached to a user will be ignored if a single deny all policy is added.

Best practices for users:

- NEVER store or "pass" your access credentials to an EC2 instance-- use SSH forwarding
- Multi-Factor Authentication (MFA) *can* and *should* be used for user accounts
- Access credentials are unique and should never be shared



Identity and Access Management (IAM)

BACK

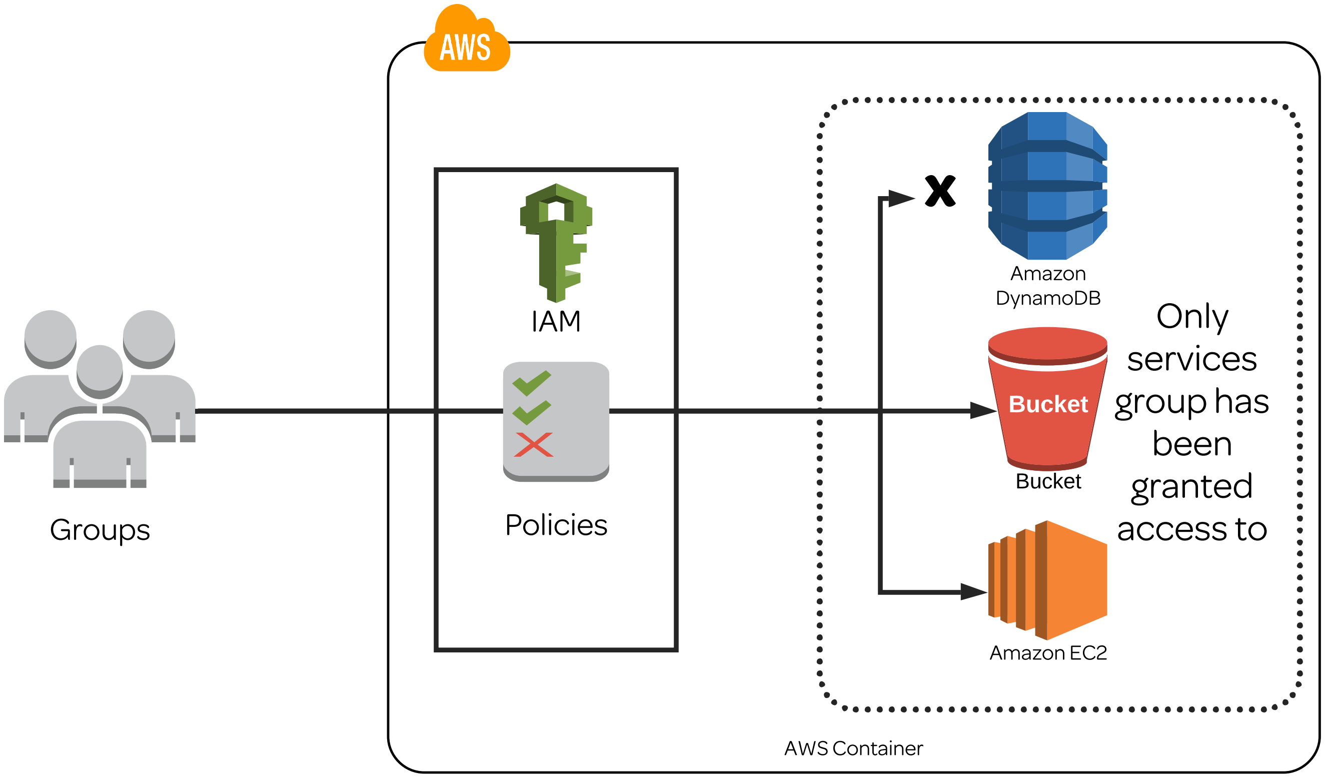
Root User | Users | Groups | Roles | Policies | Access Advisor

IAM Groups

- Allows for policy assignment to multiple users at the same time
- Is a more organized and efficient way to manage users and policies

Best practices for IAM Groups:

- Organize users by function (i.e. DB admins, developers, architects, etc.)
- Assign policies to the group, not the individual users



Only services group has been granted access to

Identity and Access Management (IAM)

BACK

Root User | Users | Groups | Roles | Policies | Access Advisor

IAM Roles

- Temporary security credentials in AWS managed by Secure Token Service (STS)
- Another entity can "assume" the specific permissions defined by the role.
- These entities include:
  - AWS resources (such as an EC2 instance)
  - A user outside of our AWS account who needs temporary access.

Click here to learn more about **Secure Token Service (STS)**

Roles with AWS Services

- Roles must be used because policies cannot be directly attached to AWS services.
- Services can only have ONE role attached at a time.
- You should never pass or store credentials in or to an EC2 instance—so roles are used instead.
- Example: An EC2 instance needs to be able to read data from an S3 bucket.
  - The instance assumes a role with S3 read-only permissions from IAM
  - The instance can then read objects from the bucket specified in the role.
- You are now able to change roles on running EC2 instances through the Console and CLI.

Other uses of roles:

- *Cross Account Access (Delegation)*
  - Provide access to another AWS user from another account
- *Identity Federation*
  - Users outside AWS can assume a “role” for temporary access to AWS accounts and resources.
  - These users assume an "Identity Provider Access" role
  - Example identity providers:
    - Active Directory
    - Single sign-on providers like Facebook, Google, Amazon, etc.

Click here to learn more about **Identity Federation**

Identity and Access Management (IAM)

BACK

Root User | Users | Groups | Roles | Policies | Access Advisor

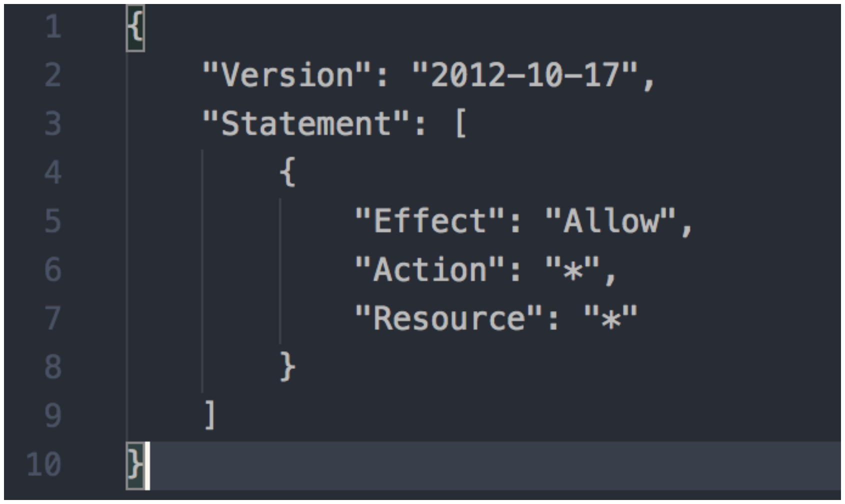
IAM Policies

- A document that states one or more permissions (JSON formatted).
- An explicit deny always overrides and an explicit allow.
- This allows for the use of a "deny all" policy to quickly restrict ALL access that a user may have.

IAM provides pre-built policy templates to assign to users and groups, examples include:
- Administrator access: Full access to ALL AWS resources.
- Power user access: Admin Access except it does not allow user/group management.
- Read only access: Only view AWS resources (i.e. user can only view what is in an S3 bucket).

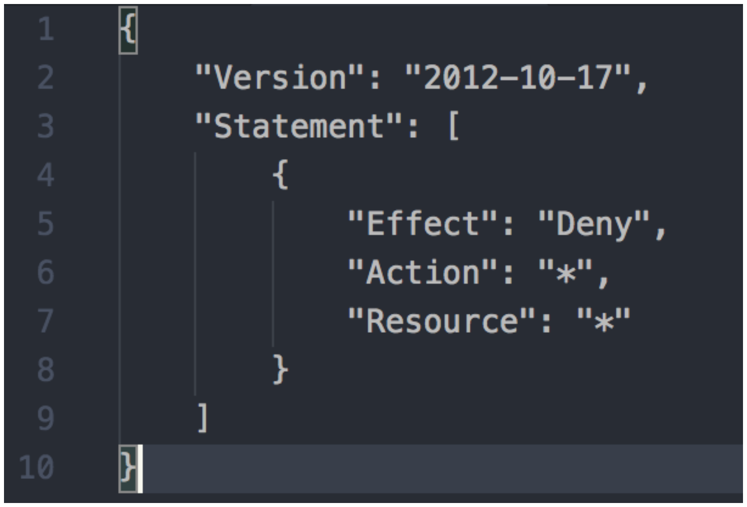
- You can also create custom IAM permission policies using the policy generator or written from scratch.
- More than one policy can be attached to a user or group at the same time.
- Policies cannot be directly attached to AWS resources (such as an EC2 instance).

A policy for
Adminstrator access



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

An explicit deny policy that
removes user permissions



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Identity and Access Management (IAM)

BACK

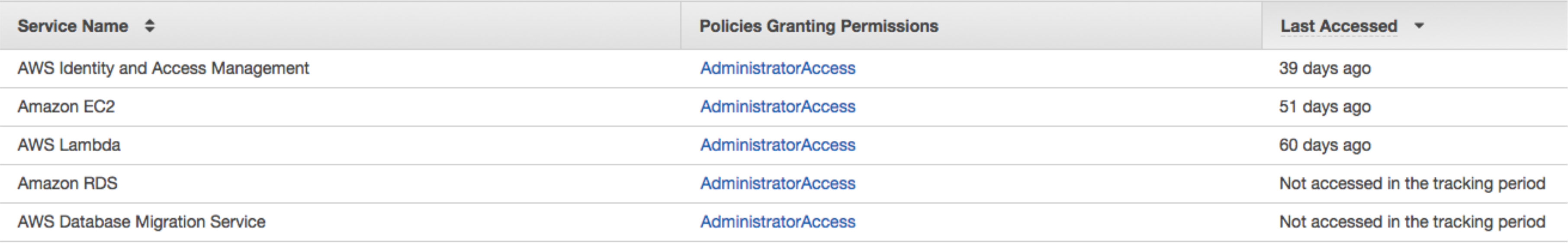
Root User | Users | Groups | Roles | Policies | Access Advisor

IAM Access Advisor

- Users should have as few permissions as possible (least privilege).
- *Important*:
    - These permissions include group memberships and assumed roles to other accounts.
- Access Advisor allows unused permissions to be identified
    - A way of auditing permissions

Can look at permissions per:

- User
- Group
- Role



Service Name	Policies Granting Permissions	Last Accessed
AWS Identity and Access Management	AdministratorAccess	39 days ago
Amazon EC2	AdministratorAccess	51 days ago
AWS Lambda	AdministratorAccess	60 days ago
Amazon RDS	AdministratorAccess	Not accessed in the tracking period
AWS Database Migration Service	AdministratorAccess	Not accessed in the tracking period

Identity Federation

Federation- Providing a non-AWS user temporary AWS access by linking that user's identity across multple identity systems .

Federation with third party providers:

- Most commonly used in web and mobile applications
- AWS *Cognito* allows for:
    - Creation of unique identities for users
    - Use identity providers to federate them
- Example providers
    - Facebook, Google, Amazon, etc.

Establishing Single Sign On (SSO) using SAML 2.0:

- Most commonly used in enterprise environments with an existing directory system
    - Active Directory, etc.
- Federated users can access AWS resources using their corporate domain accounts
- Federation also aids user management by allowing central management of accounts

Establishing Single Sign On (SSO) without using SAML:

- AWS *Directory Service* for Microsoft Active Directory
    - allows for a Windows trust relationship to be built between an on-premises Microsoft AD and your AWS Microsoft AD in the cloud.

Secure Token Service (STS)

X

- The service in AWS that allows for management of temporary security credentials

- It allows for granular control of how long the access remains active:
  - fifteen minutes to one hour (Default = 1 hour)

- Credentials are not stored with the user or service granted temporary access
  - A token is attached to the access request

- Beneficial in a number of ways:
  - Low risk of credentails being exposed (not distributed)
  - Do not have to create IAM identities for every user
  - Because they are temporary in nature, there is no need to rotate keys

STS uses a single endpoint

https://sts.amazonaws.com

- This single endpoint resides in us-east-1 (N. Virginia)
- Latency can be reduced by using STS API calls to regions that support them
- Temporary credentials have global scope, just like IAM

AWS Security Essentials

Encryption Essentials

BACK

Overview

Symmetric

Asymmetric

HSM

KMS

BACK

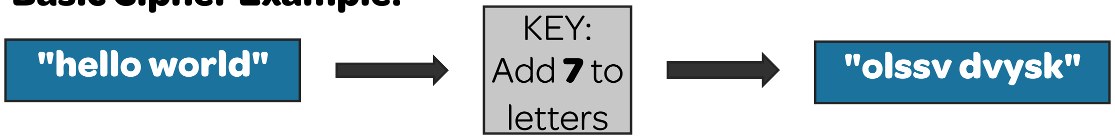
Encryption Essentials

Overview | Symmetric | Asymmetric | HSM | KMS

Encryption Overview

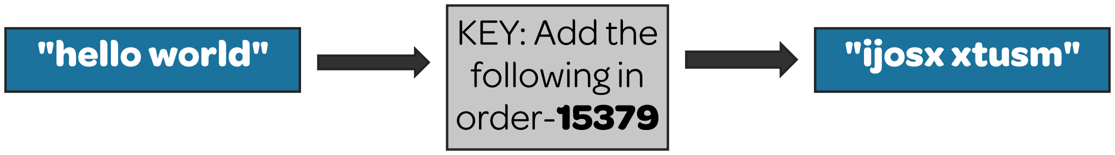
Key - The "cipher" used to encrypt and decrypt data
- Keys have gotten longer as computing power has grown.
- Keys can also be used to encrypt other keys (Master Key)
  - This process is known as enveloping

Basic Cipher Example:



"hello world" → KEY: Add **7** to letters → **"olssv dvysk"**

A More Advanced Cipher Example Using Block CIpher:



"hello world" → KEY: Add the following in order-**15379** → **"ijosx xtusm"**

Server-Side Encryption
- Data is encrypted as it is written to disk, then decrypted as it is read from the disk.
-Often referred to as encryption "at rest"

Client-Side Encryption
- Data is encrypted by the client before it is sent to the server, then decrypted when the client receives data from the server
- Often referred to as encryption "in transit"

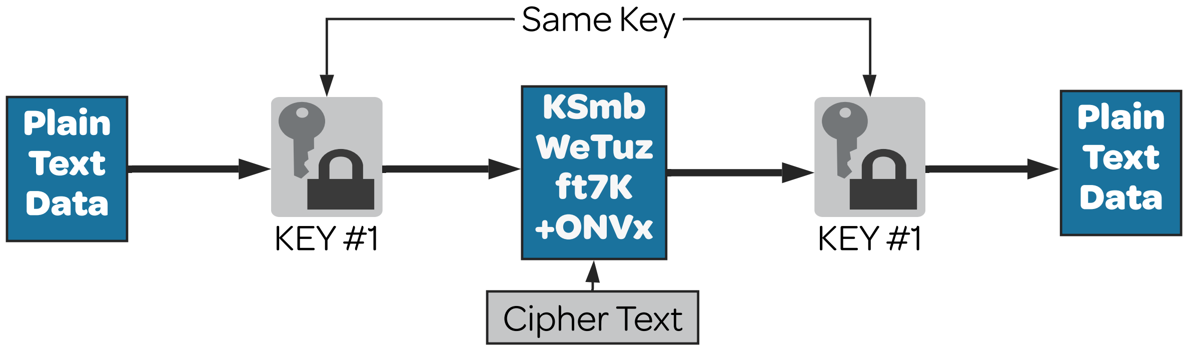
Encryption Essentials

BACK

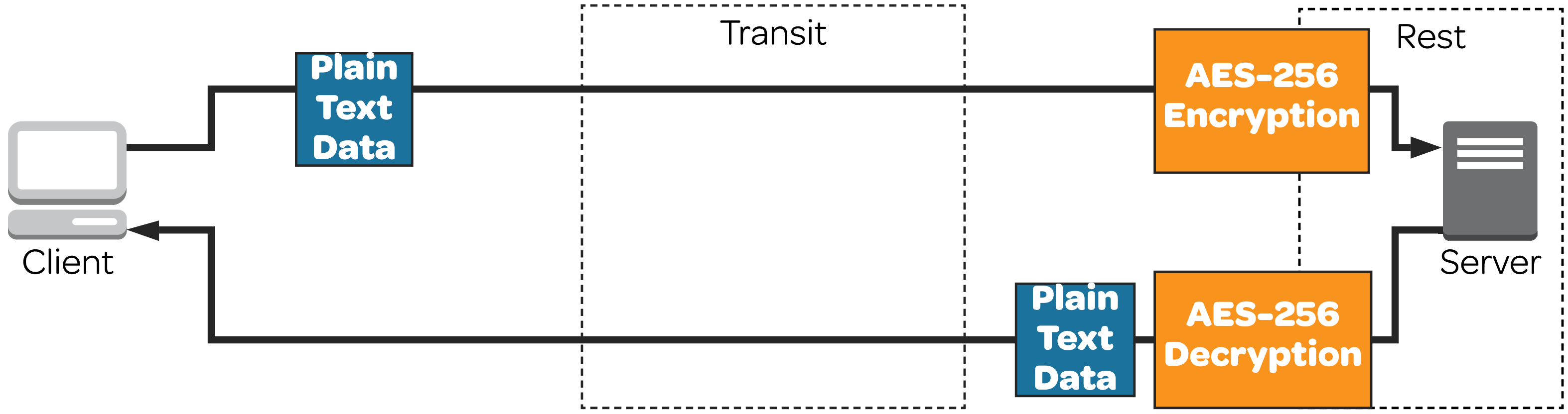
Overview | Symmetric | Asymmetric | HSM | KMS

Symmetric Encryption

- Uses the same key to encrypt and decrypt
- Example: Advanced Encryption Standard (AES) - 128, 192, and 256 bit



AES-256 Example:



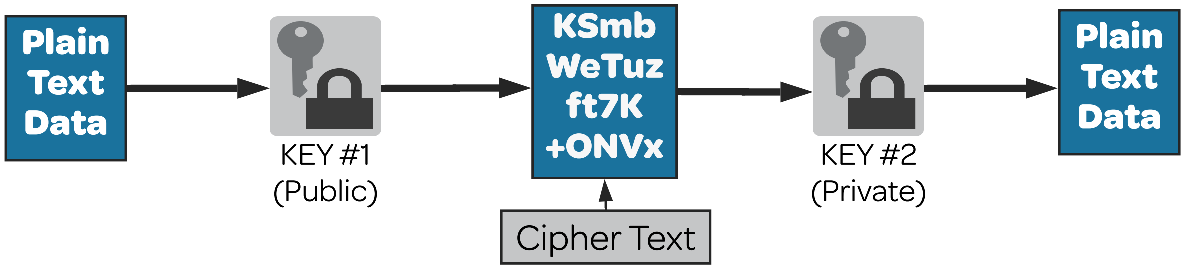
Encryption Essentials

BACK

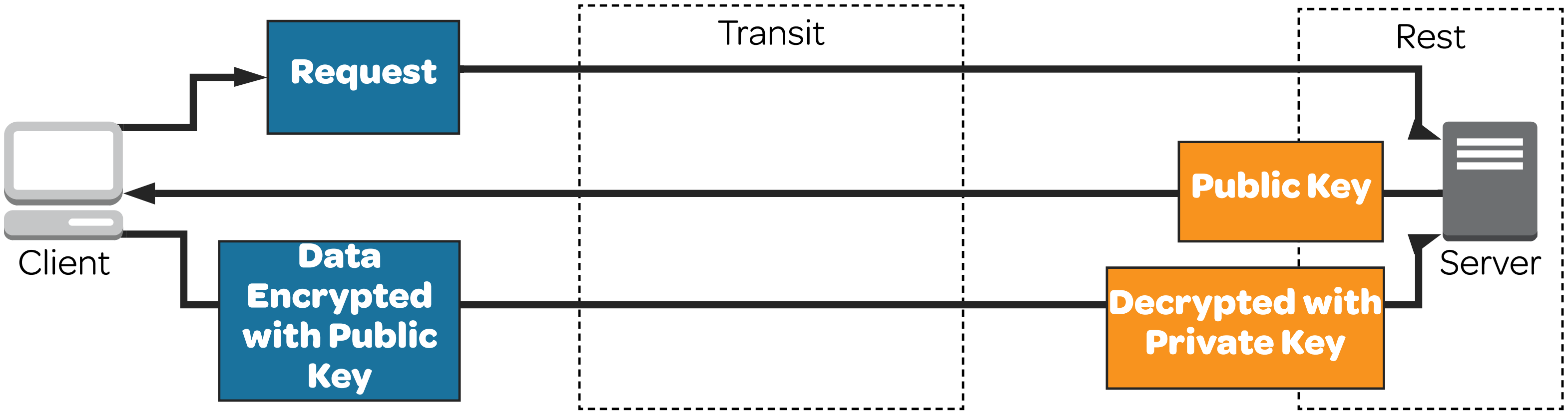
Overview | Symmetric | Asymmetric | HSM | KMS

Asymmetric Encryption

- Uses different keys to encrypt and decrypt, a public and a private key
- The private key cannot be derived from the public key
- The public key is available to any entity
- Example: Secure Sockets Layer (SSL), Transport Layer Security (TLS), SSH



SSL/TLS Example:



BACK

Encryption Essentials

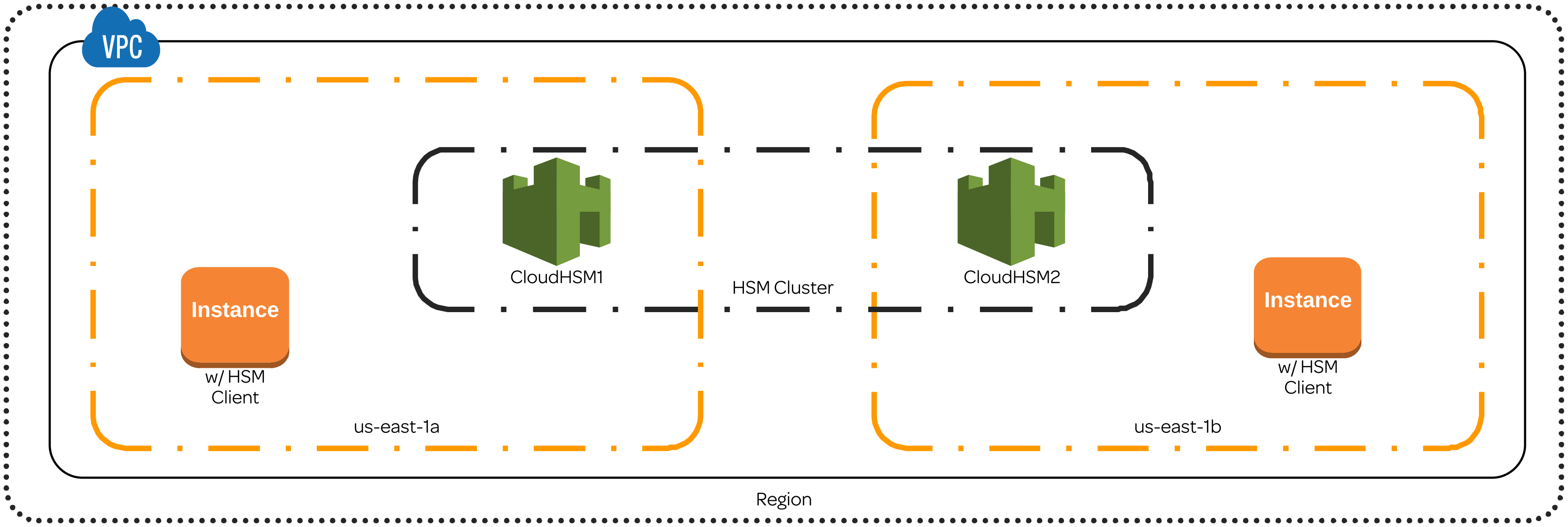
Overview | Symmetric | Asymmetric | HSM | KMS

Hardware Security Module (HSM)

- Physical device used for secure key storage and management
- On-Premises

AWS CloudHSM

- Dedicated device, managed by AWS (they do not have access to keys)
- In your VPC and separated from other networks for latency and security reasons
- You control and manage your keys
- Can be placed in multiple AZs and clustered
    - Load balances and replicates keys
- Perfect solution if your organization requires keys be kept on dedicated hardware
- Use case: Asymmetric "handshakes" can increase processing but can be offloaded to CloudHSM



Encryption Essentials

BACK

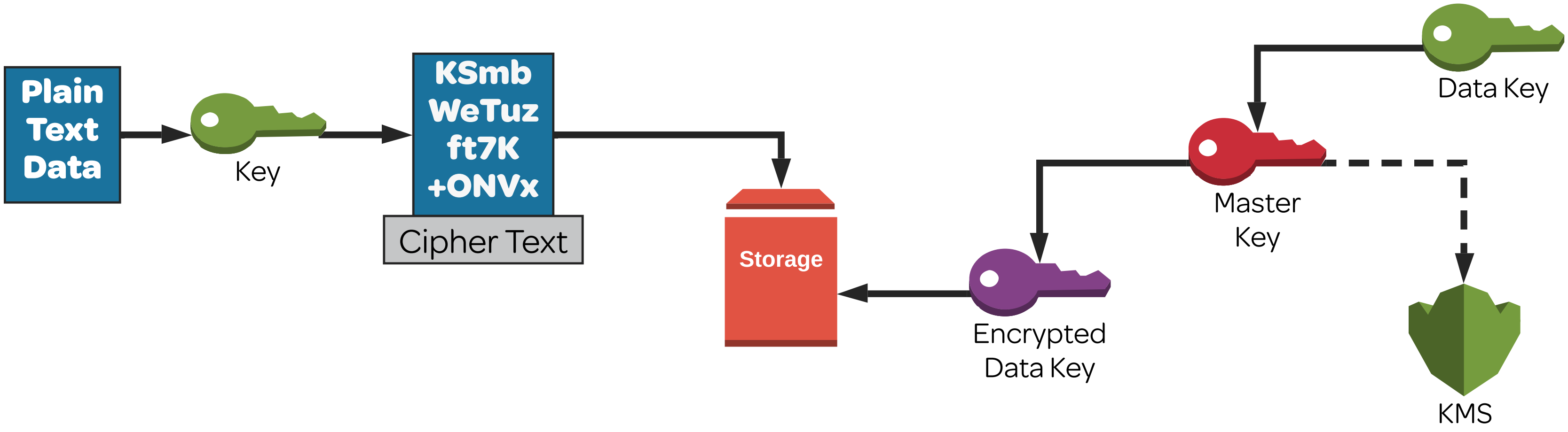
Overview | Symmetric | Asymmetric | HSM | KMS

Key Management Service (KMS)

- Managed service that allows you to create and control your encryption keys
- Advantages over HSM are:
    - Can use IAM policies for KMS access
    - AWS services integrate directly with KMS

Important Concepts

- KMS stores **Customer Master Keys (CMK)**
- The process follows symmetric encryption but has a major twist
- In some HSMs, there can be any number of **key encryption keys (KEK)**
    - Process known as **enveloping**
- KMS only envelopes one layer and stores the "top" key, the CMK
- The encrypted data key is stored with the data



BACK

OS-Level Security

Overview

SSH

Bastion Host

Linux Example

Windows Example

OS-Level Security

BACK

Overview | SSH | Bastion Host | Linux Example | Windows Example

EC2 OS-Level Security

Shared responsibility model

- EC2 falls under the infrastructure model
    - Customer is responsible for application, platform, data, networking, and IAM
    - Everything that is not directly a part of the global infrastructure or foundational services.
- In addition to OS and the application that is deployed, AWS customer must think about:
    - IAM policies
    - Encryption strategies
    - Security groups and NACLs

How to access our instances securely:

Linux/macOS accessing Linux instances (cloud-init):

- Use the built-in terminal to SSH to the instance
- Authenticate with EC2 key pairs

For any OS connecting to Windows (ec2config):

- Use the EC2 key pair to decrypt the Administrator password
- Use Remote Desktop

For Windows connecting to Linux instances:

- Use PuTTY
- *Link to a guide in the course downloads*

OS-Level Security

BACK

Overview | SSH | Bastion Host | Linux Example | Windows Example

SSH Walkthrough

- SSH actually uses both types of encryption, symmetric and asymmetyric, depending on the purpose.

Connection (symmetric)

- Initial connection gets encrypted with both sides using an agreed upon session key.
- This process is much faster for data transfer

EC2 key pair authentication (asymmetric)

- Keys are generated (public and private) - industry-standard RSA key pairs
- Public key gets copied to the server (~/.ssh/ authorized_keys) using *cloud-init*
    - This happens automatically when you launch an instance with an associated key pair
- Private key is downloaded to the user's computer and permissions need to be updated
    - chmod 400 <keyname>.pem
    - Very important this key is protected, do not copy it to an instance.
- Server sends a challenge message to the client encrypted with the public key , and it gets decrypted using the client's private key
- That string is sent back to the server, and if the string matches, access is granted.
- Windows servers using the ec2config service, use the private key to decrypt the administrator password

Click here to see the **Connection** graphic

Click here to see the **Key Pair Authentication** graphic

OS-Level Security

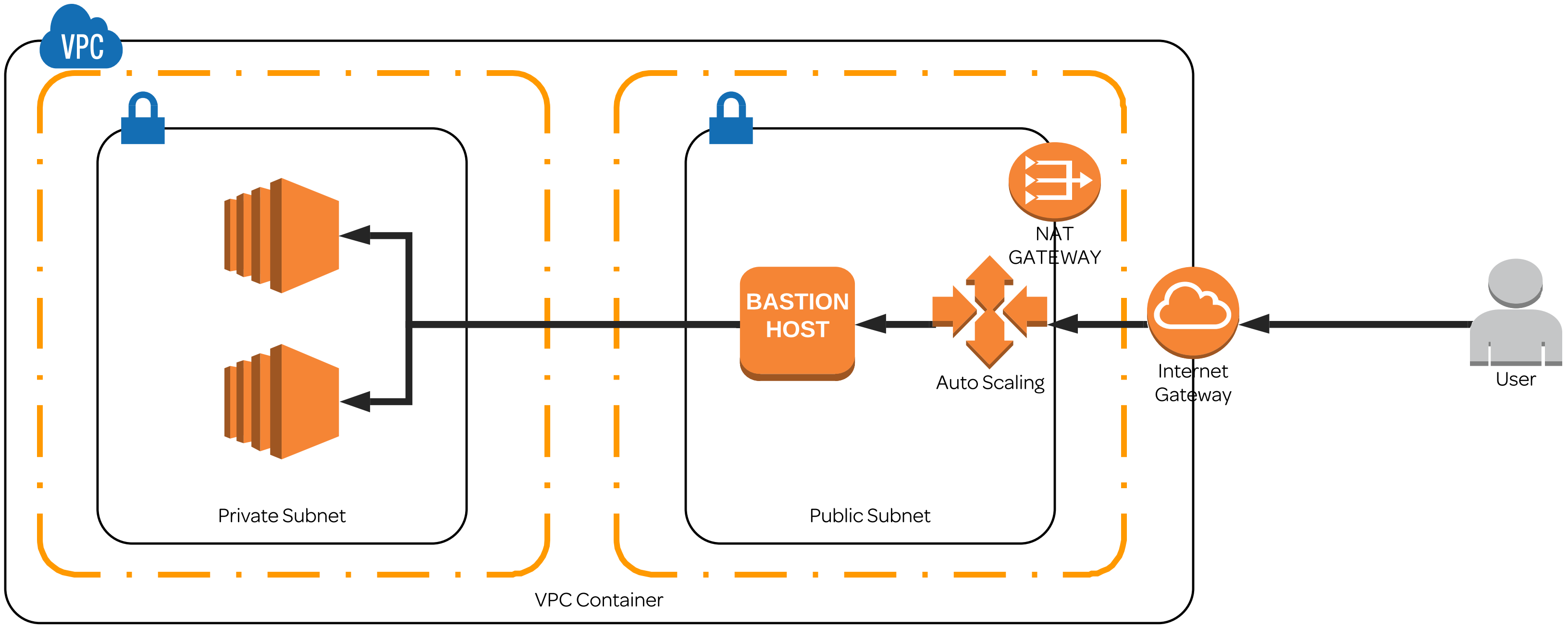
BACK

Overview | SSH | Bastion Host | Linux Example | Windows Example

Bastion Host

- Functions as a "jumpbox"
- Allows us to securely access instances in private subnets without making those instances public in any way

Best Practices
- Deploy in two availability zones
- Use Auto Scaling to ensure the number of bastion hosts
- Deploy in public subnets (DMZ)
- Access is locked down and only allowed from known CIDR ranges
- Ports are limited to only ports the bastion host needs
- **Do Not** copy keys or access information to the bastion host or any other instance.



BACK

OS-Level Security

Overview | SSH | Bastion Host | Linux Example | Windows Example

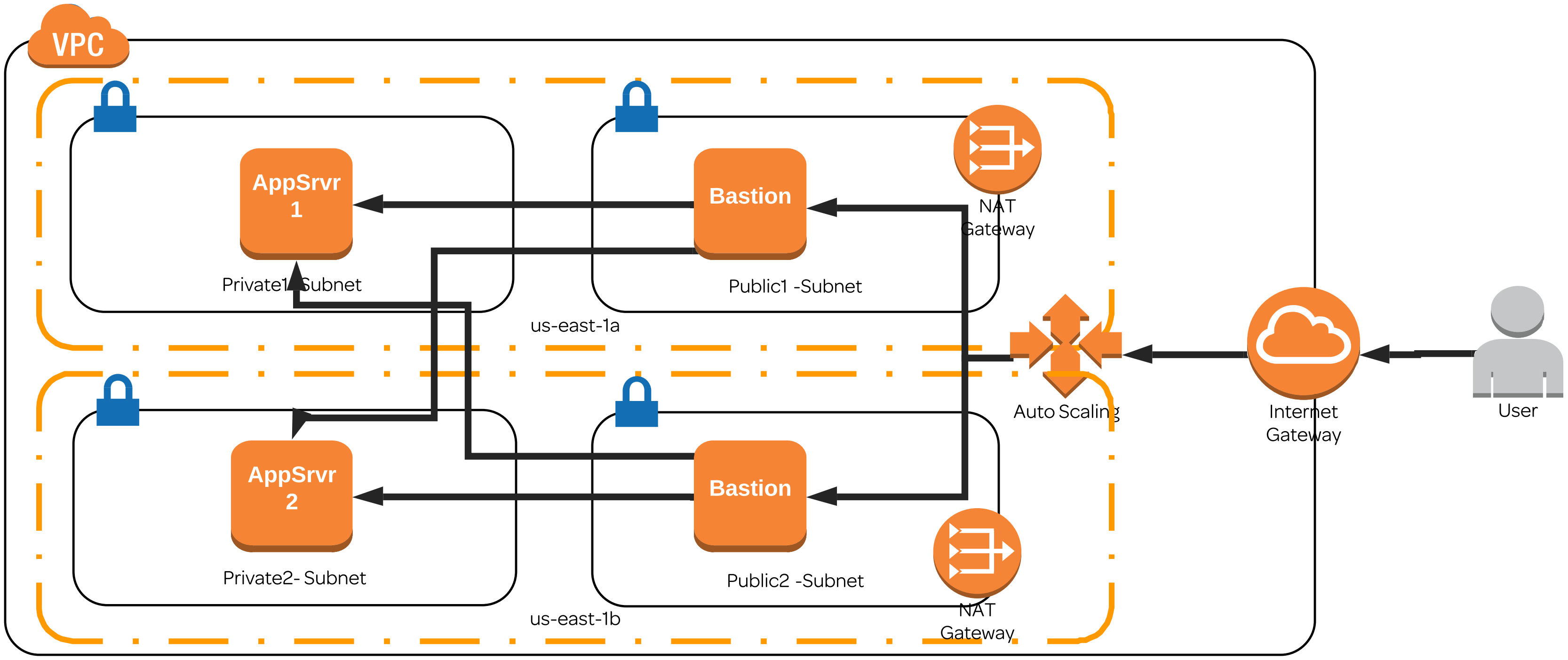
Linux/macOS Example

- Use SSH forwarding whenever possible.
- Do not copy access keys to EC2 instances, per best practices

Commands:

```
    chmod 400 <path-to-key>.pem
    ssh-agent bash
    ssh-add <path-to-key>.pem
    ssh -A ec2-user@<ipaddress>
    ssh ec2-user@<ipaddress> (second host)
```

Infrastructure of the walkthrough example:



OS-Level Security

BACK

Overview | SSH | Bastion Host | Linux Example | Windows Example

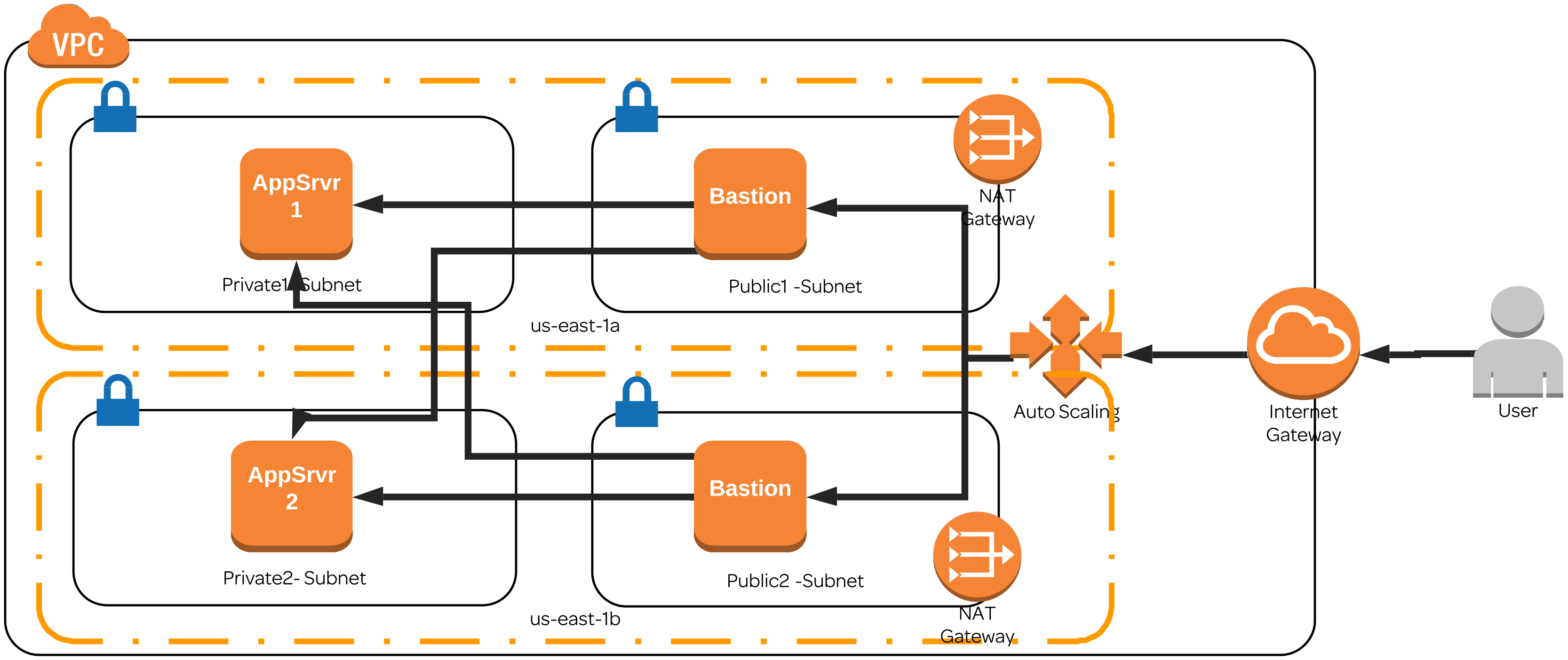
Windows Examples

Remote Desktop
- Use the private key with ec2config to decrypt the administrator password
- Use Remote Desktop to access the instance

Bash for Windows
- A bash shell can now be installed on Windows 10 (Anniversary Update)
- Link to a guide on installing is in the course downloads

PuTTY for Windows
- PuTTY is a terminal program for Windows that allows us to use SSH connections
- Link to a guide on running PuTTY and Pageant (for SSH forwarding) is in the course downloads

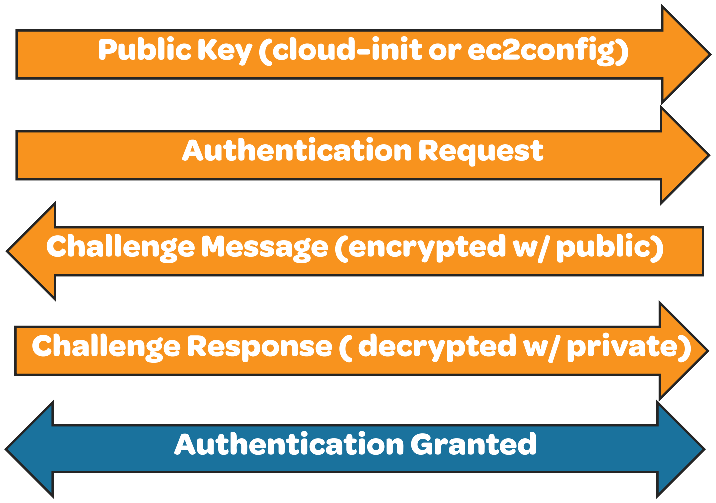
Infrastructure of the walkthrough example:



BACK

OS-Level Security

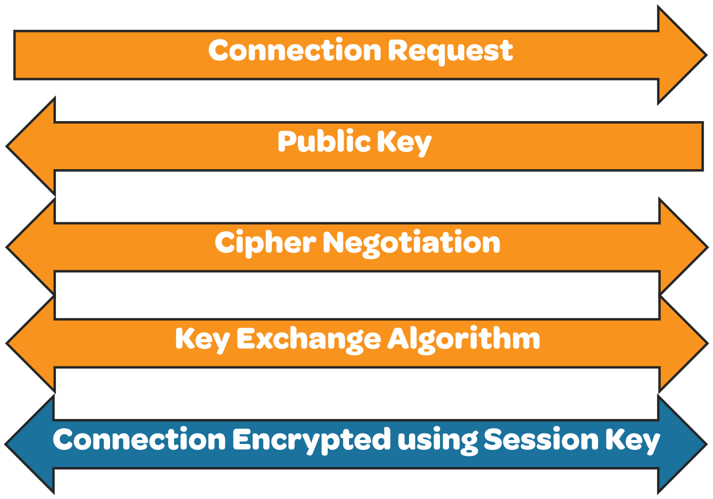
Overview | SSH | Bastion Host | Linux Example | Windows Example


Public Key (cloud-init or ec2config)
Authentication Request
Challenge Message (encrypted w/ public)
Challenge Response (decrypted w/ private)
Authentication Granted

OS-Level Security

BACK

Overview SSH Bastion Host Linux Example Windows Example



BACK

Data Security

Protecting Data at Rest

Decommission Data and Media

Protecting Data in Transit

Data Security

BACK

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

S3 | Glacier | EBS | RDS | DynamoDB | EMR

Implementation Concerns with Protecting Data at Rest

- Accidental information disclosure

- Data integrity compromised

- Accidental deletion

- Availability

BACK

Data Security

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

Concerns with communicating over public links (Internet)

- Accidental information disclosure
- Compromised data integrity
- Identity spoofing (man-in-the-middle)

Approaches for protecting data in transit

- Use HTTPS whenever possible for web applications (SSL/TLS)
- Can offload HTTPS processes to an Elastic Load Balancer if processing is a concern
- Remote Desktop Protocol accessible servers should have X.509 certificates to prevent identity spoofing
- SSH is preferred for administrative connections to Linux servers
- Database server traffic should use SSL/TLS as well (supported by most)
- AWS Console and AWS APIs use SSL/TLS for connections to clients

Example Services: S3, RDS, DynamoDB, EMR

X.509 certificates

- X.509 certificates are used by the client browser to authenticate identity
- Carries a public key and binds that key to an identity

AWS Certificate Manager

- Allows AWS users to easily create and manage SSL/TLS Certificates
- Works with:
    - Elastic Load Balancer
    - Amazon CloudFront
    - API Gateway
    - Elastic Beanstalk/CloudFormation (for deployment)
- Automatic certificate renewal
- Import third-party certificates as well
- Free

Data Security

BACK

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

S3 - Protecting Data at Rest

Permissions
- Bucket-level and object-level permissions along with IAM policies
- Rule of least privilege
- MFA delete

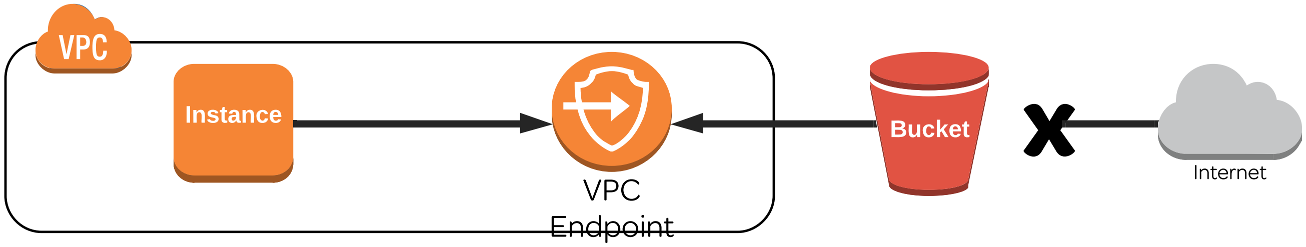
Versioning
- Enable to store new versions for every modification or delete
- Helps with accidental deletion by creating a version for deleted objects

Replication
- Objects are replicated across Availablity Zones automatically
- Standard and reduced redundancy options at different price points

Backup
- Replication and versioning make backups unneccessary
- Can set rules to back up objects to another region

Server-side encryption
- Use either S3 master-key or KMS master-key
- Assists with accidental data exposure as long as the keys are not compromised

VPC Endpoint
- Can use data inside the VPC without making it public



BACK

Data Security

Protecting Data at Rest

Decommission Data and Media

Protecting Data in Transit

Elastic Block Storage - Protecting Data at Rest

Replication
- EBS stores two copies of each volume in the same Availability Zone
- Helps with hardware failures but is not intended to help availability

Backup
- Snapshots (point in time captures)
- Can use IAM to control access to these snapshot objects

Server-side Encryption
- AWS KMS master-key
- Microsoft Encrypted File System
- Microsoft Bitlocker
- Linux dmcrypt
- Third-party solutions

BACK

Data Security

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

Relational Database Service - Protecting Data at Rest

Permissions

- Use IAM policies on users, groups, and roles to limit access
- Rule of least privilege

Encryption

- Key Management Service (KMS) is integrated for most instance sizes (not t2.micro)
- MySQL, Oracle, and Microsoft SQL have cryptographic functions at the platform level
  - Keys are managed at the application level
  - Must reference the encryption and key in queries on encrypted database fields

BACK

Data Security

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

Glacier - Protecting Data at Rest

Server-side encryption

- All data stored is encrypted using AES-256
- Each archive gets a unique key
- A master key is then created and stored securely

BACK

Data Security

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

DynamoDB - Protecting Data at Rest

Permissions

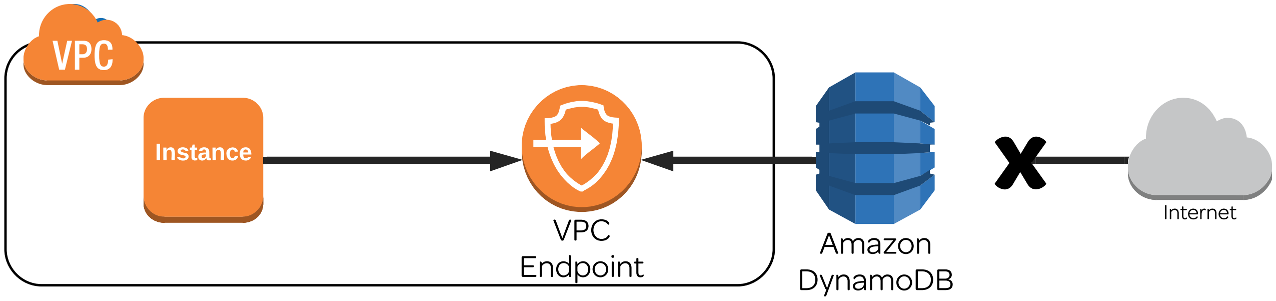
- Use IAM policies on users, groups, and roles to limit access
- Rule of least privilege

Encryption

- Same as RDS; can encrypt at the application layer (effects the query process)
- Best practice to use raw binary or Base64-encoded fields when storing encrypted fields

VPC Endpoint

- Can use data inside the VPC without making it public



Elastic Map Reduce - Protecting Data at Rest

Amazon-managed service

- AWS provides the AMIs (no custom AMIs)
- EMR instances do not encypt data at rest

Data Store

- S3 or DynamoDB
- HDFS (Hadoop Distributed File System)
- If HDFS, AWS defaults to Hadoop KMS

Techniques to improve data security

- S3 server-side encryption (if not using HDFS)
- Application-level encryption
- Hybrid

BACK

Data Security

Protecting Data at Rest | Decommission Data and Media | Protecting Data in Transit

Decommision Data and Media Securely

Different than on-prem decommisioning

- When a delete request is made, AWS does not decommision the underlying hardware
    - Storage blocks are marked unallocated
    - Secure mechanisms reassign the blocks elsewhere

Reading and writing to blocks

- When an instance writes to a block of storage:
    - Previous block is overwritten
    - Then overwritten with your data
- If instance reads from a block previously written to:
    - Previously stored data is returned
    - If there is no previous data from that instance, the hypervisor returns a zero

End of life

- AWS follows techniques in:
    - DoD 5220.22-M ("National Inidustrial Security Program Operating Manual")
    - NIST SP 800-88 ("Guidelines for Media Sanitization")
    - Both documents are in the downloads section for this course
- If the device is unable to adhere to these two standards, it is physically destroyed.

OS Security

BACK

Custom AMIs | Bootstrapping | Systems Manager | Malware | Mitigating Abuse

Recommendations

- Disable root user API access keys
- Use limited source IPs in security groups
- Password protect .pem files on user machines
- Keep authorized_key file up to date on your instances
- Rotate credentials (access keys)
- Use Access Advisor to identify and remove unnecessary permissions
- Bastion hosts

Above all:

- Develop configuration standards for all resources and regularly review them.

OS Security

BACK

Custom AMIs | Bootstrapping | Systems Manager | Malware | Mitigating Abuse

Securing Custom AMIs

- AMIs can be public or private
- Allows for a "base" configuration to be deployed on instances:
    - Operating system
    - Applications
    - Security settings (authorized_keys, local accounts, file and directory permissions)
- **IMPORTANT:** This is a point in time "snapshot" of an instance and therefore needs to be updated frequently to include new and changed configuration standards.

Clean-up/Hardening tasks to perform before publishing

- Disable insecure applications (ex. Telnet)
- Minimize exposure - Disable all ports except management and those necessary for the application itself.
- Protect credentials:
    - Access keys, certificate, or third-party credentials are deleted.
    - Sofware should not be using default accounts
    - SSH keys must not be published
    - Disable guest account (Windows)
- Protect data:
    - Delete shell history and logs (event log on Windows)
- Remove printer and file sharing, or any other sharing service that is on by default (Windows)
- Make sure your systems do not violate AWS Acceptable Use Policy
    - Example: Open SMTP relays or proxy servers
    - A link to this policy will be provided in the downloads of this course.

OS Security

BACK

Custom AMIs | Bootstrapping | Systems Manager | Malware | Mitigating Abuse

Bootstrapping

cloud-init, cfn-init, tools like Puppet and Chef

Considerations:

- Patching/Updates
    - Dependencies should be coinsidered
    - Security software updates might update beyond the patch level of the AMI
    - Application updates might patch beyond the build in the AMI
    - Solution: Keep AMIs updated frequently
- Bootstrapping configurations should take into account differences in:
    - Production environmemt
    - Test environment
    - DMZ/Extranet environment
- Instance updates might break external management and security monitoring
    - Test on non-critical resources

OS Security

BACK

Custom AMIs | Bootstrapping | Systems Manager | Malware | Mitigating Abuse

AWS Systems Manager Features

Resource Groups: Allows you to group your resources logically (Prod, Test, DMZ)

Insights: Aggregates CloudTrail, CloudWatch, Trusted Advisor, and more into a single dashboard for each resource group

Inventory: A listing of your instances and software installed on them
- Can collect data on applications, files, network configs, services, and more

Automation: Automate IT operations and management tasks through scheduling, triggering from an alarm, or directly

Run Command: Secure remote managment replacing need for bastion hosts or SSH

Patch Manager: Helps deploy OS and software patches across EC2 or on-prem

Maintenance Window: Allows for scheduling administrative and maintenance tasks

State Manager and Parameter Store: Used for configuration management

OS Security

BACK

Custom AMIs | Bootstrapping | Systems Manager | Malware | Mitigating Abuse

Malware

- Executing untrusted code on a system can result in rootkits, botnets, and more
- That system is no longer yours

Combat malware by:

- Use only trusted AMIs, software, and software depots
- Use the principle of least privilege
- Keep patches up to date (which means updating AMIs regulary, as well)
- Use antivirus/antispam software
- Host-based IDS (can detect rootkits and check file integrity)

Suggested resolutions:

- Antivirus might be able to "clean" the system
- Best practice: Save the data and reinstall the system, application, and data from trusted sources.
    - Could be as simple as terminating the instance (Auto Scaling)

OS Security

BACK

Custom AMIs | Bootstrapping | Systems Manager | Malware | Mitigating Abuse

Mitigating Abuse and Compromise

Abuse activities: Externally observed behavior of AWS customer's instances or resources that are mailicious, offensive, illegal, or could harm other internet sites

AWS will shut down malicious abusers, but many of the abuse complaints are about customers conducting legitimate business on AWS.

Causes of abuse that is not intentional:

- Compromised resource - EC2 instance becoming a botnet
- Unintentional abuse - Web crawlers can sometimes register as a DOS attack
- Secondary abuse - End user of your services posts an infected file
- False complaints - Internet users mistake legitimate activities for abuse

Best practices for response to abuse:

- Do not ignore AWS abuse communications and make sure they have the most effective email address on file
- Follow security best practices
- Mitigate identified compromises

Infrastructure Security

BACK

VPC

Segmentation

Strengthening

Threat Protection

Testing/ Measurement

WAF & Shield

AWS Security Essentials

BACK

Infrastructure Security

VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

VPC Security

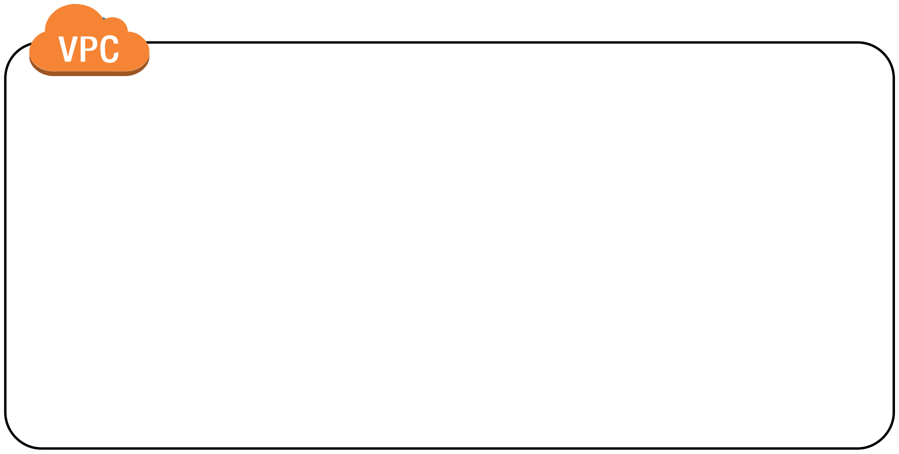
Connecting to your VPC: (click the boxes to display the diagrams)

Internet Only

IPSec tunnel over Internet

AWS Direct Connect

Hybrid



AWS Security Essentials

Infrastructure Security

BACK

VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Network Segmentation

VPC (virtual private cloud)

- Isolate workloads into separate VPCs (based on application, department, test, dev, etc.)

Security groups

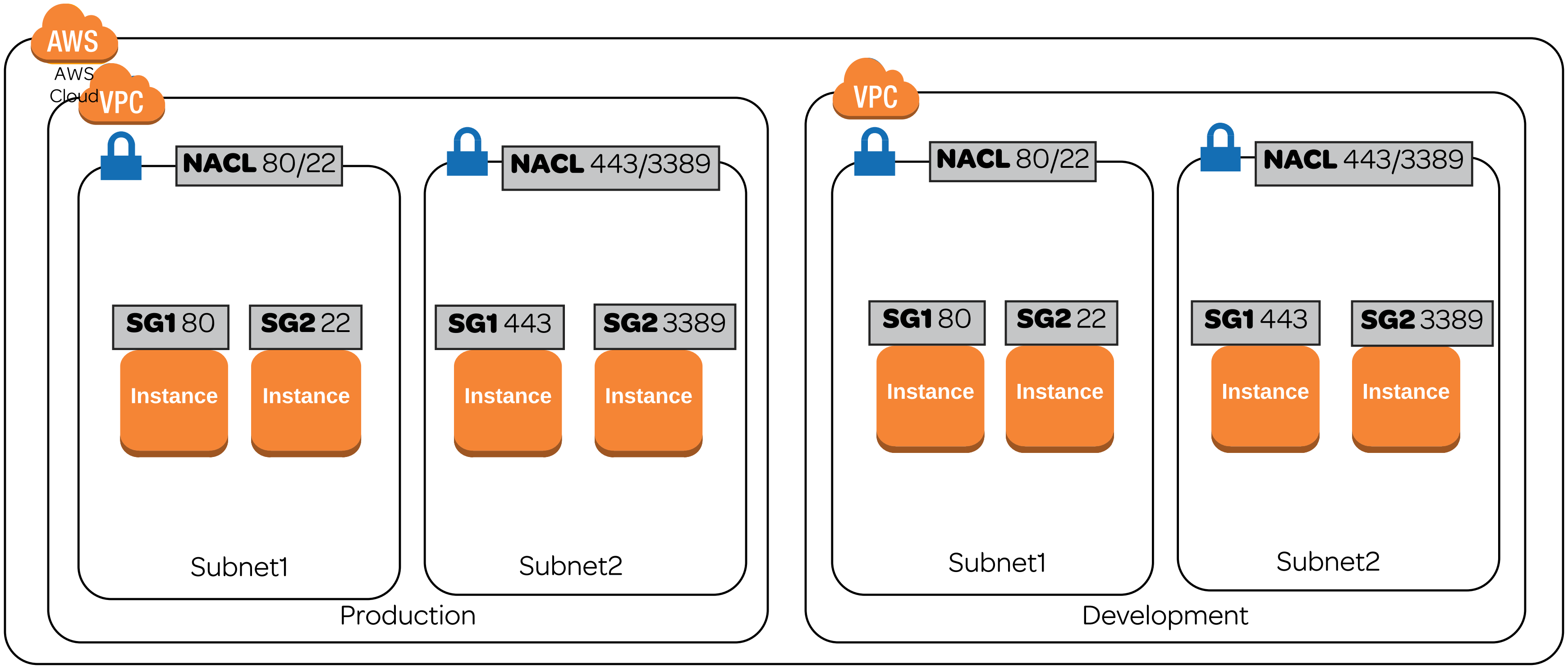
- Group instances with similar functions.
- Stateful = every allowed TCP or UDP port will be allowed in both directions

NACLs (network access control lists)

- Stateless = inbound and outbound rules are separate, no dependencies
- Granular control over IP protocols (allow and deny rules for inbound and outbound evaluated **in order**)
- Work with security groups (NACL applies for the whole subnet, security groups apply to members
- Ephemeral ports - Client requests depending on OS (ports 1024-65535)

Host-based firewalls

- OS-level firewalls



BACK

Infrastructure Security

VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Strengthening the Network

Customer side of the shared responsiblity model

- Controlling access
- Network security within your VPCs
- Securing traffic inbound and outbound

Best Practices

- Use security groups (multiples can be applied to an instance)
- Use NACLs to prevent unwanted traffic from entering the subnet to begin with
- Use Direct Connect or IPSec for connections to other sites
- Encrypt data in transit as often as possible
- Layer your network security (public and private subnets)
- Check VPC flow logs to gain information about traffic in your VPC

Secure periphery systems

- DNS - Use SSL/TLS to prevent spoofing; Route 53 is secure
- Active Directory, LDAP - Use directory service to further secure
- Time Servers - Centralize a server for syncing to a trusted source, then direct other resources there
- Repositories - Do not post crendentials to repos like GitHub (SSH access keys. API keys, etc.)

BACK

Infrastructure Security

VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Threat Protection Layers

- Layered security is generally considered best practice, not just at the edge
- Main concern is "untrusted" connections
- Organizations have varying requirements regarding this.

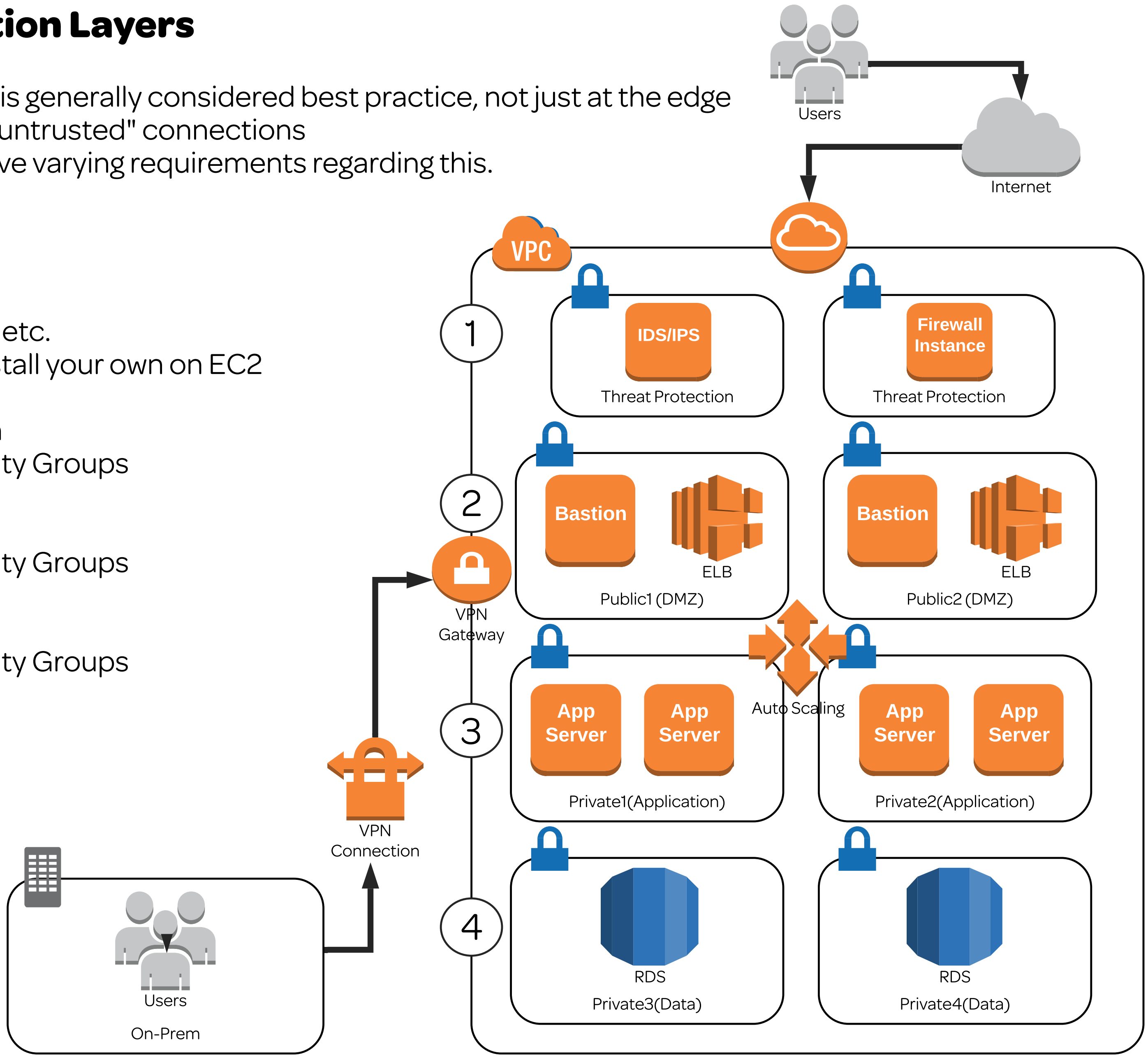
Layers

1. Threat Protection
   - IDS, IPS, Firewall, etc.
   - Third party or install your own on EC2

2. DMZ/ Presentation
   - NACL and Security Groups

3. Application
   - NACL and Security Groups

4. Data
   - NACL and Security Groups



Infrastructure Security

Testing and Security Management

Vulnerability/ Risk Assessment

- Third party evaluation with little "inside" knowledge of the infrastructure
- Aspects can be overlooked with familiarity

Penetration Testing

- AWS must be notified before any testing is done
  - Third party or in-house
- AWS Vulnerability Penetration Testing Form
  - Must be submitted
  - Contains information about specific times, instances, and a terms and conditions agreement
- Cannot test m1.small or t.1 micro instance types

Measuring Your Risk Management

- Monitor procedures - Detecting security events, identifying breaches, effective actions
- Review effectiveness regularly - Audits, incidents, effectiveness feedback, along with improvement plans
- Measure effectiveness - Security requirements have been met
- Risk assessments at regular intervals - The security landscape changes over time
- Internal audits - Conducted by the organization itself
- Management reviews- Making sure the scope is accurate and relevant

Infrastructure Security

BACK

VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Protecting Web Applications

AWS Web Application Firewall (WAF)

- Allows for conditions or rules to be set on web traffic on CloudFront or an Application Load Balancer
- WAF can watch for cross-site scripting, IP addresses, location of requests, query strings, and SQL injection.
- When mulitple conditions exist in a rule, the result ANDS
    - **Example Rule:** Requests from 2.2.0.0/16 that appear to have SQL code
        - Both conditions must match for a block

Denial of service attacks

- Flooding a system with traffic to overwhelm and prevent legitimate traffic access to resources
- Distributed DoS (DDoS) is that same attack from multiple sources or systems
- AWS provides resilience for network and transport layer attacks.
- Web application attacks can be handle AWS Shield

AWS Shield Standard

- The basic level of DDoS protection for your web applicaitons
- Included with WAF. No additional cost

AWS Shield Advanced

- Expands services protected to include Elastic Load Balancers, Cloudfront Distributions, Route 53 hosted zones, and resources with Elastic IPs
- For EIPs, promotes your NACL to the AWS border (10 Gbps to Terrabytes of traffic)
- Contact 24x7 DDoS response Team (DRT) for assistance during an attack
- Some cost protection against spikes in a bill from DDoS attacks
- Expanded protection against many types of attacks
- WAF is included in Shield Advanced pricing
    - $3,000/month per organization
    - Plus Data Transfer Out usage fees

AWS Security Essentials

BACK

Infrastructure Security

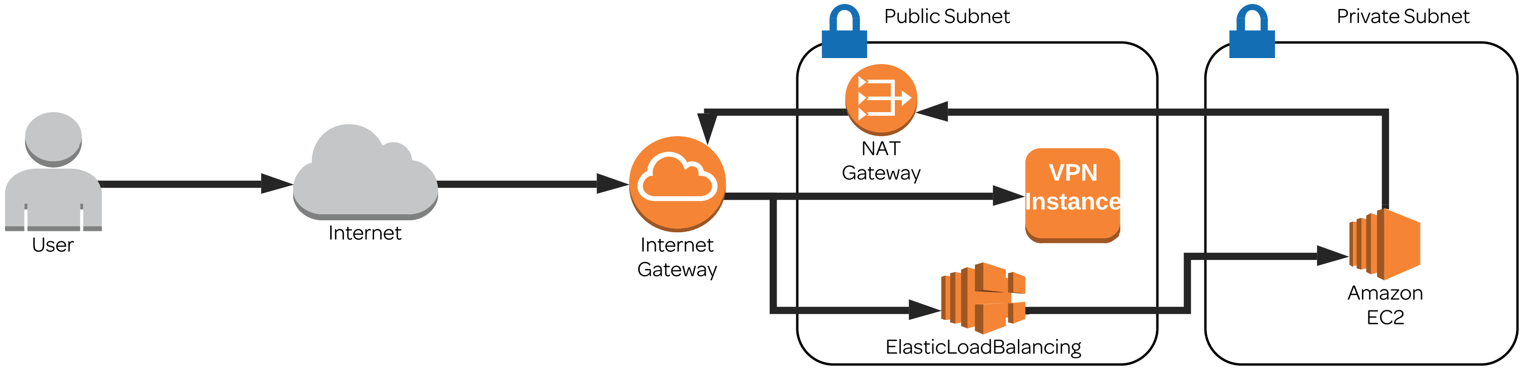
VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Scenarios

- No connection between your infrastructure and the VPC
- No on-premises infrastructure

Best Practice

- Use SSL/TLS endpoints for your applications
- Build your own VPN Solutions
- Routing and placement must be planned (public and private subnets)
- Security Groups and network access control lists (NACLs)



AWS Security Essentials

BACK

Infrastructure Security

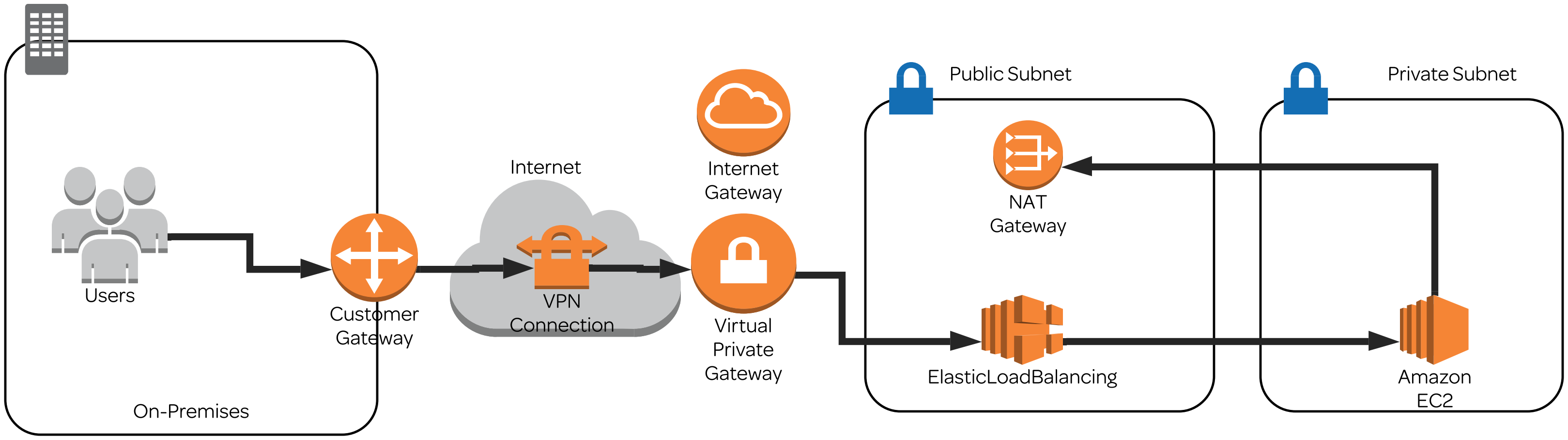
VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Scenarios

- Your organization requires secured communications
- Lesser need for dedicated throughput

Best Practice

- Deploy VPN using standard AWS VPN componenets (VPN Gateway, Customer Gateway, VPN Connection)
- Can also use custom VPN solutions if required
- VPC networking (subnets, security groups, NACLs)



BACK

Infrastructure Security

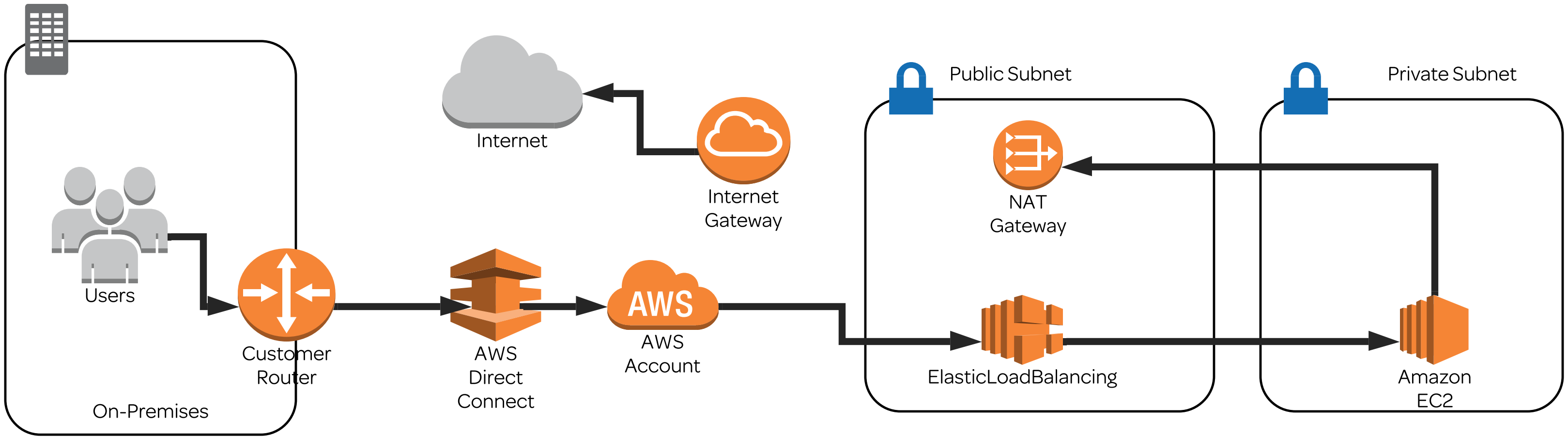
VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Scenarios

- Your organization requires dedicated links to peer to AWS
- There is a need for dedicated throughput (1 Gbps and 10 Gbps)

Best Practice

- Using a private peered connection like this might not need additional security
- Check your organization's requirements
- VPC networking (subnets, security groups, NACLs)



Infrastructure Security

BACK

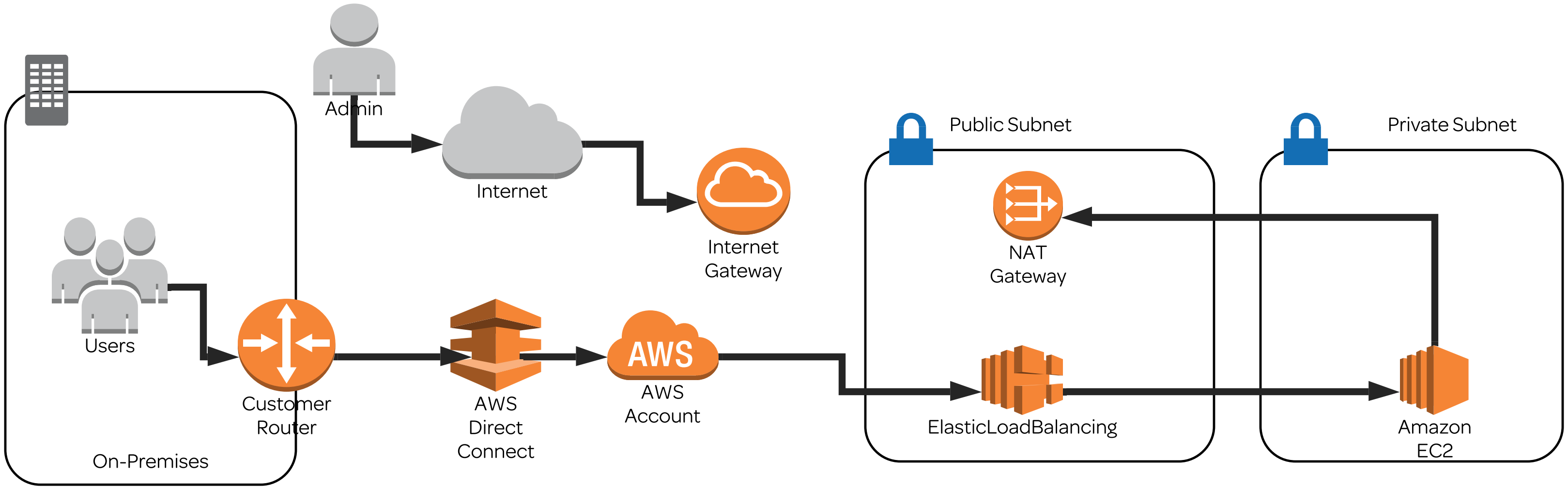
VPC | Segmentation | Strengthening | Threat Protection | Testing/ Measurement | WAF & Shield

Scenarios

- Your organization might have Direct Connect, but some users connect with internet only
- Your organization might require IPSec tunnels be used over Direct Connect

Best Practice

- Refer to best practices for each type of connection you are using in a hybrid environment
- VPC networking (subnets, security groups, NACLs)


Admin
Internet
Internet Gateway
Public Subnet
Private Subnet
NAT Gateway
Users
Customer Router
On-Premises
AWS Direct Connect
AWS
AWS Account
ElasticLoadBalancing
Amazon EC2