Cloud Security

Dr Hisain Elshaafi

Outline

- Cloud Computing
- Shared Responsibility
- Cloud VMs
- Virtual Private Cloud
- Securing VPCs
- Data Security
- Connecting Networks and services
- Identity and Access Management
- Cloud Security Monitoring
- Security-Related Best Practices
- Edge Caching
- Automating Architecture
- Securing AWS Account

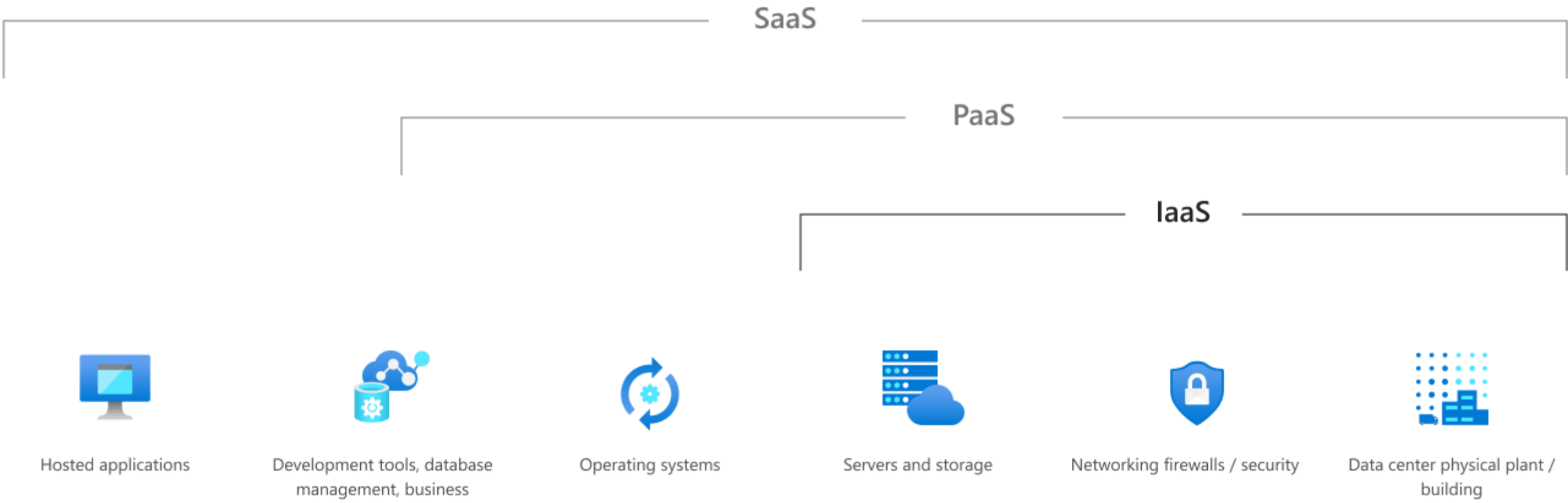
Cloud Computing

Cloud Computing

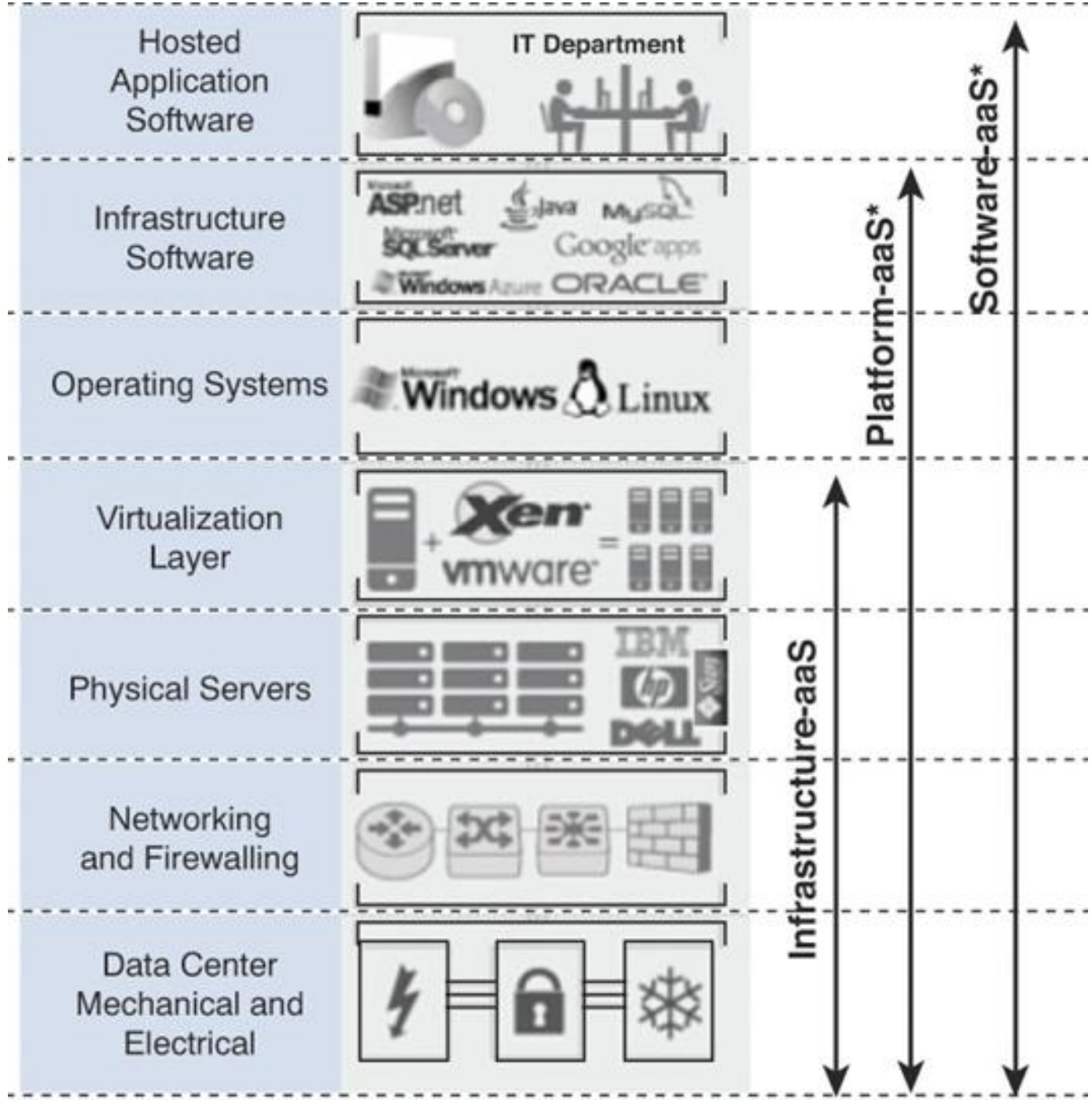
- Shared pool of configurable **computing resources**
  - Networks, servers, storage, applications, services, …
  - **Rapidly provisioned/**released with minimal management effort or SP interaction
  - Eliminates financial and technological **barriers**

Characteristics of Cloud Services

- Ubiquitous access via Internet technologies
- Shared infrastructure
- Lower costs and variable pricing
- Flexible and scalable services
- Dynamic provisioning
- Service orientation
- Managed operations

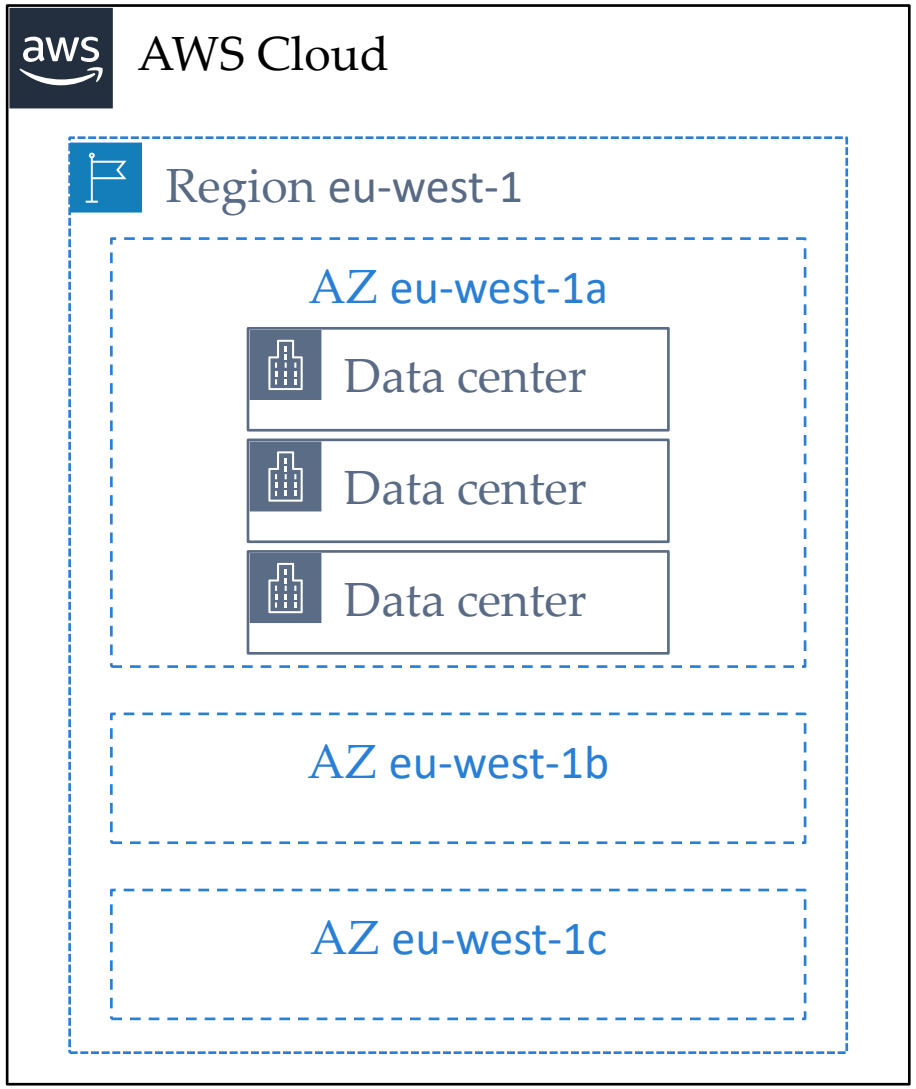


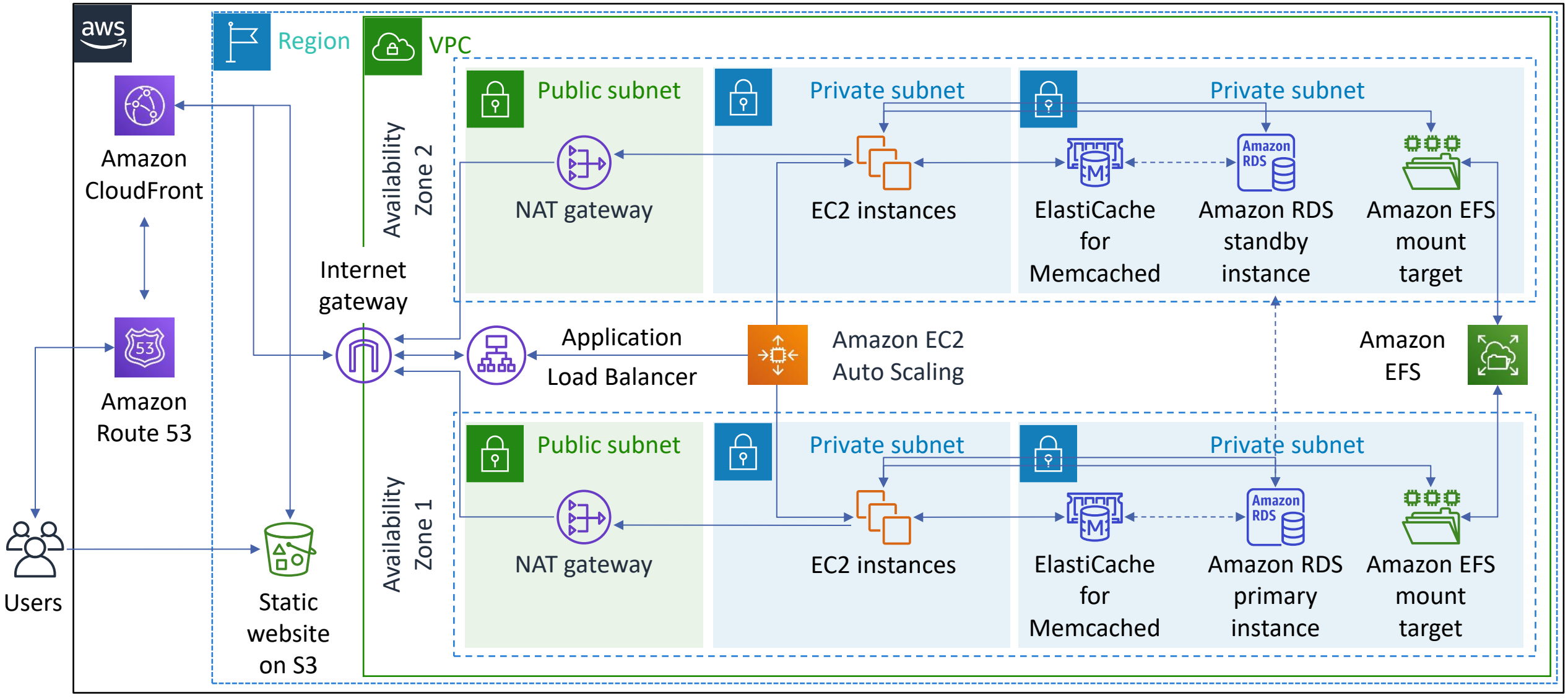
https://azure.microsoft.com/en-us/overview/what-is-paas/


Hosted Application Software
IT Department
Infrastructure Software
ASP.net
Java
MySQL
SQLServer
Google apps
Windows Azure
ORACLE
Operating Systems
Windows
Linux
Virtualization Layer
Xen
vmware
Physical Servers
IBM
DELL
Networking and Firewalling
Data Center Mechanical and Electrical
Infrastructure-aaS
Platform-aaS*
Software-aaS*

Regions and Availability Zones

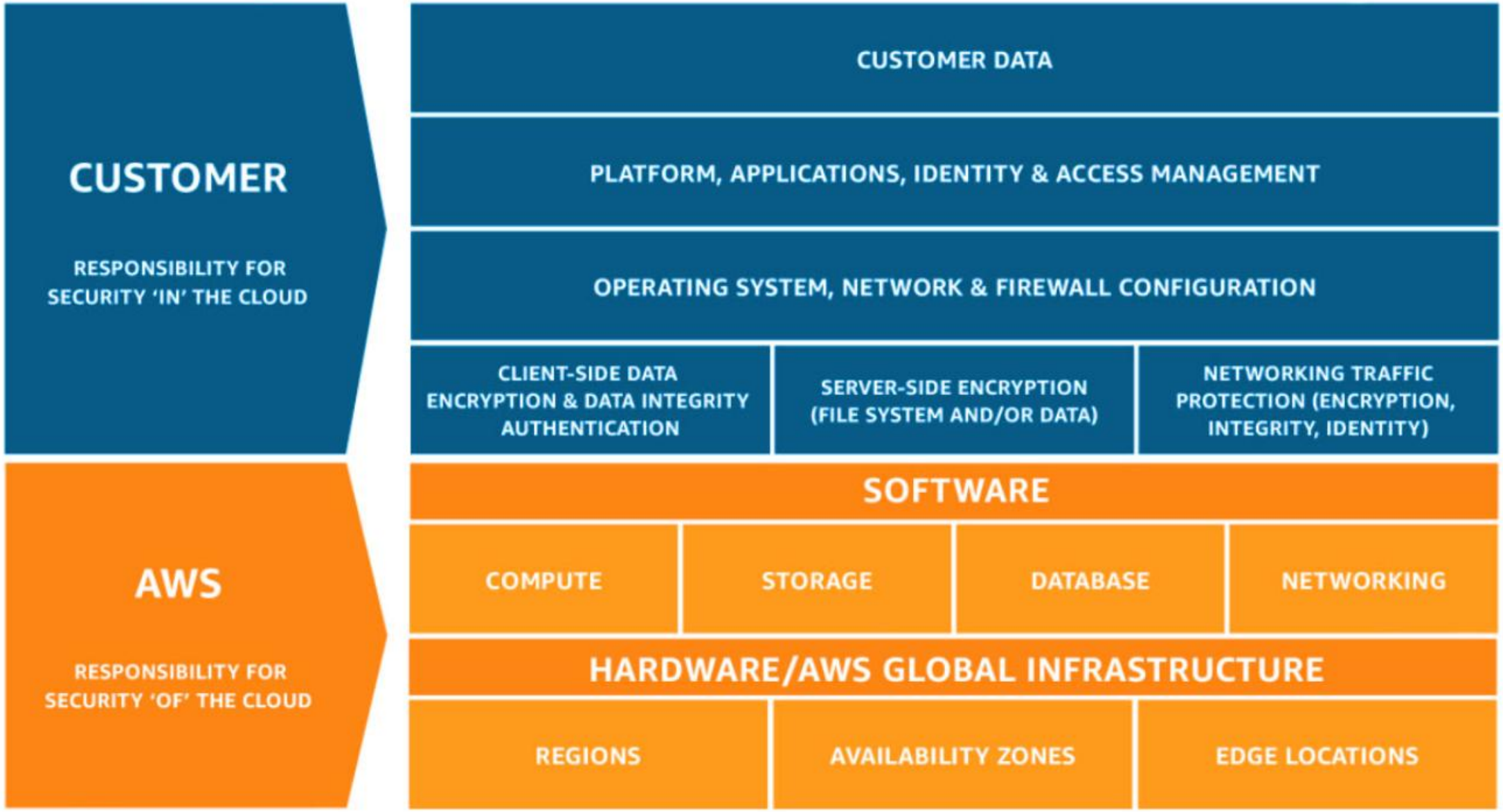
- Each region = 2 or more AZ
- Each AZ
  - 1-6 data centres
  - Independent failure zone
  - **Fault isolation**
  - Interconnected with other AZs in a region using high-speed **private links**
- AWS recommends **replicating** across AZs for **resiliency**




aws
Region
VPC
Amazon CloudFront
Amazon Route 53
Users
Static website on S3
Internet gateway
Availability Zone 2
Availability Zone 1
Public subnet
Private subnet
Private subnet
NAT gateway
EC2 instances
ElastiCache for Memcached
Amazon RDS standby instance
Amazon EFS mount target
Application Load Balancer
Amazon EC2 Auto Scaling
Amazon EFS
Public subnet
Private subnet
Private subnet
NAT gateway
EC2 instances
ElastiCache for Memcached
Amazon RDS primary instance
Amazon EFS mount target

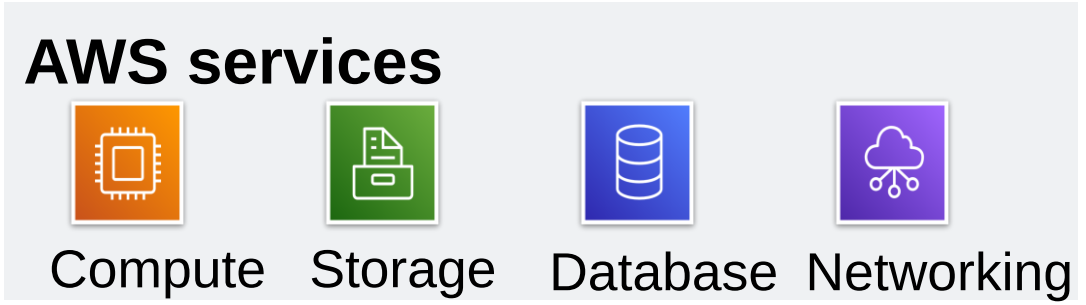
Shared Responsibility

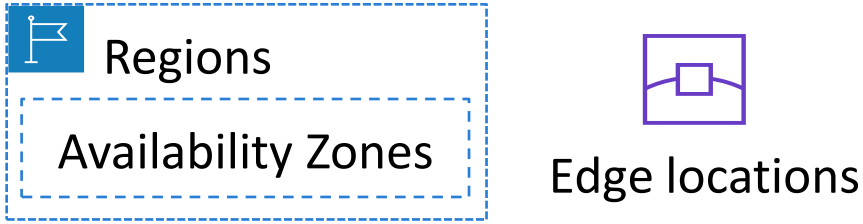
Shared Responsibility Model



AWS Responsibility

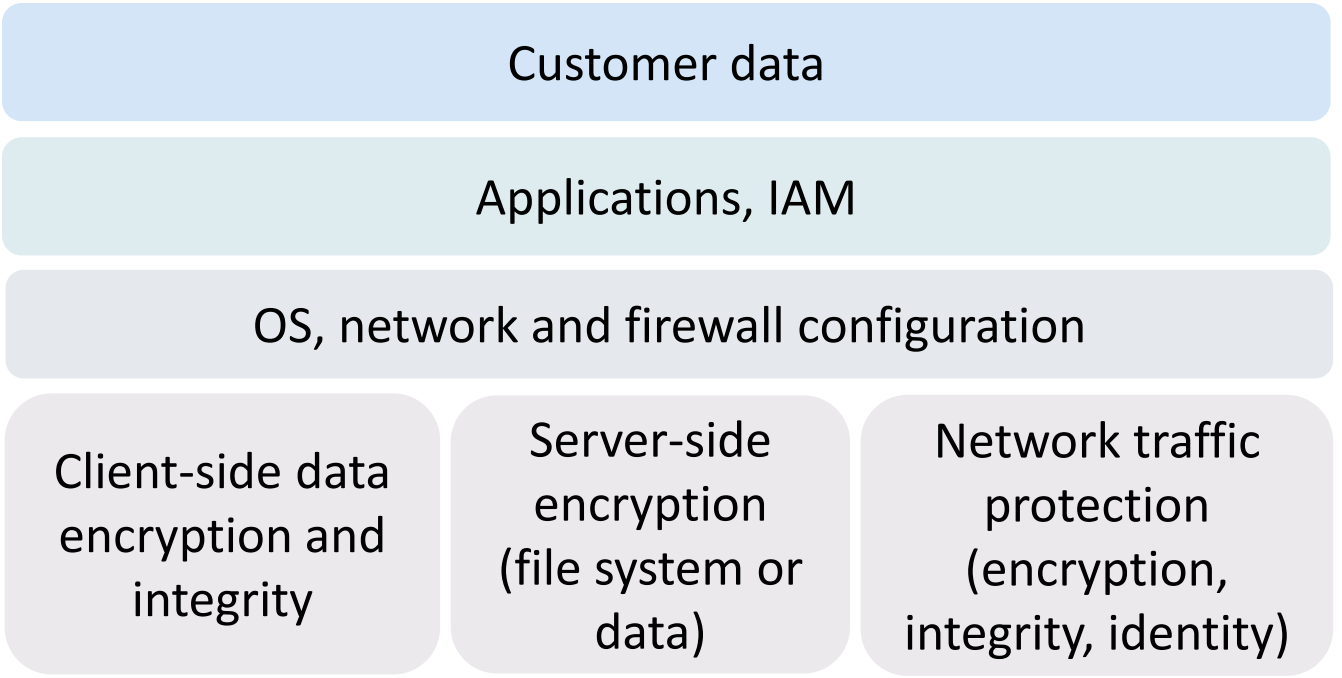
- **Physical** security of data centers
  - Controlled need-based access
- H/w and s/w **infrastructure**
  - Storage decommissioning, host OS access logging and auditing (3rd party)
- **Network** infrastructure
  - Routers, switches, firewall, IDS
- **Virtualization** infrastructure
  - Instance isolation





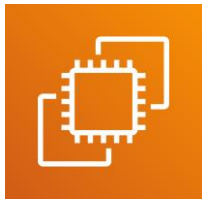
Customer Responsibility

- **EC2** instance OS
  - Including patching, maintenance
- Applications
  - Passwords, role-based access, etc
- **Security group** configuration
- Host-based **firewalls**
  - Including IDS, IPS
- **Network** configurations
- **Account** management
  - Login and permission settings for users



IaaS and Security Responsibility

- Customer
  - has more flexibility over **configuring networking** and **storage** settings
  - is responsible for managing **more aspects** of security
  - configures **access controls**



Amazon EC2

Amazon Elastic Block Store (EBS)

Amazon Virtual Private Cloud (VPC)

PaaS and Security Responsibility

- Customer **does not** need to manage infrastructure
- **AWS** handles OS, database patching, firewall configuration and disaster recovery
- Customer focus on managing **code or data**

AWS Lambda

Amazon RDS

AWS Elastic Beanstalk

SaaS and Security Responsibility

- Software is centrally hosted
- Licensed on **subscription** model or **pay-as-you-go** basis
- Services are accessed via **web browser**, mobile app or API
- Customers **do not** need to manage infrastructure that supports the service

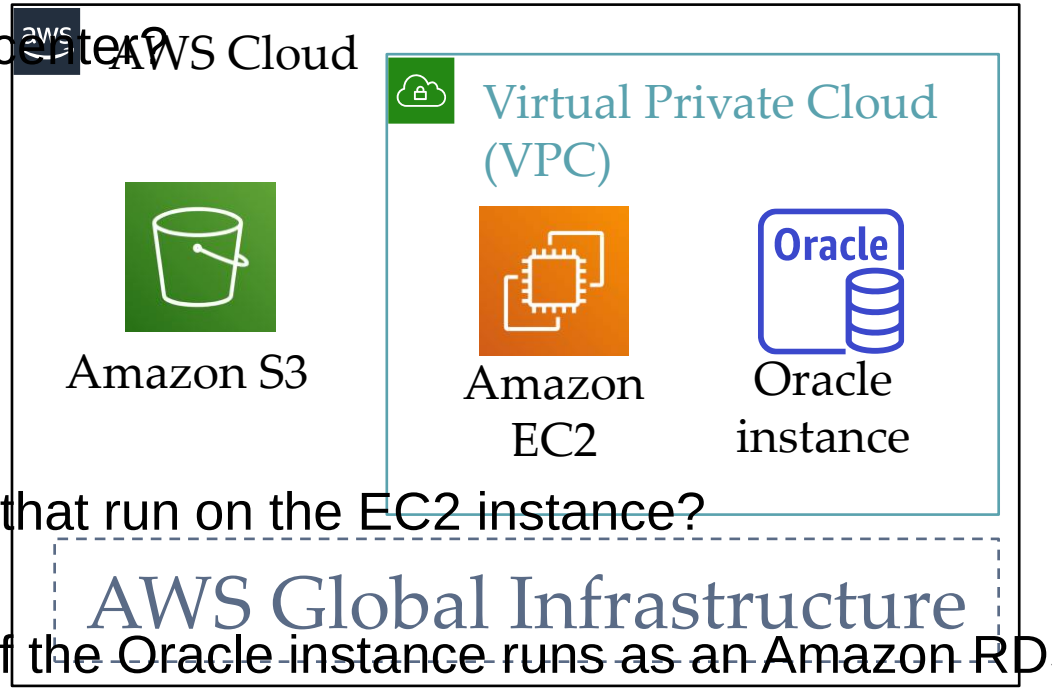
AWS Trusted Advisor

AWS Shield

Amazon Chime

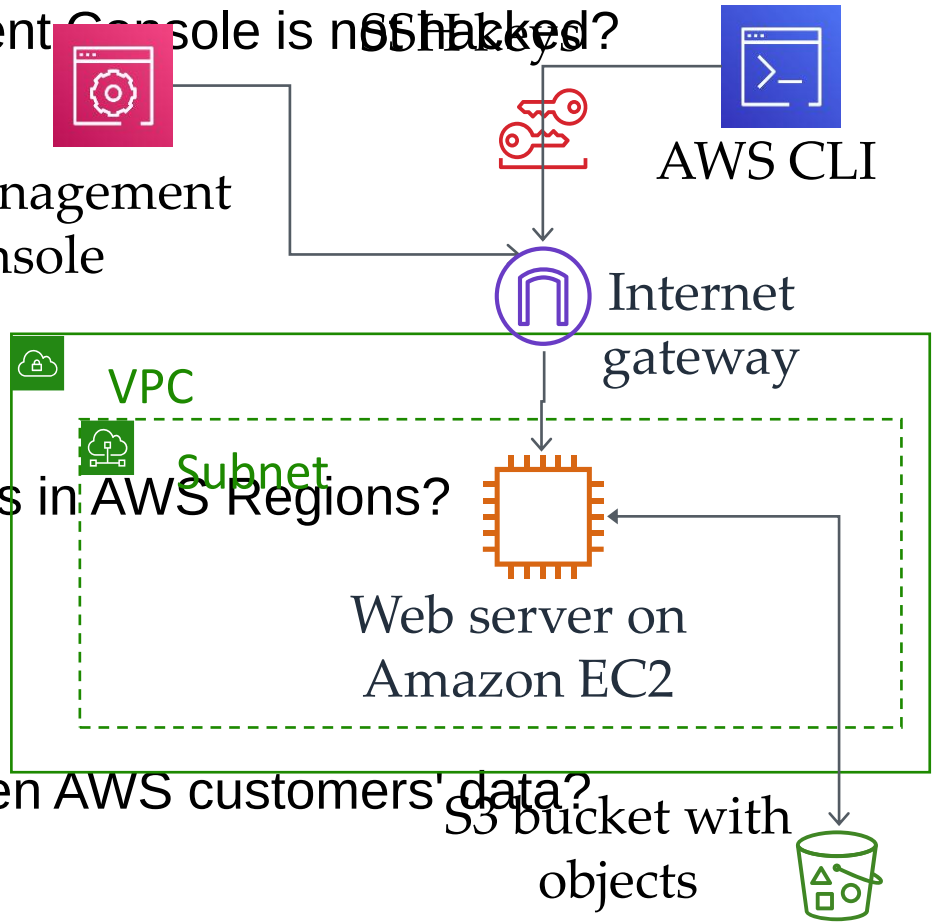
Who is Responsible?

1. Upgrades and patches to the OS on the EC2 instance?
   - **ANSWER:**
2. Physical security of the data center?
   - **ANSWER:**
3. Virtualization infrastructure?
   - **ANSWER:**
4. EC2 security group settings?
   - **ANSWER:**
5. Configuration of applications that run on the EC2 instance?
   - **ANSWER:**
6. Oracle upgrades or patches If the Oracle instance runs as an Amazon RDS instance?
   - **ANSWER:**
7. Oracle upgrades or patches If Oracle runs on an EC2 instance?
   - **ANSWER:**
8. S3 bucket access configuration?
   - **ANSWER:**



Who is Responsible?

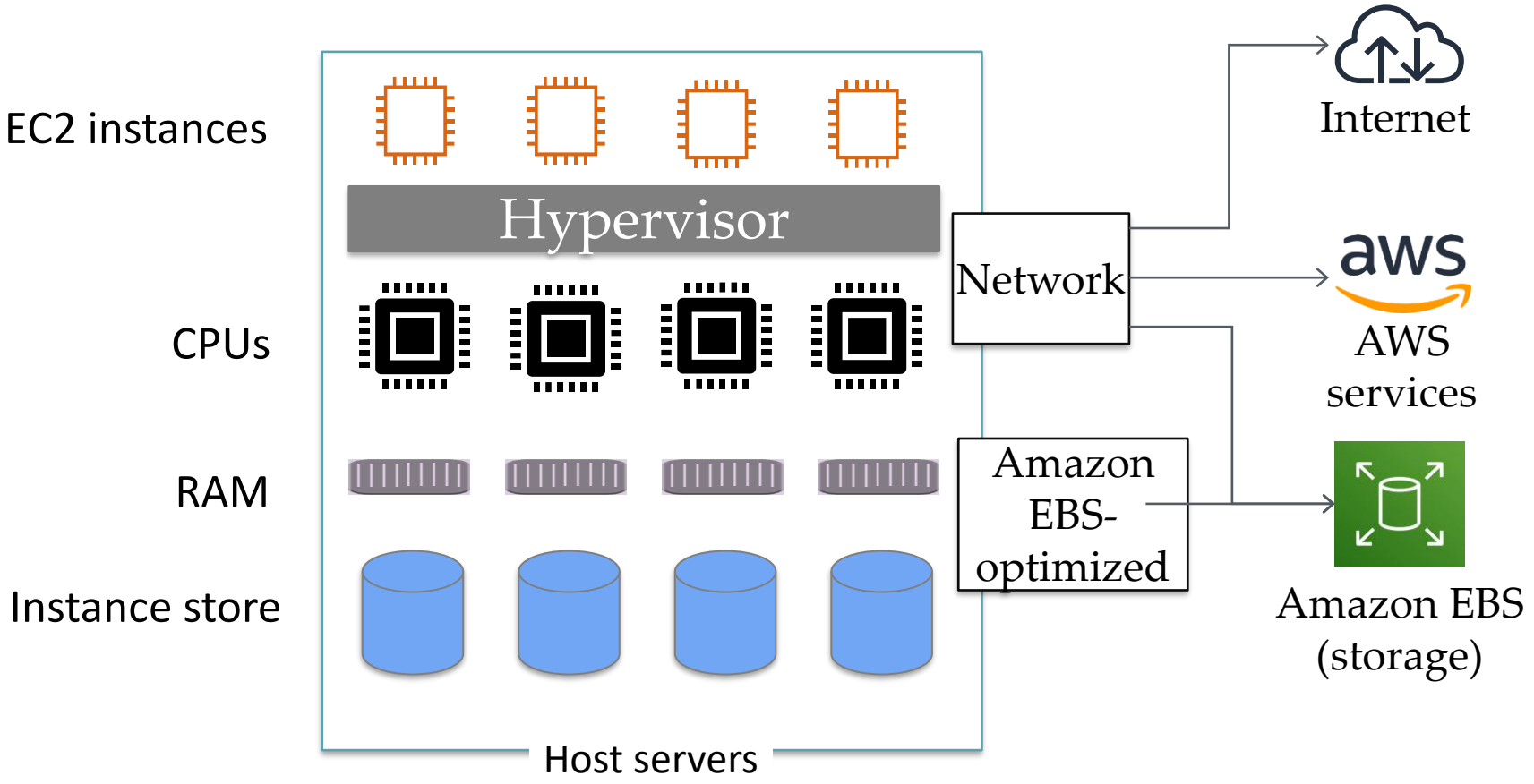
1. Ensuring that the AWS Management Console is not hacked?
   - **ANSWER:**
2. Configuring the subnet?
   - **ANSWER:**
3. Configuring the VPC?
   - **ANSWER:**
4. Protecting against network outages in AWS Regions?
   - **ANSWER:**
5. Securing the SSH keys
   - **ANSWER:**
6. Ensuring network isolation between AWS customers' data?
   - **ANSWER:**
7. Ensuring low-latency network connection between the web server and the S3 bucket?
   - **ANSWER:**
8. Enforcing multi-factor authentication for all user logins?
   - **ANSWER:**



Cloud VMs

Amazon EC2

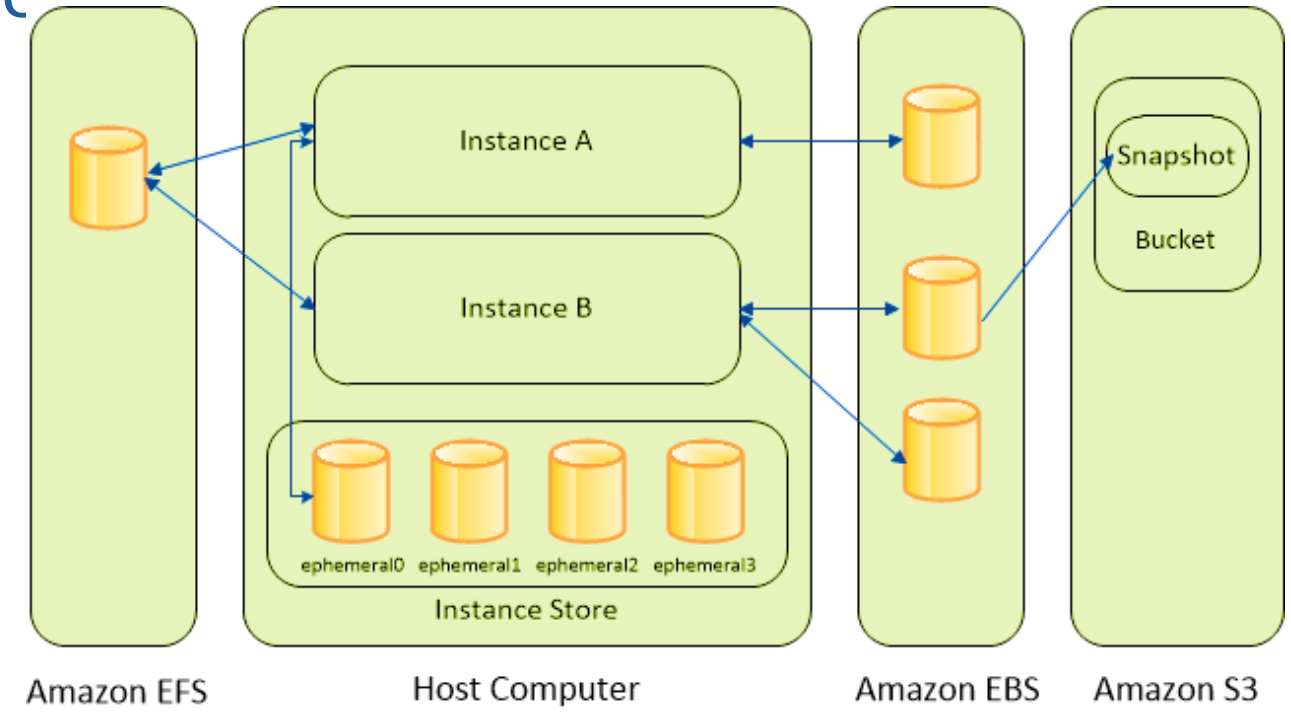
- Resizable **compute** capacity in the cloud
- EC2 instance is a **VM** that runs on physical host



Amazon EC2 virtualization

Amazon EC2 (cont)

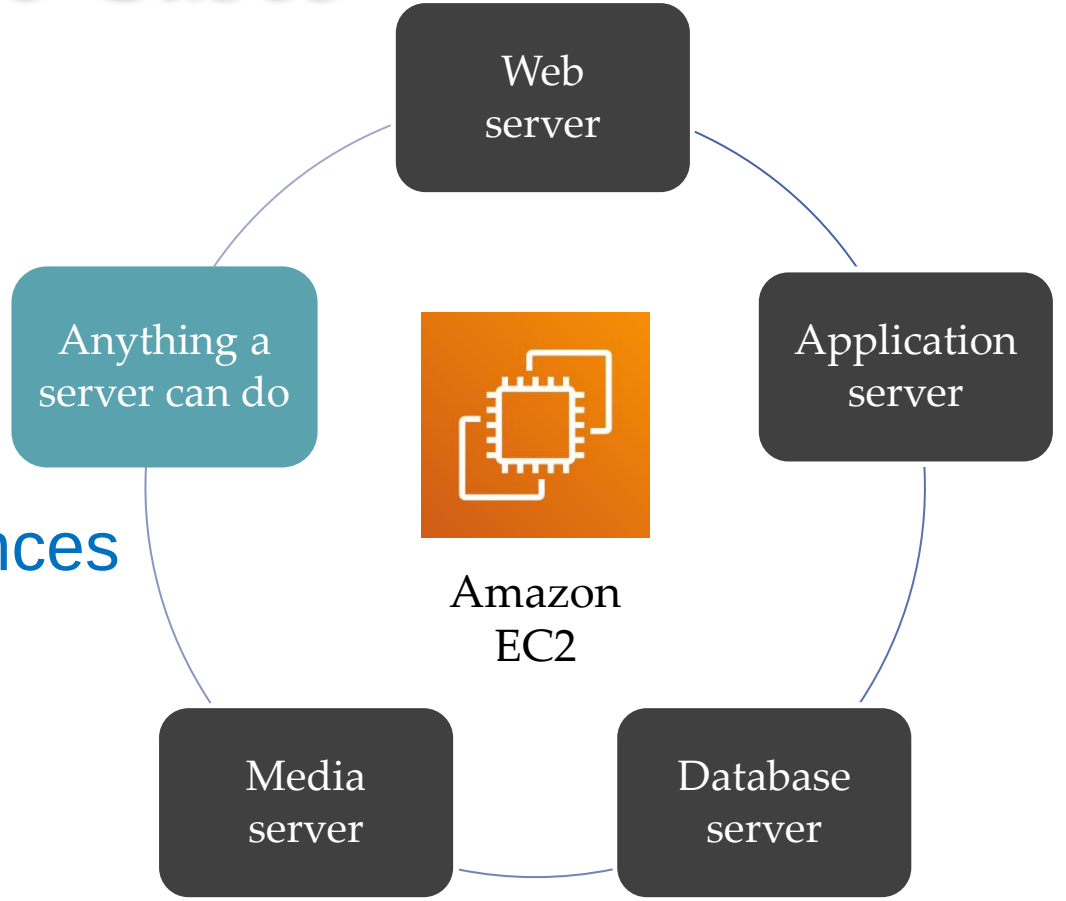
- **Different configurations** of CPU and memory capacity
- Provisions servers in **minutes**
- Can **scale capacity up or down**
- Pay for the **capacity** that you use
- **Storage** options
  - Instance store
  - **EBS**
  - EFS
  - S3



Amazon EC2 Use Cases

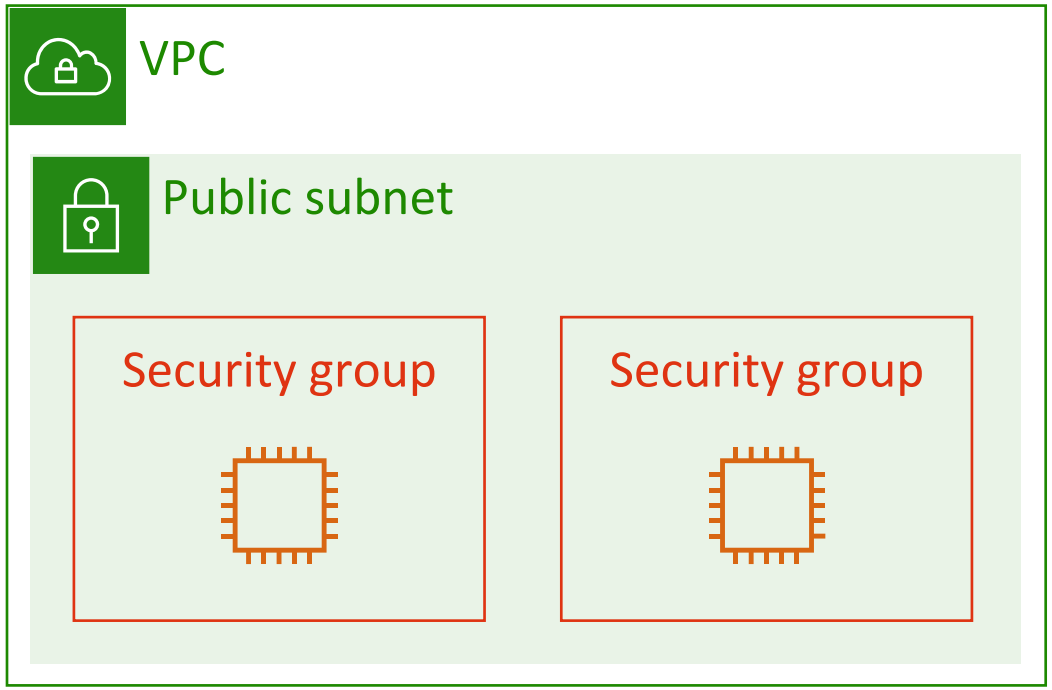
Use EC2 when you need

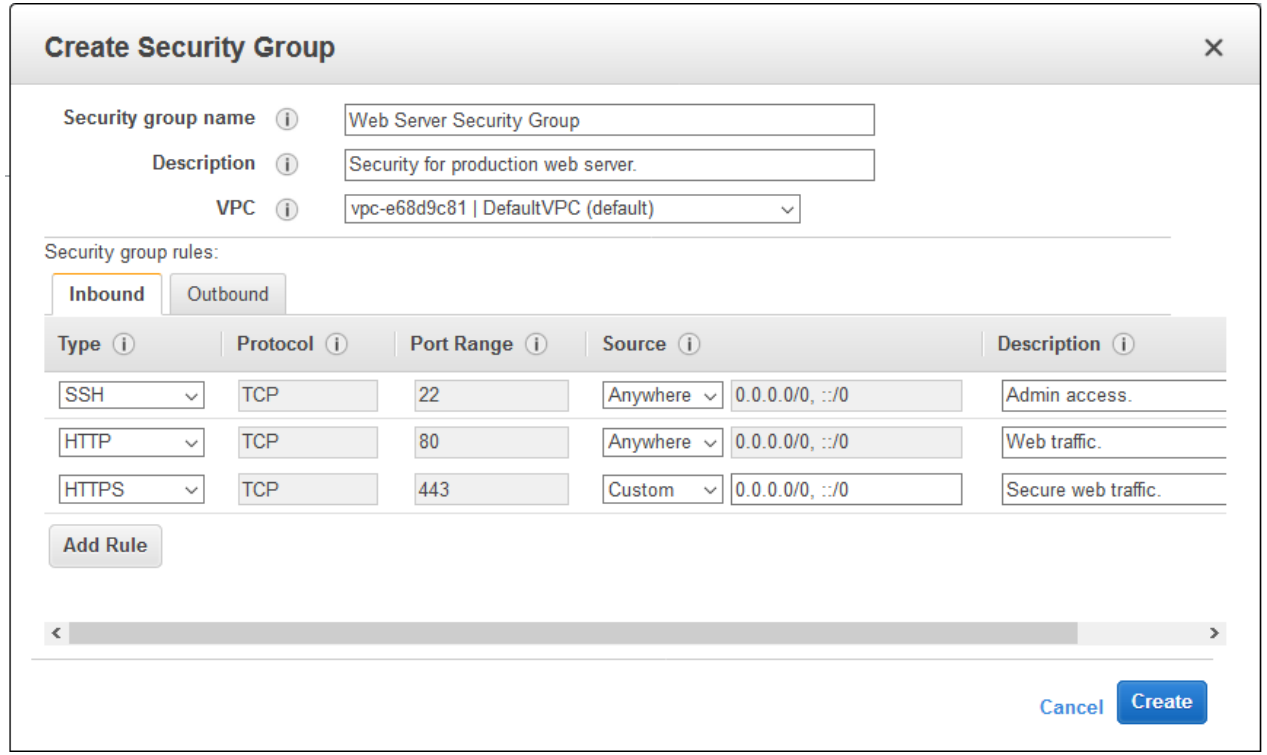
- Control of **computing resources**
  e.g. OS, processor
- Options to optimize compute **costs**
  - On-Demand, Reserved and Spot Instances
  - Dedicated hosts
- Ability to run any type of **workload** e.g.
  - Simple websites
  - Enterprise applications
  - High performance computing (HPC) applications



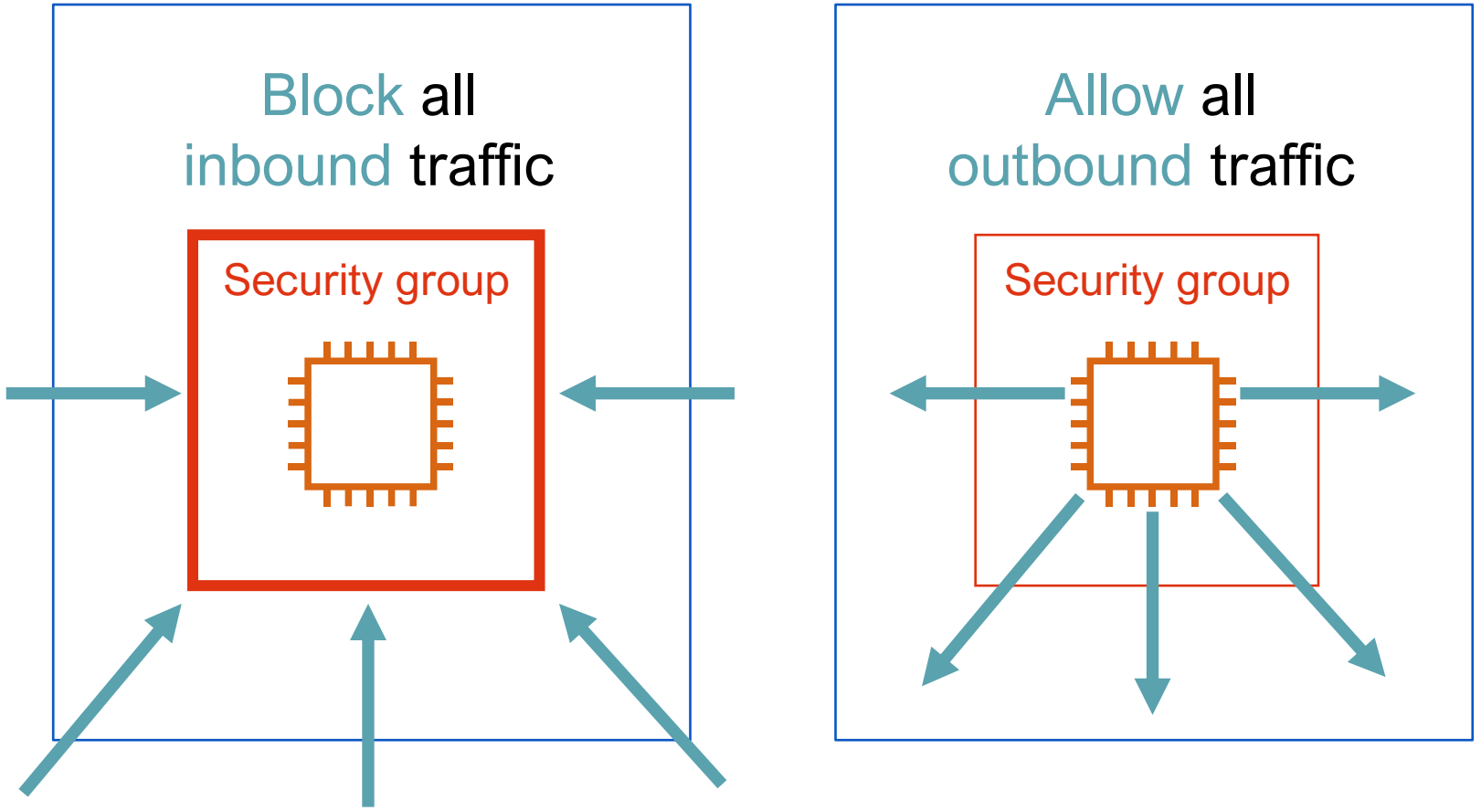
Security Groups

- **Stateful firewalls** control inbound and outbound traffic to AWS resources
- Act at the **level of the instance** or network interface
- Restrict traffic by **protocol**, **port**, src/dst IP address (individual or CIDR block)



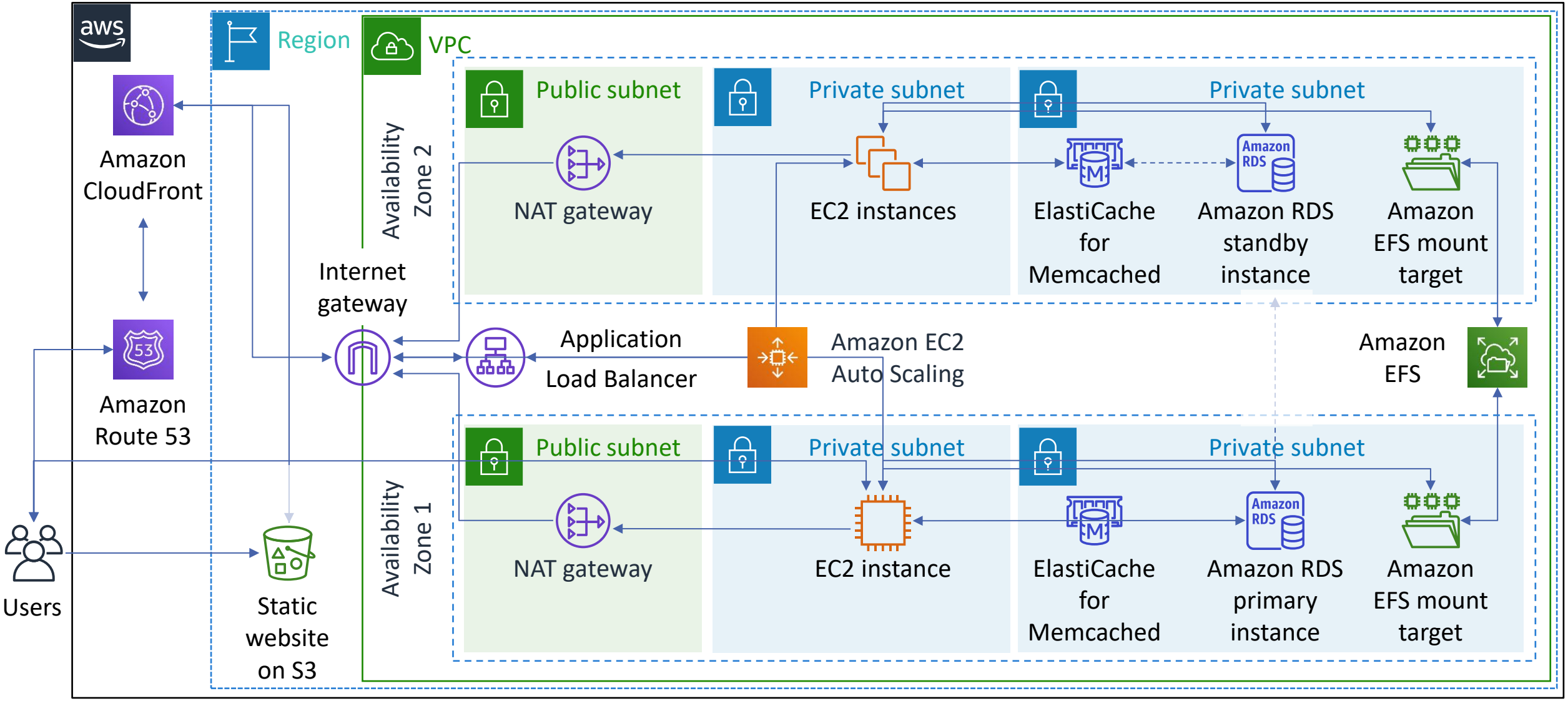


Default Security Group Configuration



When you create a security group, it has no inbound rules, it has an outbound rule that allows all outbound traffic

Virtual Private Cloud


aws
Region
VPC
Amazon CloudFront
Amazon Route 53
Users
Static website on S3
Internet gateway
Availability Zone 2
Availability Zone 1
Public subnet
Private subnet
Private subnet
NAT gateway
EC2 instances
ElastiCache for Memcached
Amazon RDS standby instance
Amazon EFS mount target
Application Load Balancer
Amazon EC2 Auto Scaling
Amazon EFS
Public subnet
Private subnet
Private subnet
NAT gateway
EC2 instance
ElastiCache for Memcached
Amazon RDS primary instance
Amazon EFS mount target

VPC

Provision a **logically isolated** section of the cloud where you can launch resources in a virtual network that you define

Bring your own network

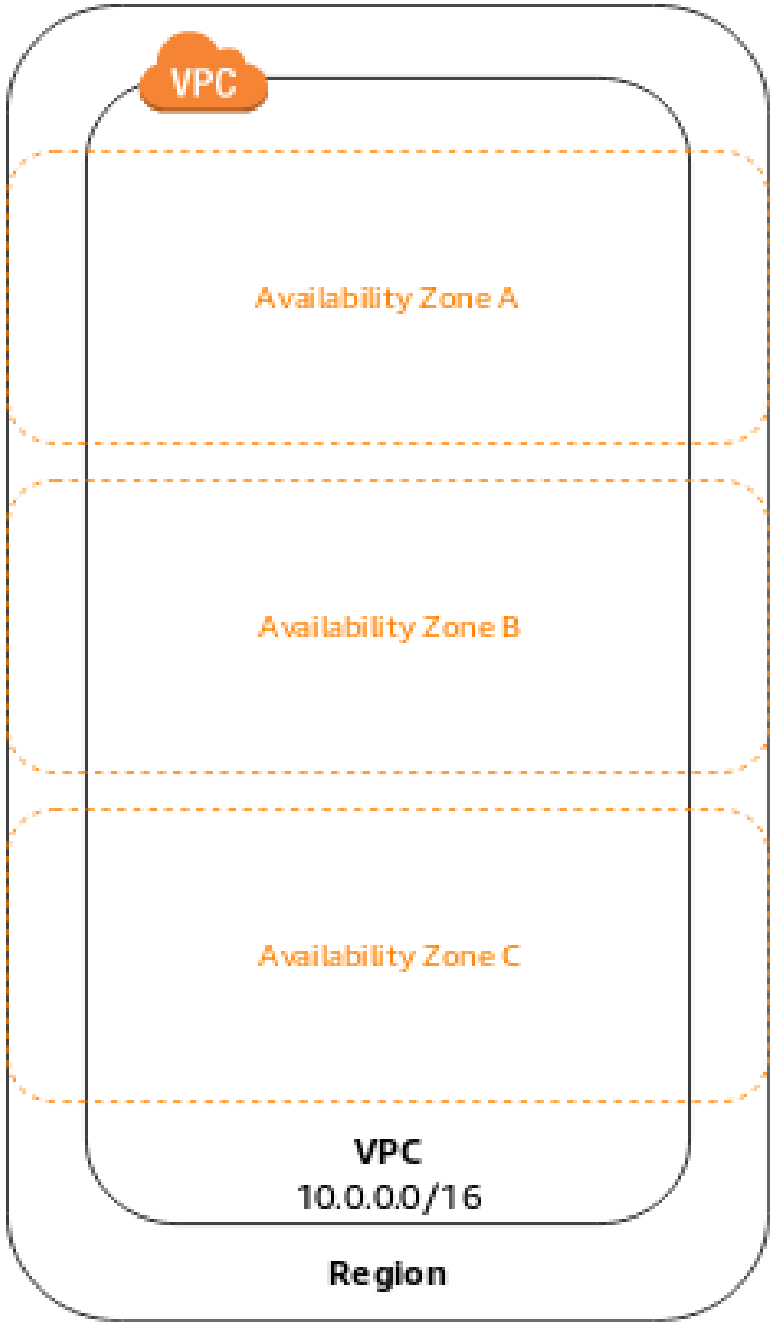
IP Addresses

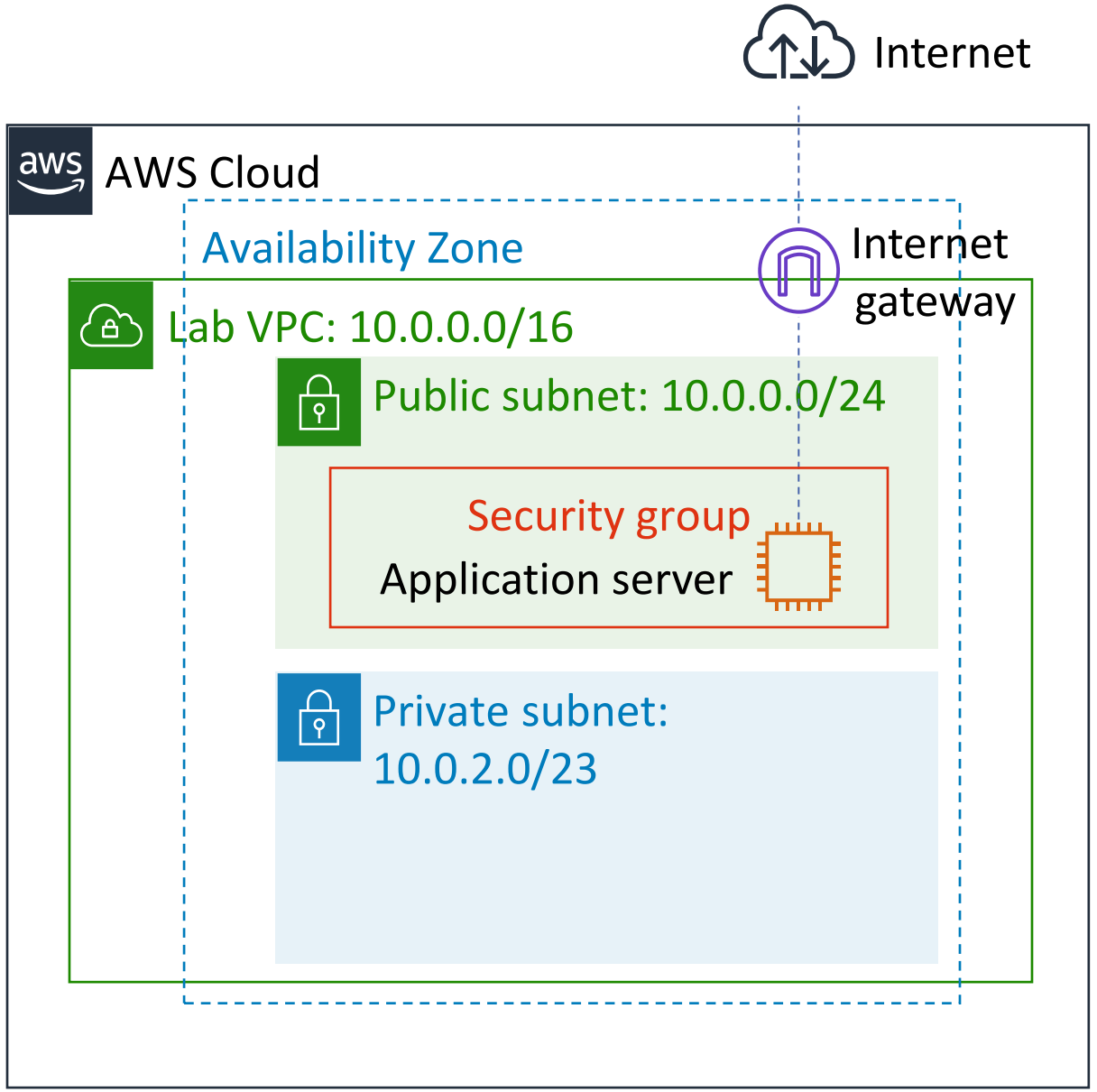
Subnets

Routing rules

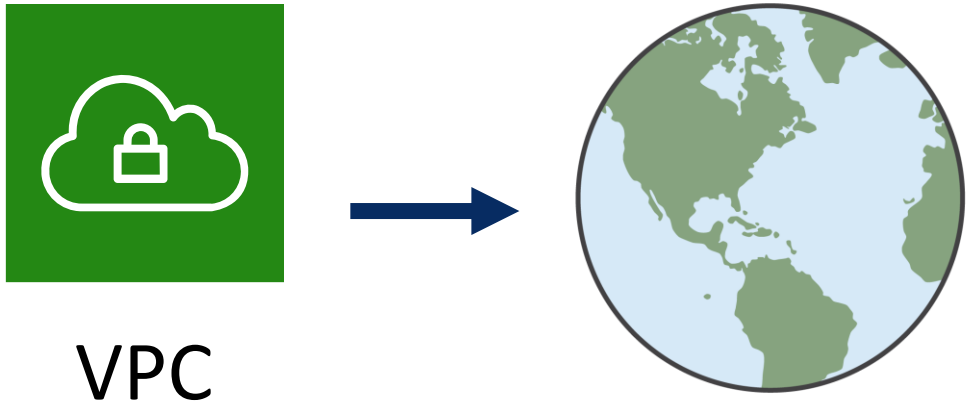
Network configuration

Security rules

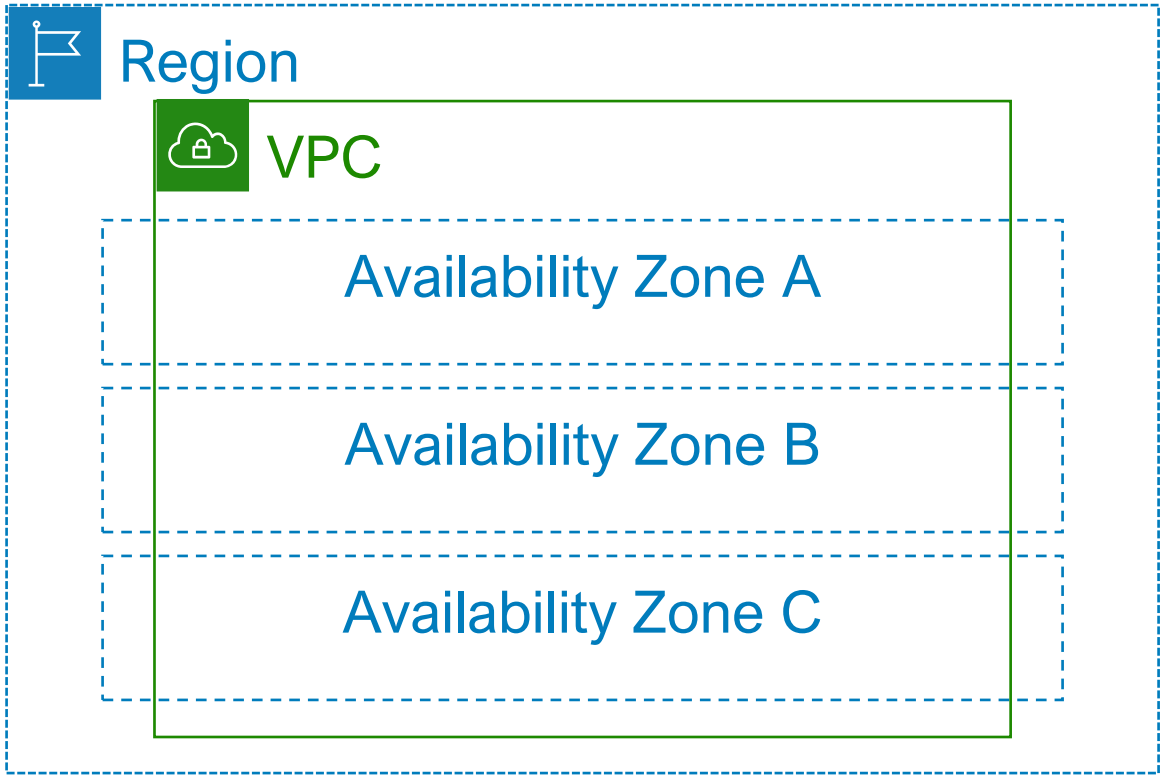

VPC
Availability Zone A
Availability Zone B
Availability Zone C
VPC
10.0.0.0/16
Region


Internet
aws
AWS Cloud
Availability Zone
Internet gateway
Lab VPC: 10.0.0.0/16
Public subnet: 10.0.0.0/24
Security group
Application server
Private subnet:
10.0.2.0/23

VPC Deployment



You can deploy a VPC in any AWS Region

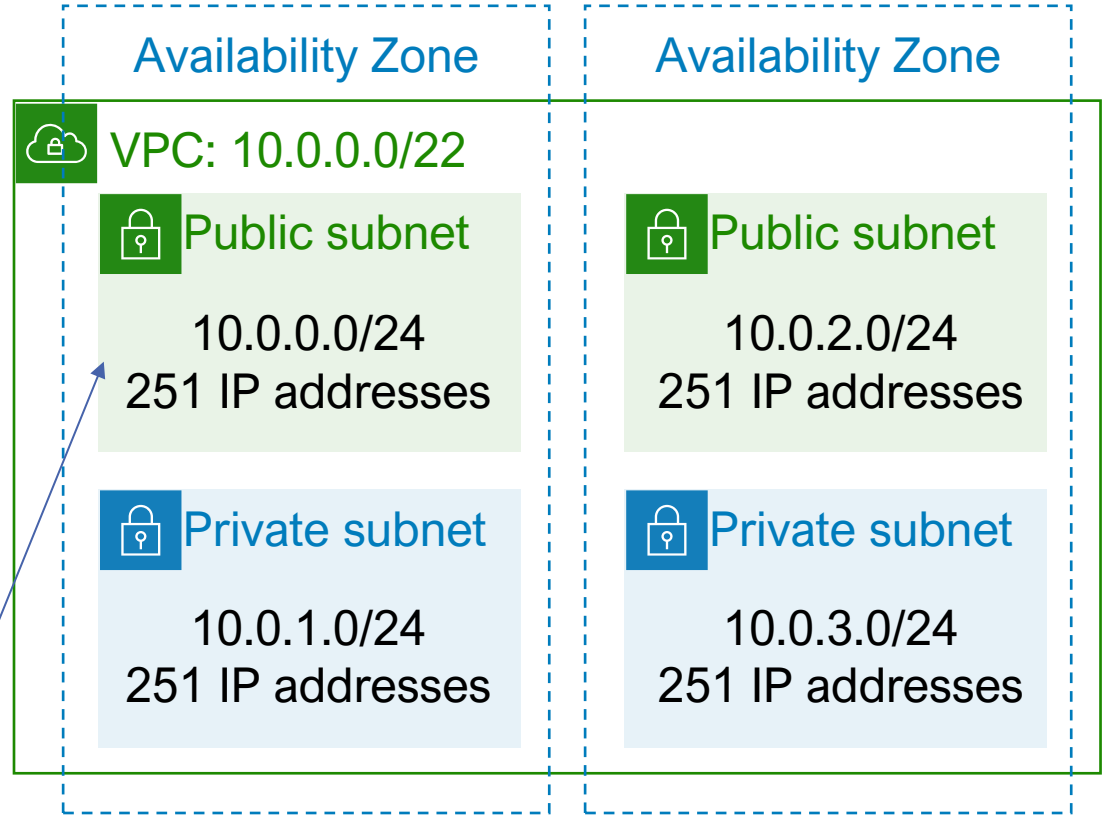


A VPC can host supported resources from any Availability Zone within its Region

VPC Subnets

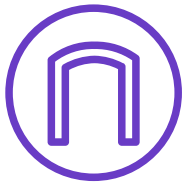
Classless Inter-Domain Routing

- A **segment** of VPC's IP address range to allocate resources
- Subnet CIDR blocks **cannot overlap**
- Each subnet resides within **one** AZ
- You can add **one or more subnets** in each AZ
  - AWS reserves **1st 4** IP addresses and **last** IP address in each subnet
    - 10.0.0.0: Network address
    - 10.0.0.1: VPC local router
    - 10.0.0.2: Domain Name System (DNS) resolution
    - 10.0.0.3: Future use
    - 10.0.0.255: Network broadcast address



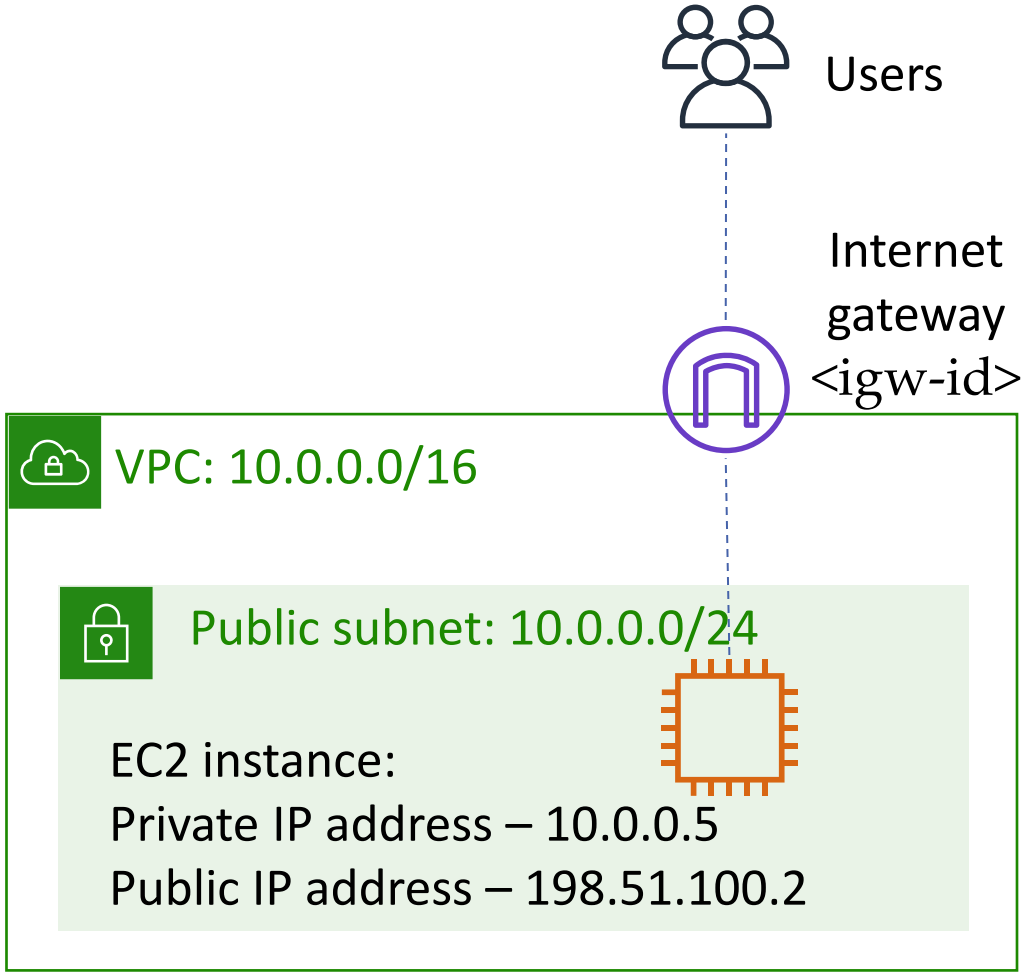
Example: A VPC with CIDR /22 includes 1,024 total IP addresses

Creating a Public Subnet



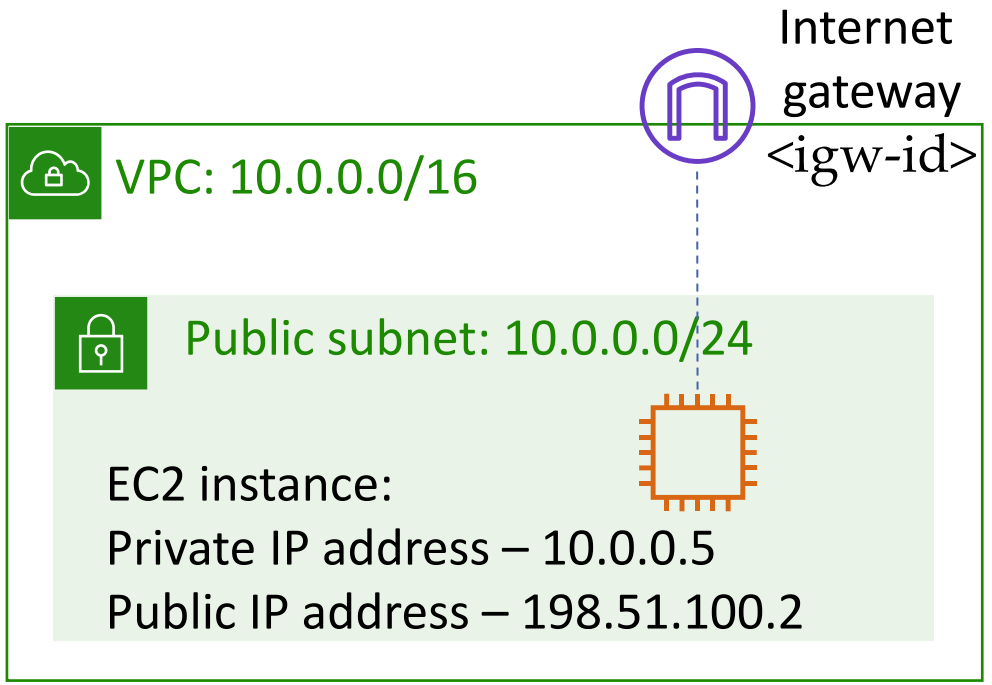
Internet gateway

- Allow communication between resources in VPC and **internet**
- **Horizontally scaled**, redundant, and highly available
- Provide a target in subnet **route tables** for internet-routable traffic



Directing Traffic Between VPC Resources

- **Route tables** are required to direct traffic between VPC resources
- Each VPC has a **main (default) route table**
- Each subnet must be associated with **one route table**
- You can create **custom route tables**
- Best practice: *Use custom route tables for each subnet*

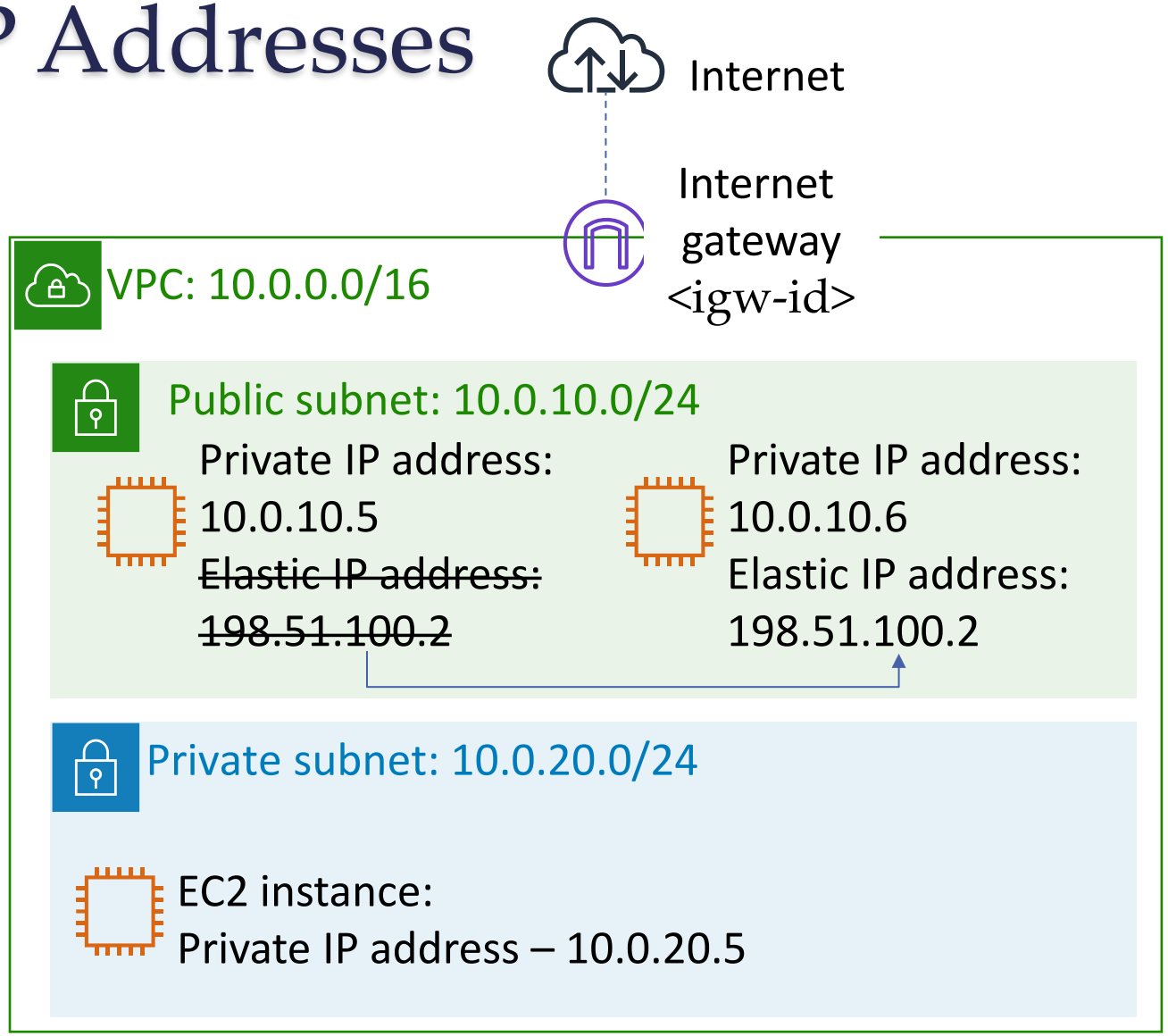


Public route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Elastic IP Addresses

- **Static public IPv4 addresses** associated with AWS account
- Can be associated with an **instance** or **network interface**
- Can be **remapped** to another instance in your account
- Useful for **redundancy** when load balancers are not an option



Connecting Private Subnets to the Internet

NAT gateways

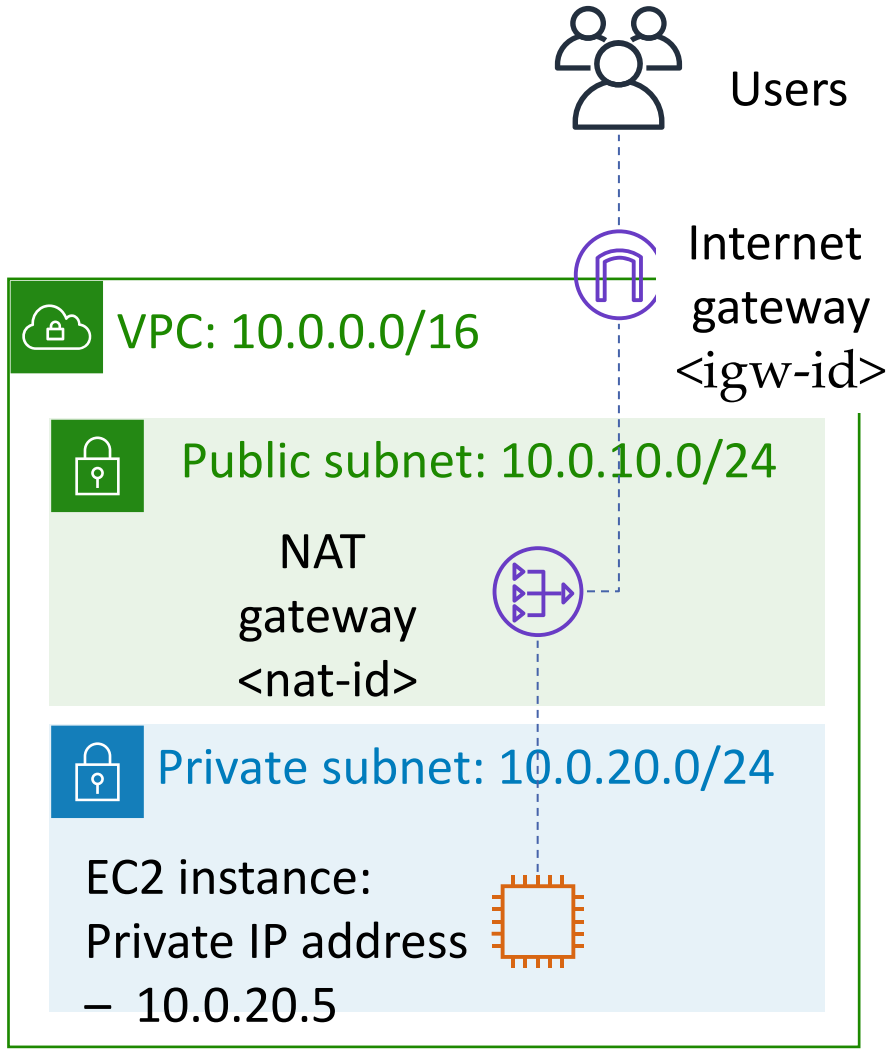
- Enable instances in a private subnet to initiate **outbound traffic** to internet or other AWS services
- Prevent private instances from receiving **inbound connection requests** from internet

Public route table

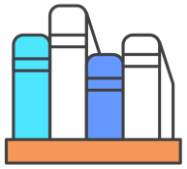
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Private route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<nat-id>



Example Subnet Use Case



Data store instances → **Private** subnet

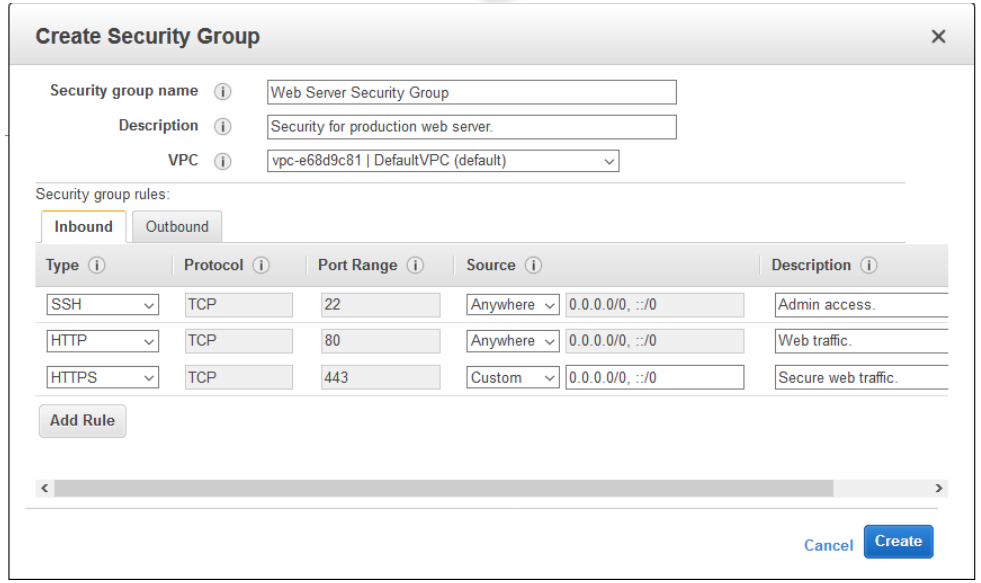
Batch-processing instances → **Private** subnet

Backend instances → **Private** subnet

Web application instances → **Public or private** subnet

behind load balancer

Securing VPCs



Security Groups

- **Stateful firewalls** control inbound and outbound traffic to AWS resources
- Act at the **level of the instance** or network interface
- Restrict traffic by **protocol**, **port**, and src/dst IP address (individual or CIDR block)

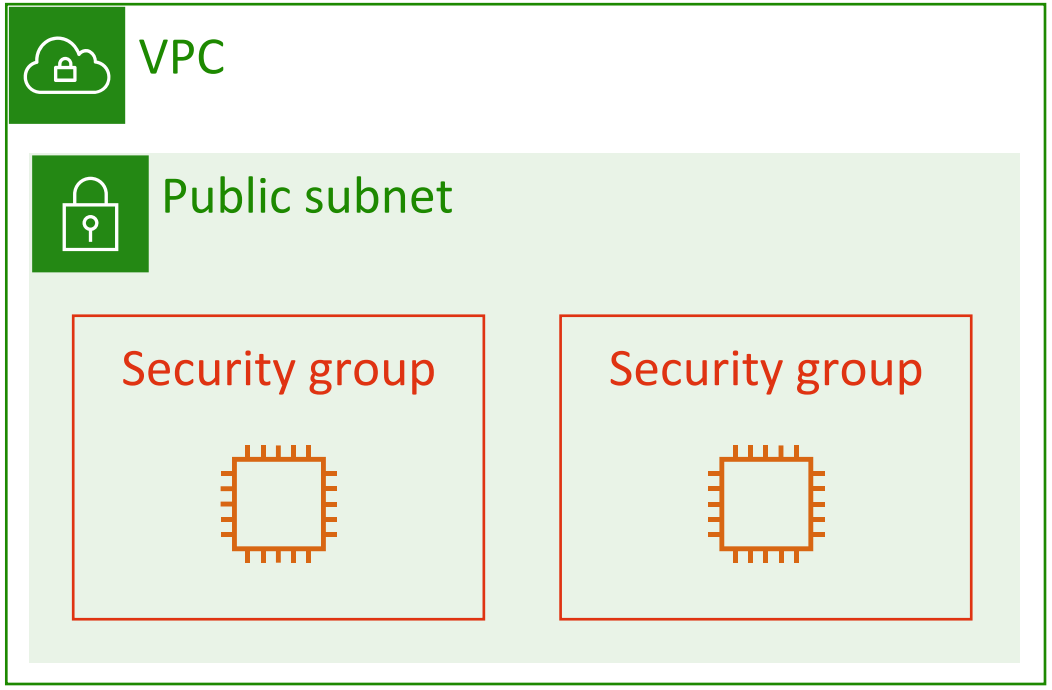
default security group

Inbound rule

Source	Protocol	Port range	Description
The security group ID (its own resource ID)	All	All	Allows inbound traffic from network interfaces and instances that are assigned to the same security group.

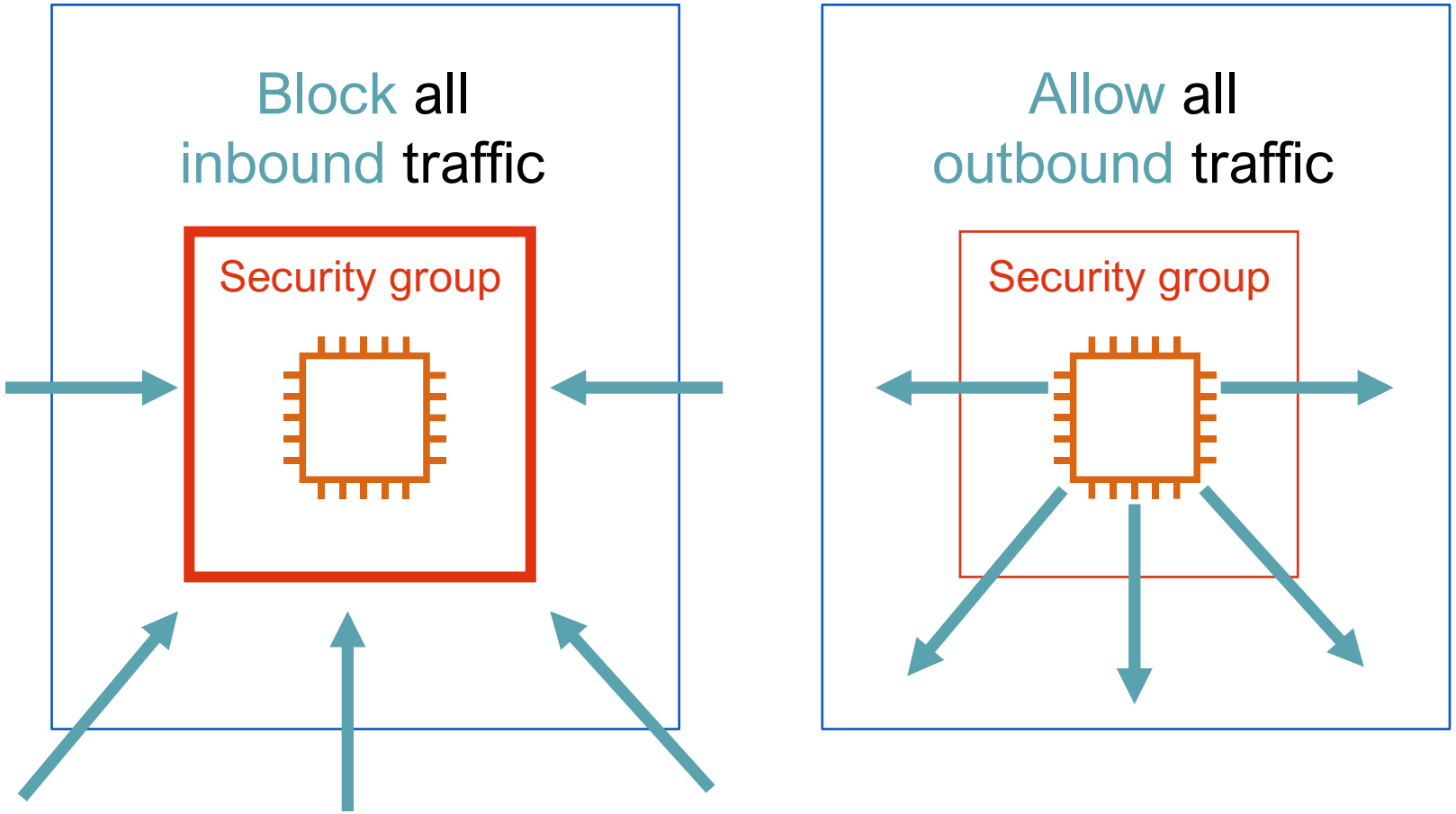
Outbound rules

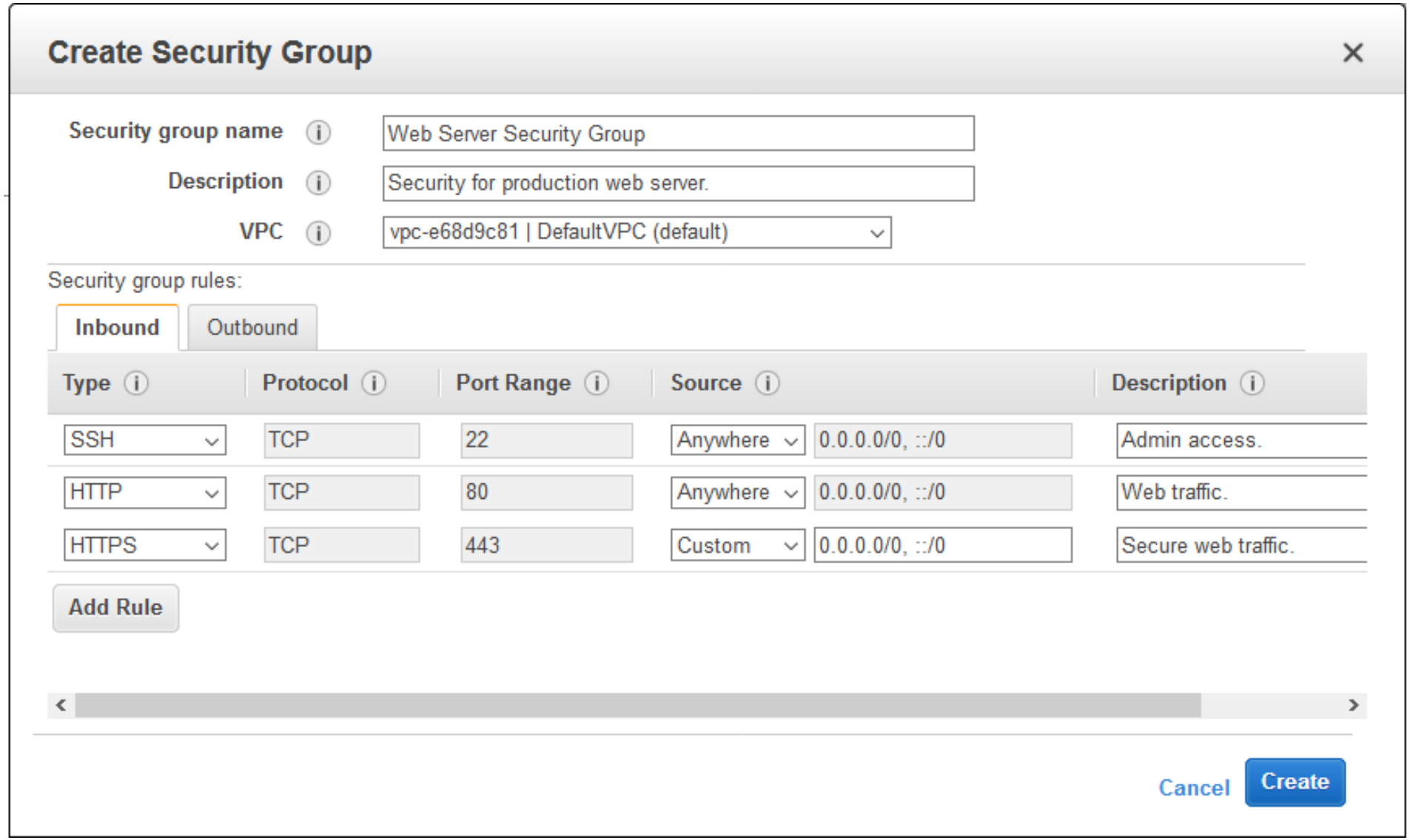
Destination	Protocol	Port range	Description
0.0.0.0/0	All	All	Allows all outbound IPv4 traffic.
::/0	All	All	Allows all outbound IPv6 traffic. This rule is added only if your VPC has an associated IPv6 CIDR block.



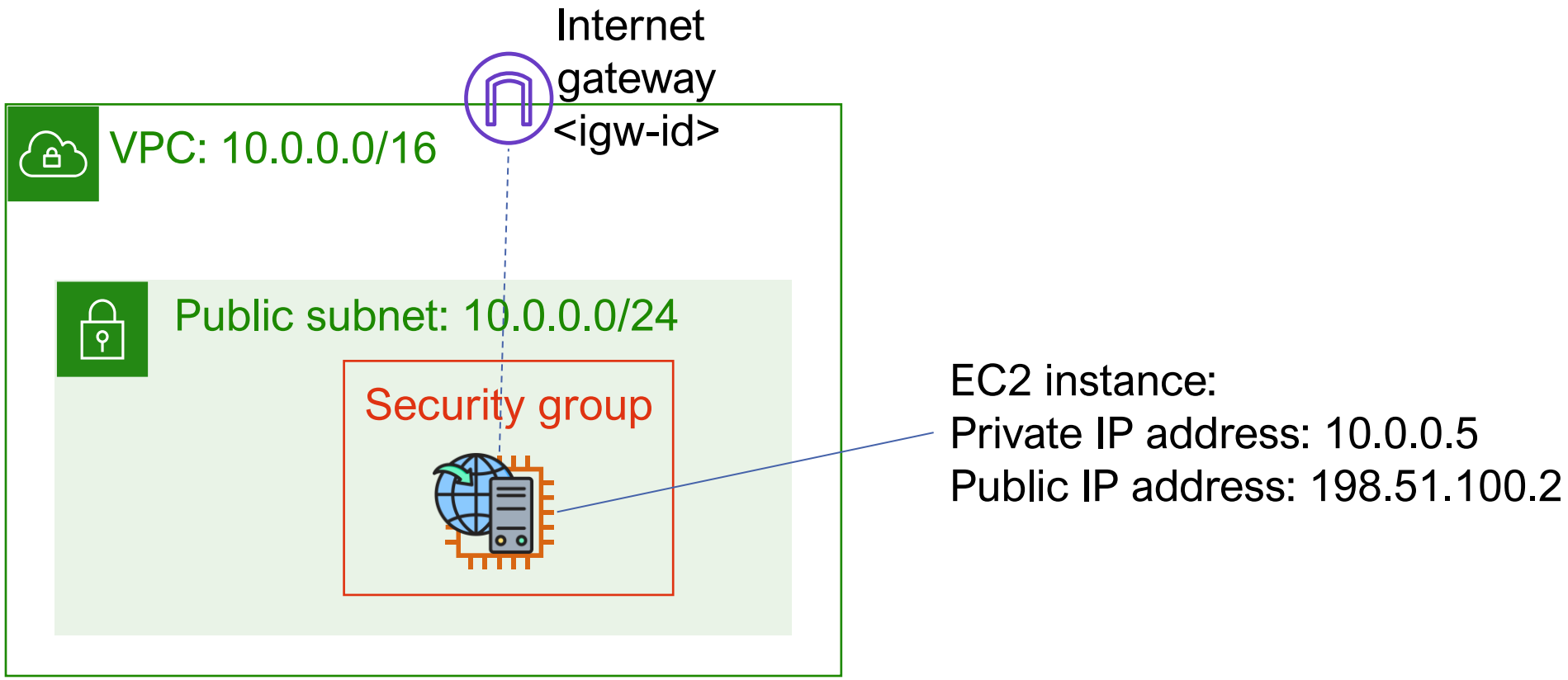
Create a Security Group

Default Security Group Configuration



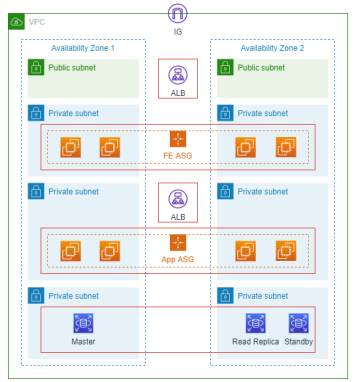

Create Security Group
Security group name
Web Server Security Group
Description
Security for production web server.
VPC
vpc-e68d9c81 | DefaultVPC (default)
Security group rules:
Inbound
Outbound
Type
Protocol
Port Range
Source
Description
SSH
TCP
22
Anywhere
0.0.0.0/0, ::/0
Admin access.
HTTP
TCP
80
Anywhere
0.0.0.0/0, ::/0
Web traffic.
HTTPS
TCP
443
Custom
0.0.0.0/0, ::/0
Secure web traffic.
Add Rule
Cancel
Create

Custom Security Groups



Inbound				
Type	Protocol	Port Range	Source	Destination
HTTP	TCP	80	Anywhere	Allow web access

Chaining Security Groups



Inbound rules

Allow: HTTP (port 80) or HTTPS (port 443)
Source: 0.0.0.0/0 (any)

Allow: SSH (port 22) to Web tier
Source: Corporate IP range

Inbound rule
Allow: SSH (port 22) to Application tier
Source: Corporate IP range

Inbound rule
Allow: SSH (port 22) to Database tier
Source: Corporate IP range

Web tier security group

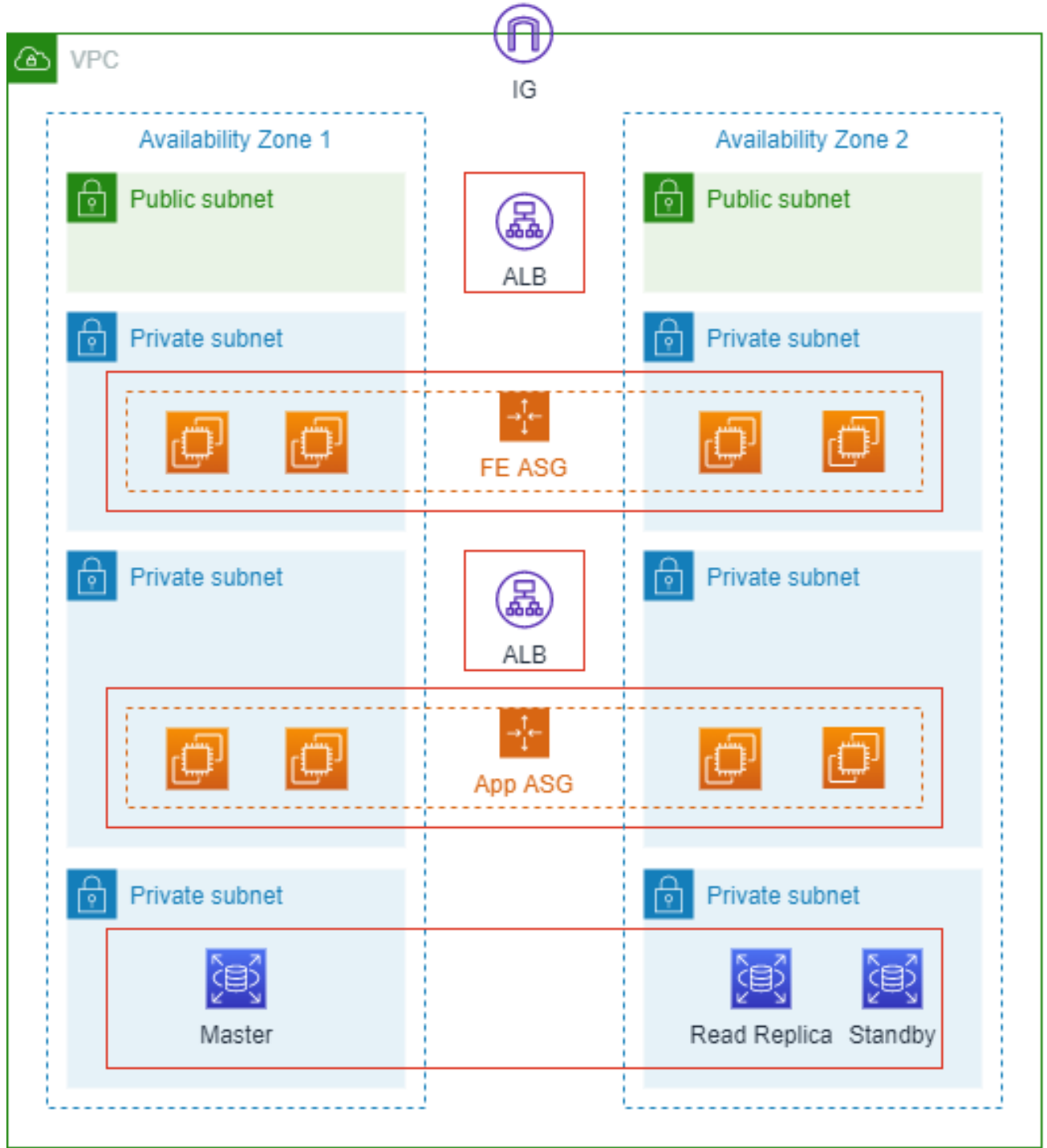
Application tier security group

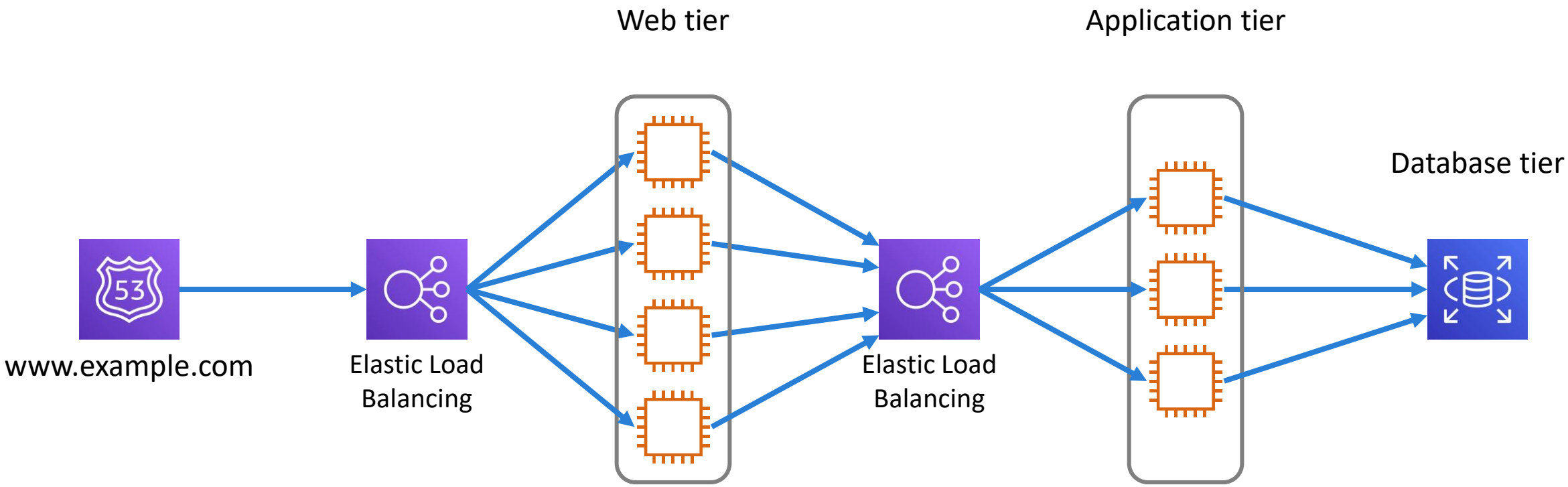
Database tier security group

Inbound rule
Allow: HTTP port 8000 (application specific)
Source: Web tier

Inbound rule
Allow: TCP port 3306
Source: Application tier

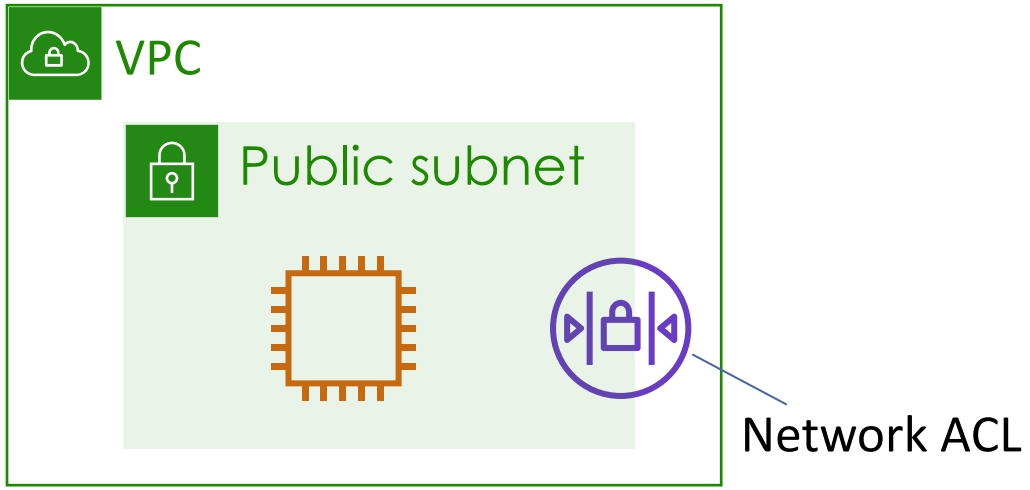
All other ports blocked by default


VPC
IG
Availability Zone 1
Availability Zone 2
Public subnet
Public subnet
ALB
Private subnet
Private subnet
FE ASG
Private subnet
Private subnet
ALB
App ASG
Private subnet
Private subnet
Master
Read Replica
Standby

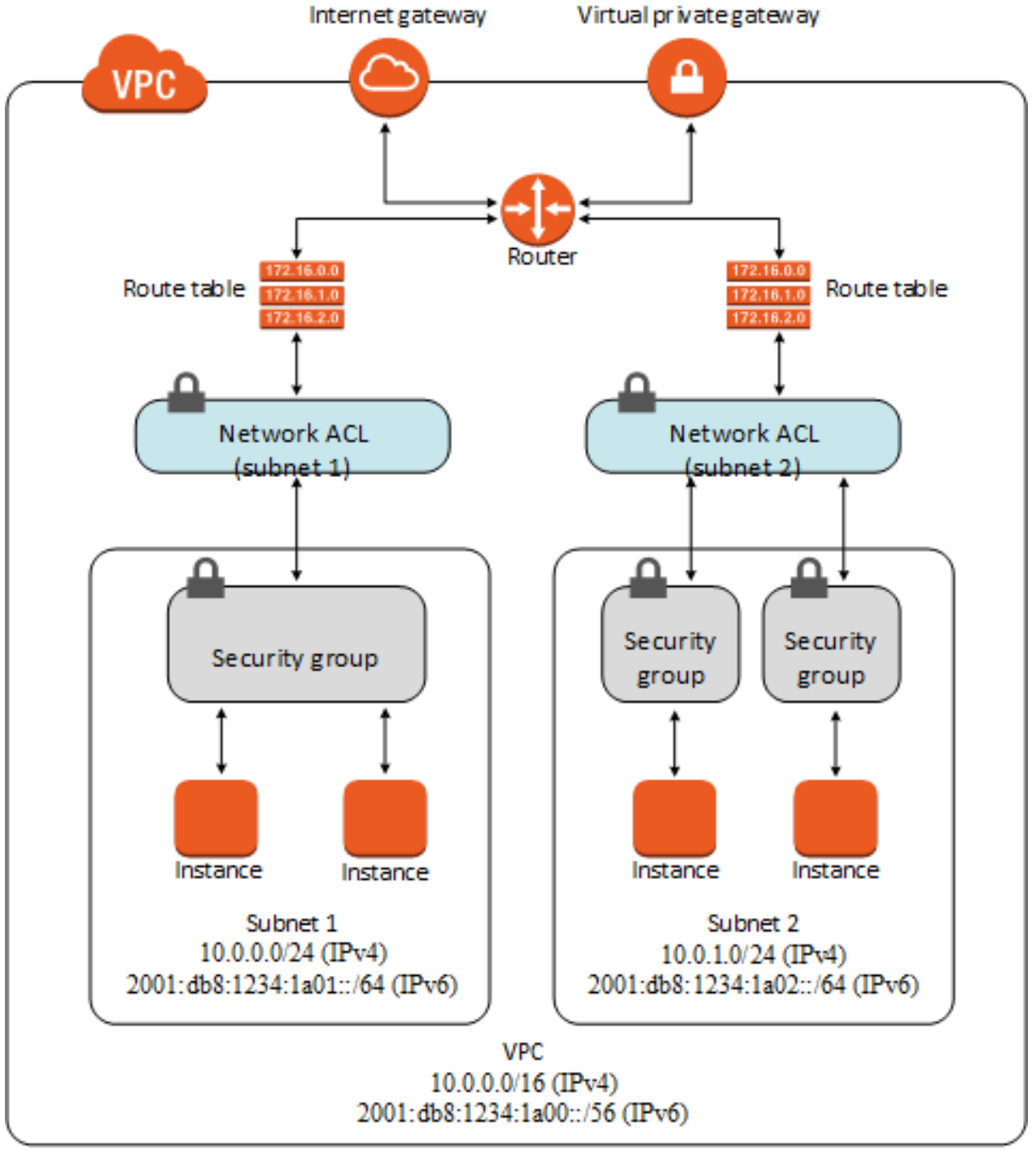

Web tier
Application tier
Database tier
www.example.com
Elastic Load Balancing
Elastic Load Balancing

Network ACLs

- Act at the subnet level
- Allow all inbound and outbound traffic by default
- **Stateless**
  - require explicit rules for both inbound and outbound traffic



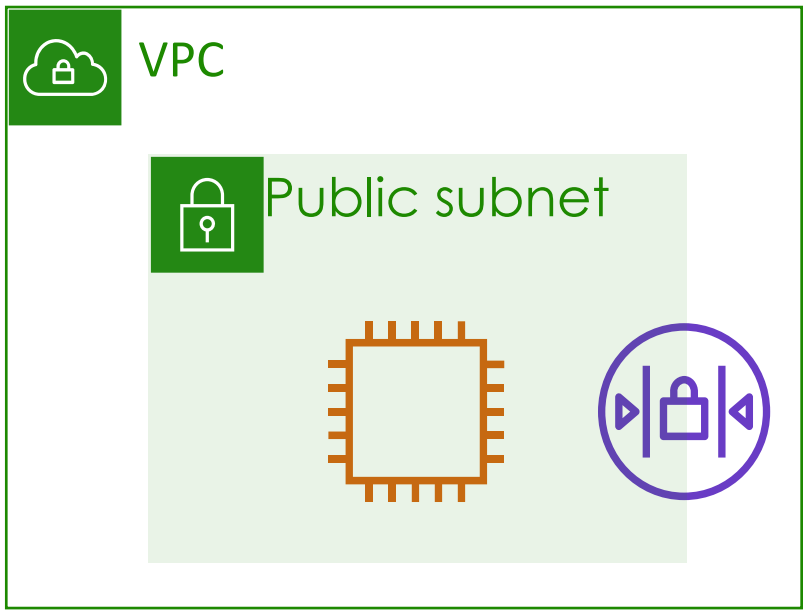
ACLs
vs
Security Groups



Security group	Network ACL
Operates at the instance level	Operates at the subnet level
Allow rules only	Allow rules and deny rules
Stateful: Return traffic is automatically allowed	Stateless: Return traffic must be explicitly allowed
Evaluates all rules before deciding whether to allow traffic	Processes rules in order, starting with the lowest numbered rule, when deciding whether to allow traffic
Applies to an instance only if someone specifies the SG when launching the instance, or associates the SG with the instance later on	Automatically applies to all instances in the subnets that it's associated with Additional layer of defense

Custom network ACLs

- Recommended for **specific** network security requirements
- By default, each custom network ACL **denies all** inbound and outbound traffic until you **add rules**

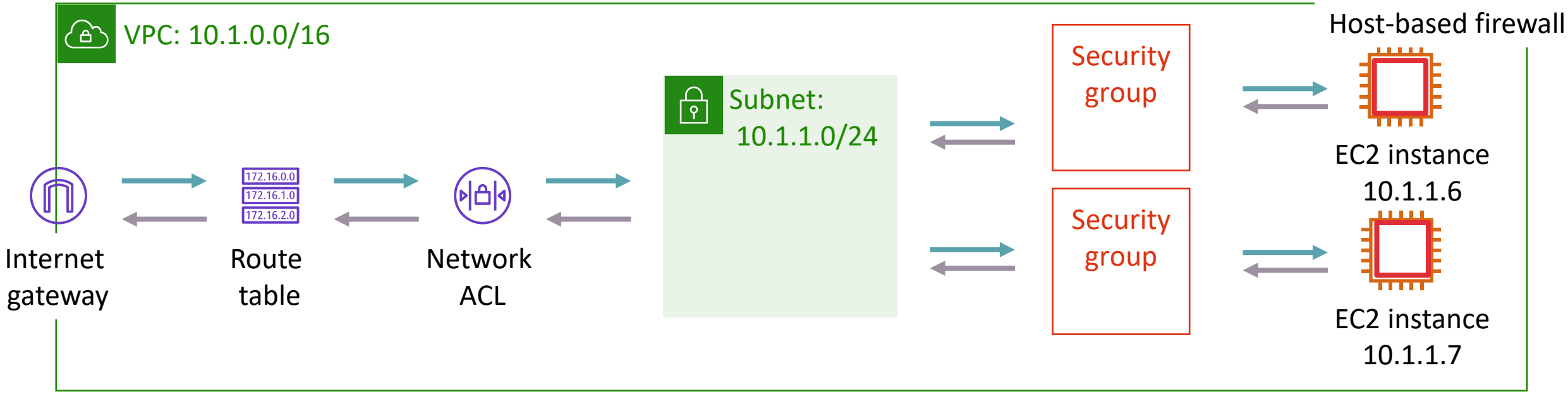


Nacl-11223344

Inbound:
Rules # 100: SSH 172.31.1.2/32 ALLOW
Rules # *: ALL traffic 0.0.0.0/0 DENY

Outbound:
Rules # 100: Custom TCP 172.31.1.2/31 ALLOW
Rules # *: All traffic 0.0.0.0/0 DENY

Multiple Layers of Defense



Public Subnet

To create a public subnet to allow communication between instances in your VPC and the internet:

Attach an internet gateway to your VPC

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Point subnet's route table to the internet gateway

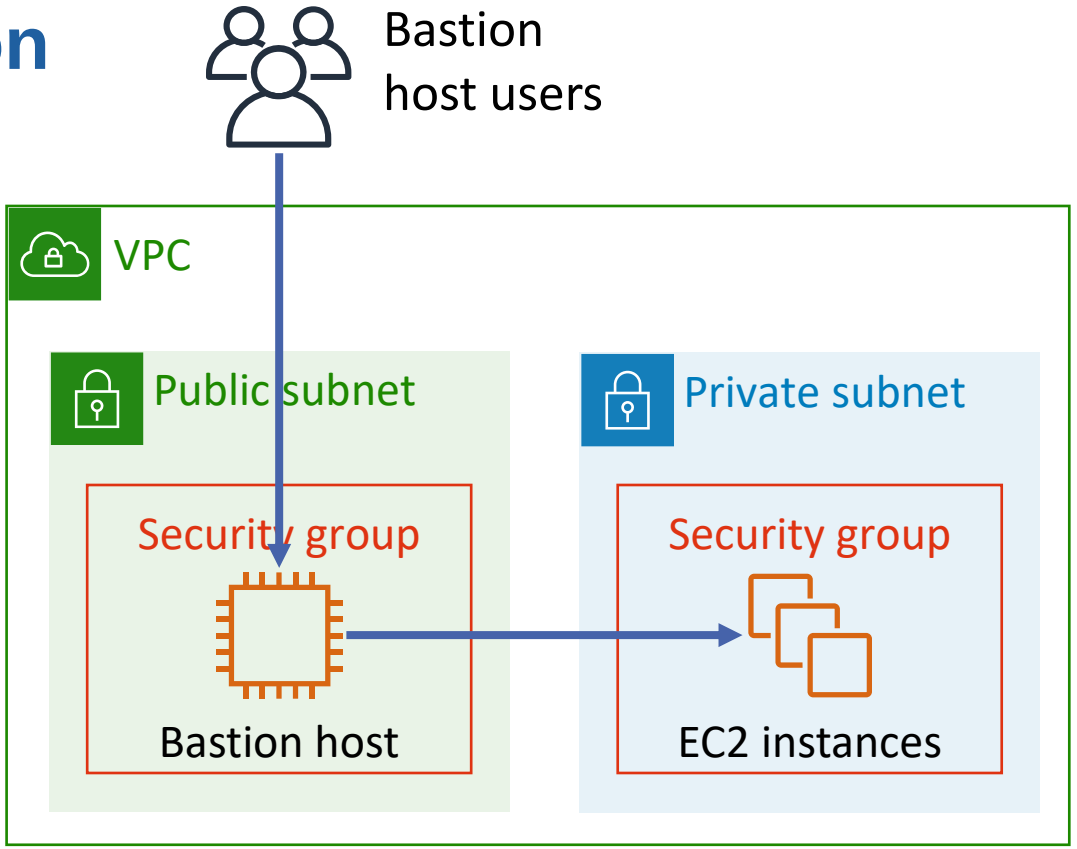
Make sure that instances have public IP or Elastic IP addresses

Security group

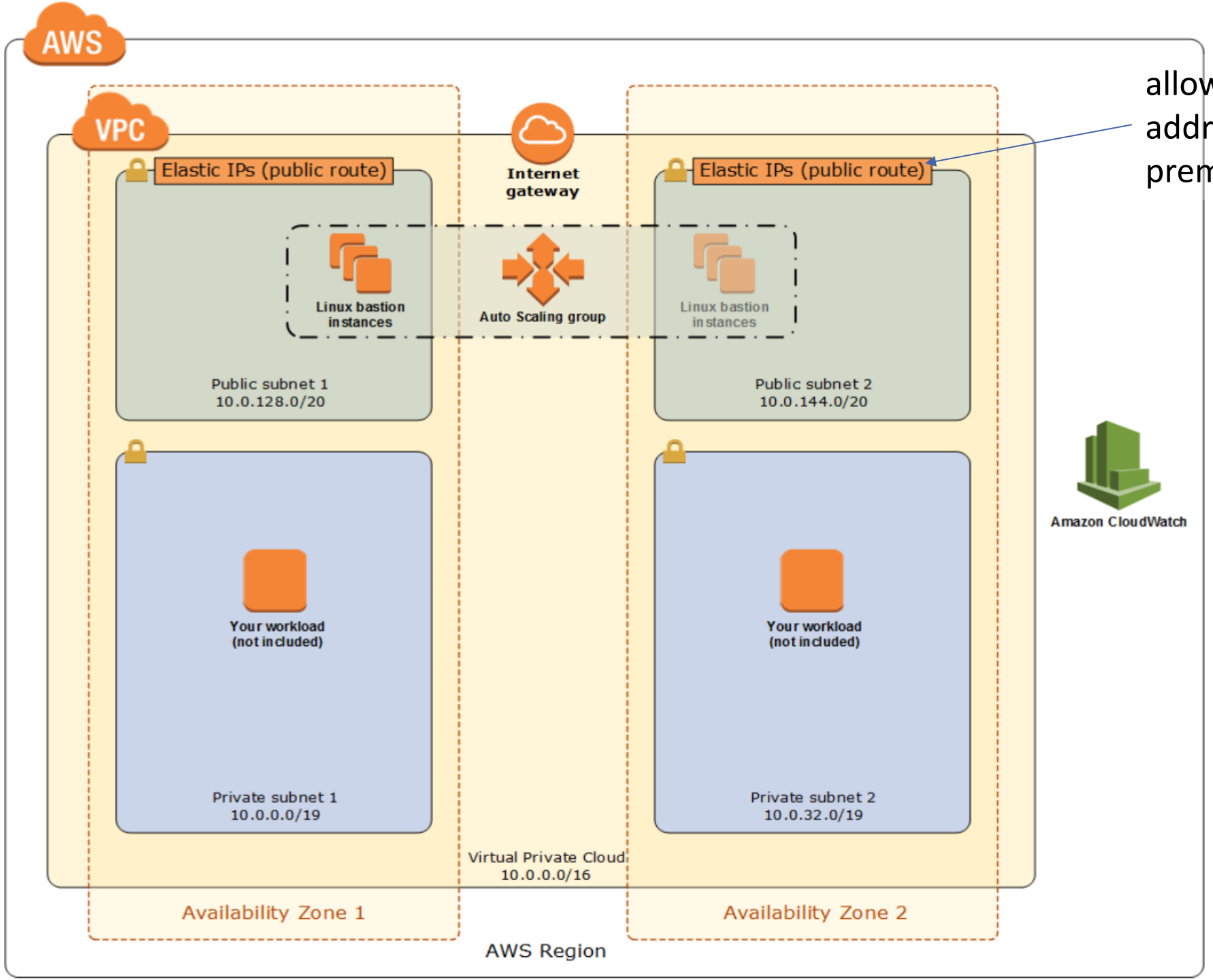
Make sure that security groups and network ACLs allow relevant traffic to flow

Bastion Hosts

- **Server** used to provide access to private network from outside
- Minimize the chances of **penetration**
- Instances in private subnet are in **security group** that allows **SSH** from security group attached to **bastion host**
- For added security, sets up **CloudWatch Logs** for shell history logs



Amazon CloudWatch



allow the elastic IP addresses on on-premises firewalls

Cloud Security Monitoring

AWS CloudTrail

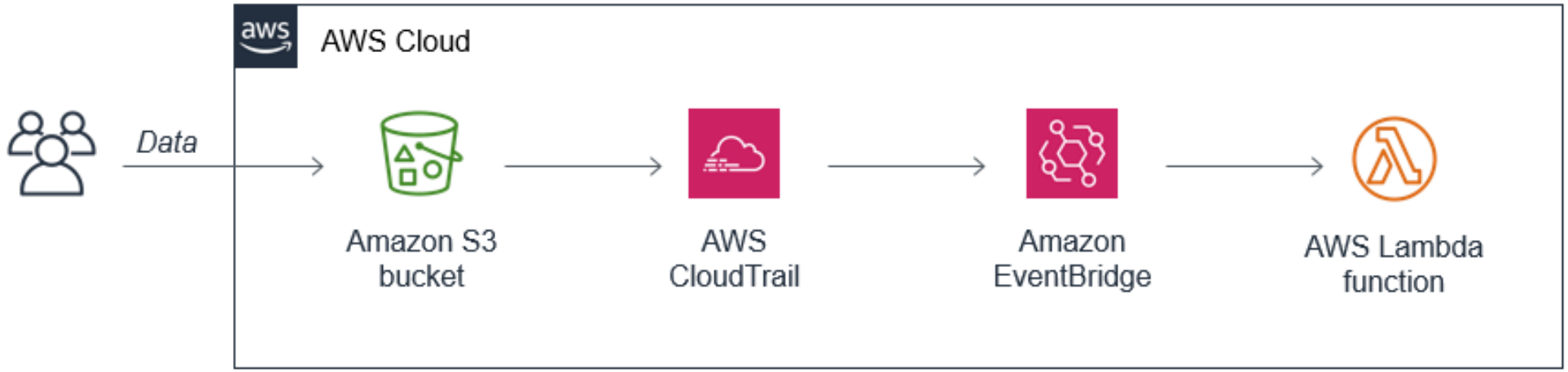
- **Logs** and **monitors** user activity
- Provides **event history** of AWS account
  - Actions taken through the **Management Console, SDKs, CLI**
  - Increases visibility into **user** and **resource activity**
  - **90-day** event history by default, at no cost
- Identify
  - **Who** accessed account
  - **When** and from **where**
  - **What** action they took on an AWS service

Uses of CloudTrail

- Perform **security analysis**
- Track **resource changes**
- Operational **troubleshooting**

e.g. discover which calls were **blocked** by IAM policies

- Meet **compliance** and **auditing** requirements
- **Automatically** respond to **security threats**

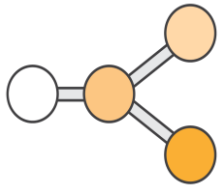


Amazon CloudWatch

- Collects and tracks **metrics** for resources and applications
- Allows responding to **performance changes**, **optimizing** resource utilization, **unified view** of operational health
- Helps **correlate**, **visualize**, and **analyze** metrics and logs
  - Built-in and custom metrics
- Enables creating **alarms** and detect **anomalous behavior**
  - Send **notifications** or make **changes** to monitored resources

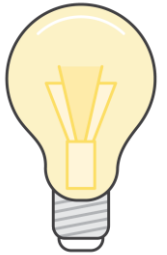
How CloudWatch Responds

Metrics



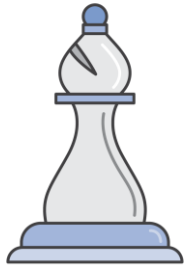
Logs

Alarms



Events

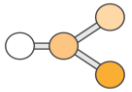
Rules



Targets

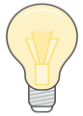
CloudWatch Metrics

Metrics



Logs

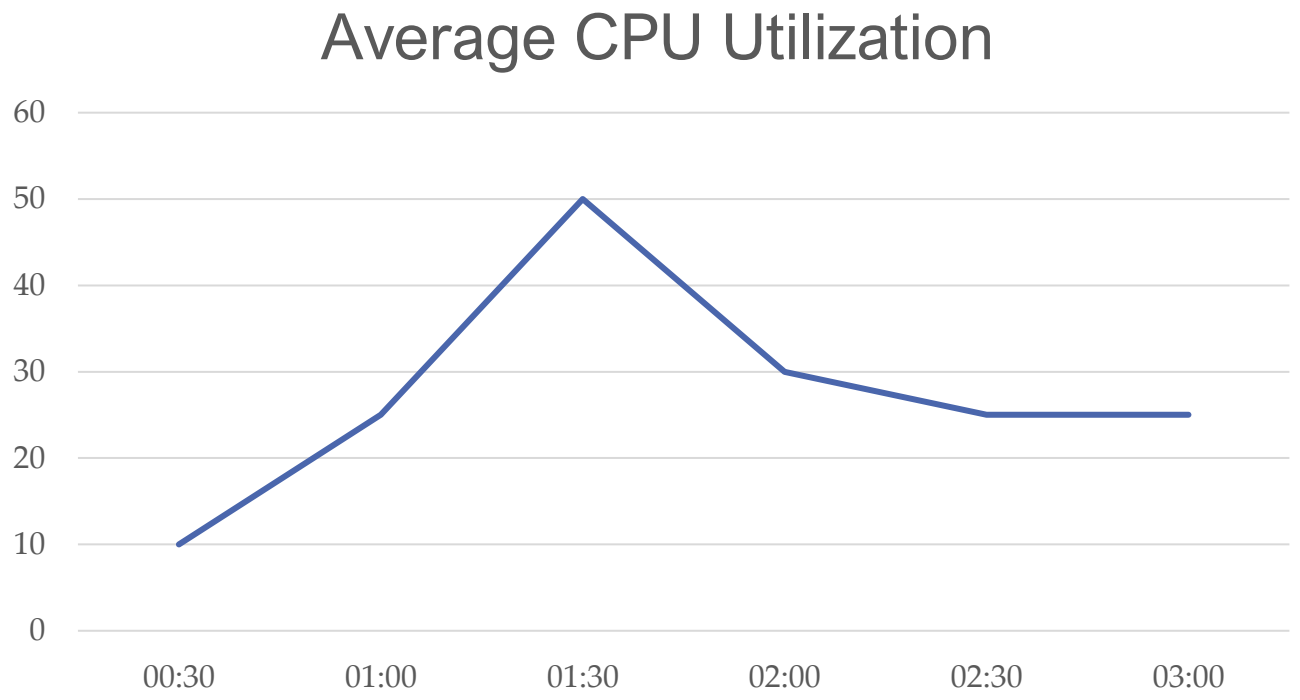
Alarms



Events

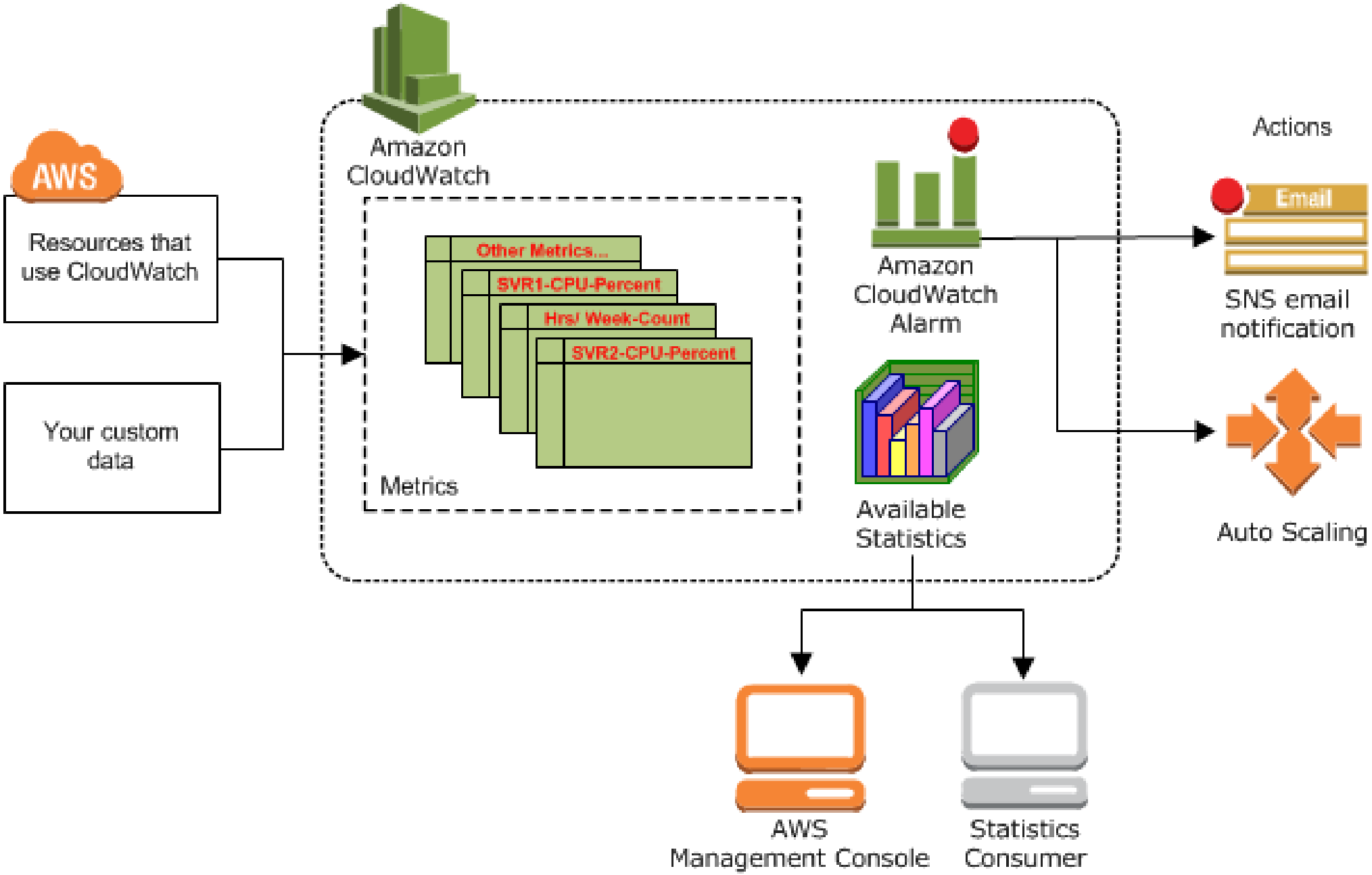
Rules

Targets



Metric data is kept for 15 months

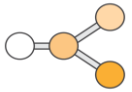
By default, many AWS services provide **metrics for resources**, e.g. EC2 instances, EBS volumes, RDS DB instances


Amazon CloudWatch
AWS
Resources that use CloudWatch
Your custom data
Other Metrics...
SVR1-CPU-Percent
Hrs/ Week-Count
SVR2-CPU-Percent
Metrics
Amazon CloudWatch Alarm
Available Statistics
Actions
Email
SNS email notification
Auto Scaling
AWS Management Console
Statistics Consumer

Amazon CloudWatch Logs

Monitor, store and access your log files


Metrics

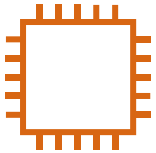

Logs


Alarms


Events


Rules


Targets


Application

Log_File.txt
Errors: 3
Warnings: 12
Connections: 20
Print out…


Amazon CloudWatch

e.g. send a notification when rate of error exceeds a **threshold** you specify

CloudWatch Logs **Insights** to **query** logs (identifies log fields)

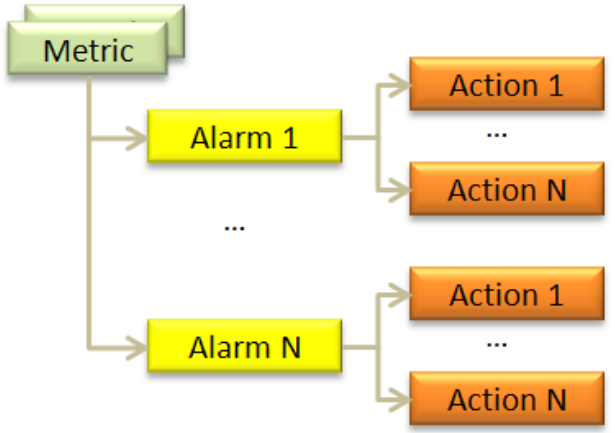

Amazon S3

Source examples

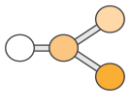
- **VPC Flow Logs**
- CloudTrail
- Route 53
- Elastic Load Balancing

CloudWatch Alarms

Use to automatically initiate actions

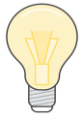


Metrics



Logs

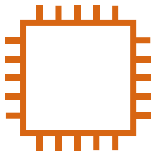
Alarms



Events

Rules

Targets



Application

CPUUtilization metric

80% **60%** **45%** **25%** **10%** **10%** **10%** **10%** **5%**

Alarm

If CPUUtilization metric is > 50% for 5 minutes

Trigger an action e.g.

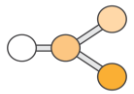
- Send a notification to the development team
- Create another instance to handle the load

amazon SNS

AWS ASGs

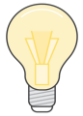
Amazon EventBridge Events

Metrics



Logs

Alarms

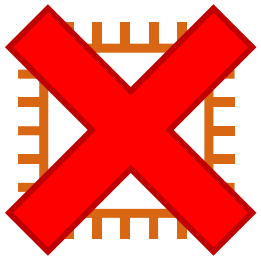


Events

Rules

Targets

Event: EC2 instance termination



Amazon EventBridge

Event examples

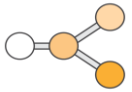
- Change in AWS resource e.g.
  - Console sign-in
  - EC2 instance state change
  - EC2 Auto Scaling state change
  - EBS volume creation
- AWS API call **(CloudTrail)**
- Events from customer applications

- Ingests a stream of real-time data from applications and AWS services, **routes** it to **targets** e.g. AWS Lambda

Amazon EventBridge Rules

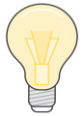
Rule matches incoming events and routes them to targets for processing

Metrics



Logs

Alarms

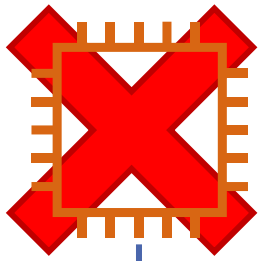


Events

Rules

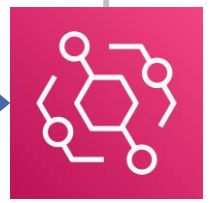
Targets

Event



Rule example

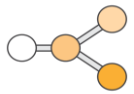
```
{
 "source": [
   "aws.ec2"
],
 "detail-type": [
   "EC2 Instance State-change Notification”
],
 "detail": {
   "state": [
      “terminated" ]
 }
}
```



Amazon EventBridge

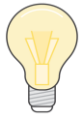
Amazon EventBridge Targets

Metrics



Logs

Alarms

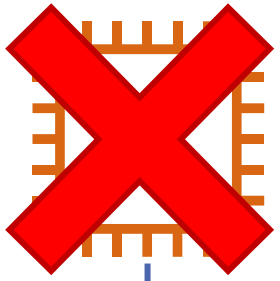


Events

Rules

Targets

Event



Rule example

```
{
 "source": [ "aws.ec2" ],
 "detail-type": [ "EC2 Instance
State-change Notification" ],
 "detail": {
 "state": [ “terminated" ]
 }
}
```

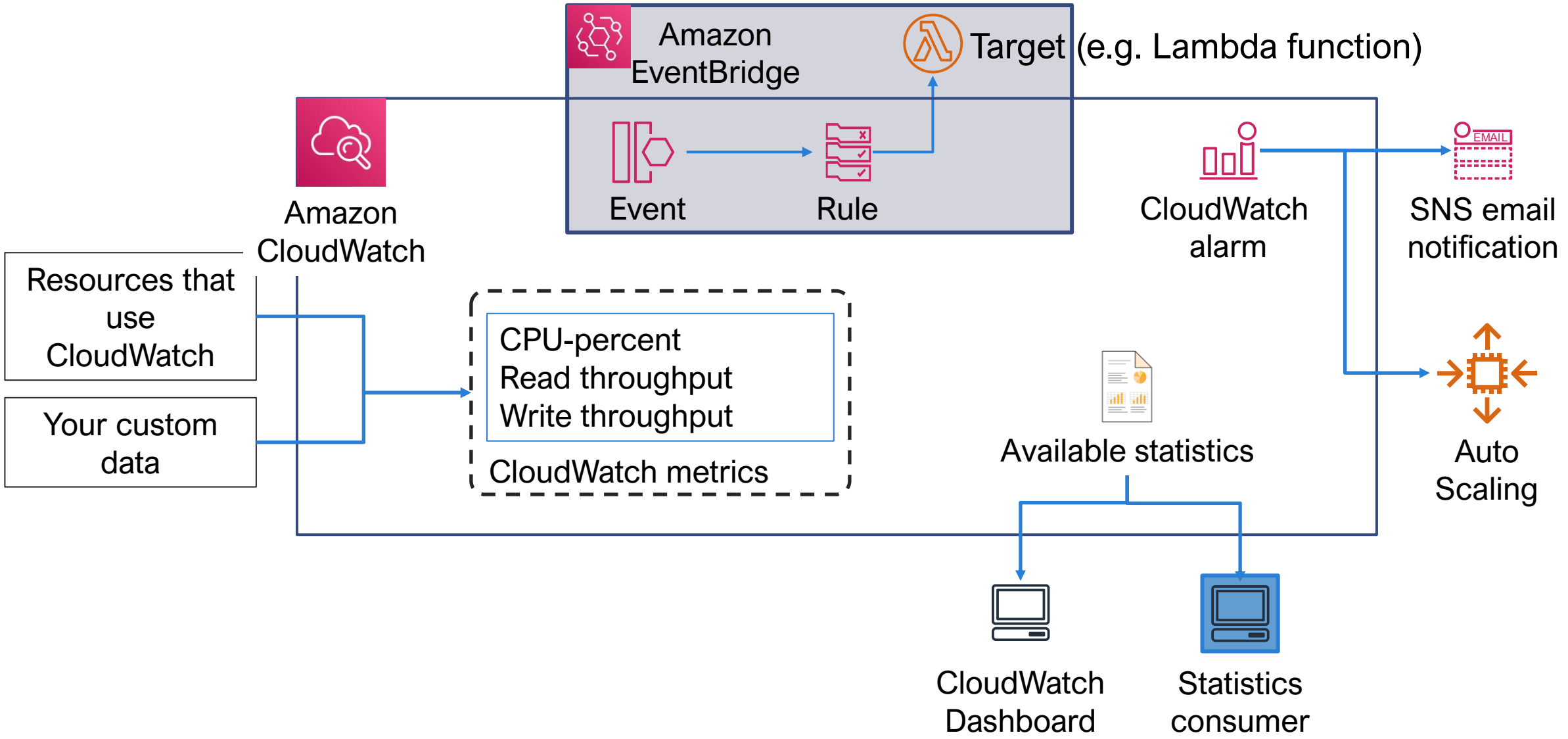
A target processes events

Target examples

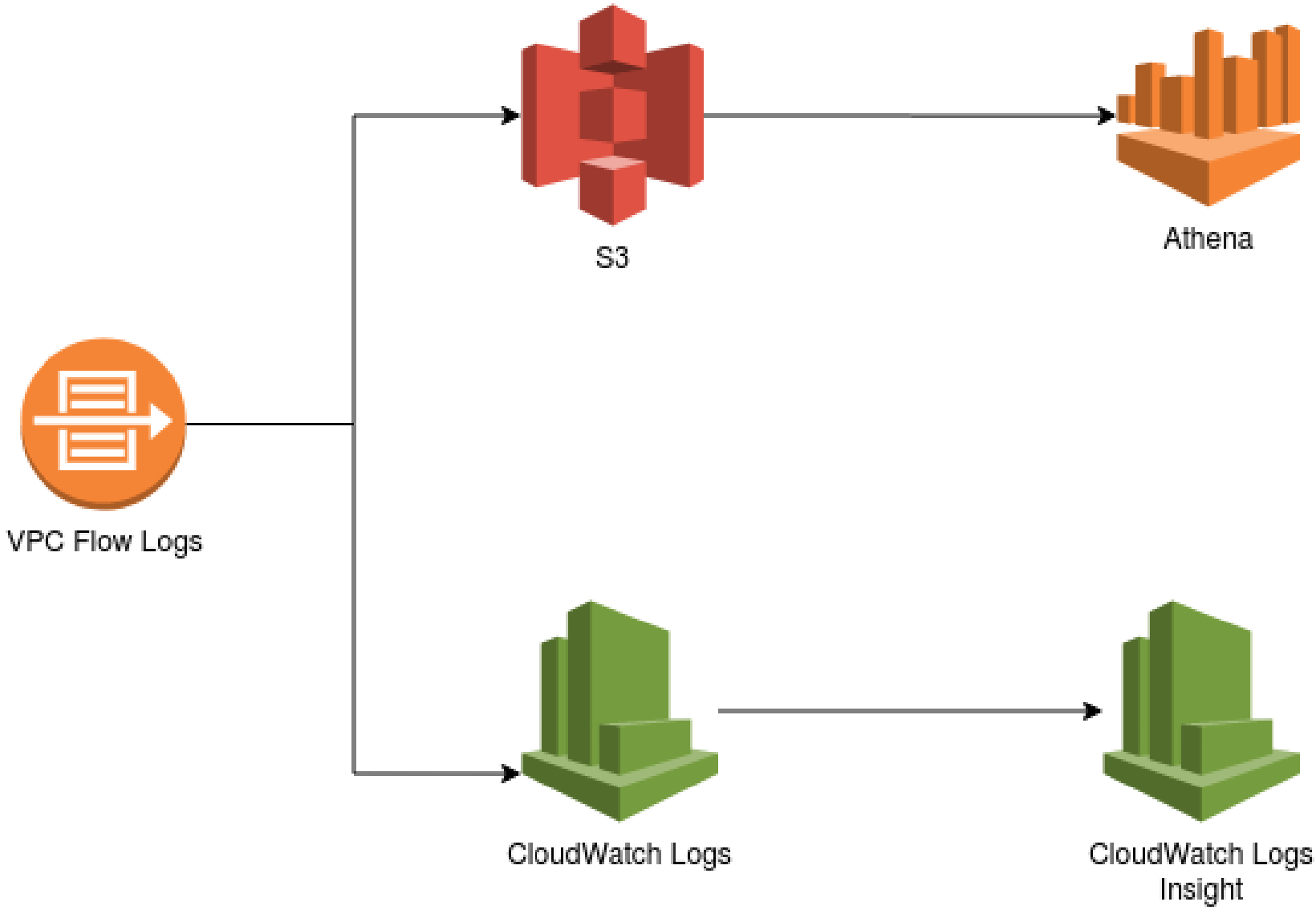
- EC2 instances
- AWS Lambda
- Kinesis streams
- Amazon ECS
- Step Functions
- Amazon SNS
- Amazon SQS

Amazon EventBridge

CloudWatch and EventBridge Summary

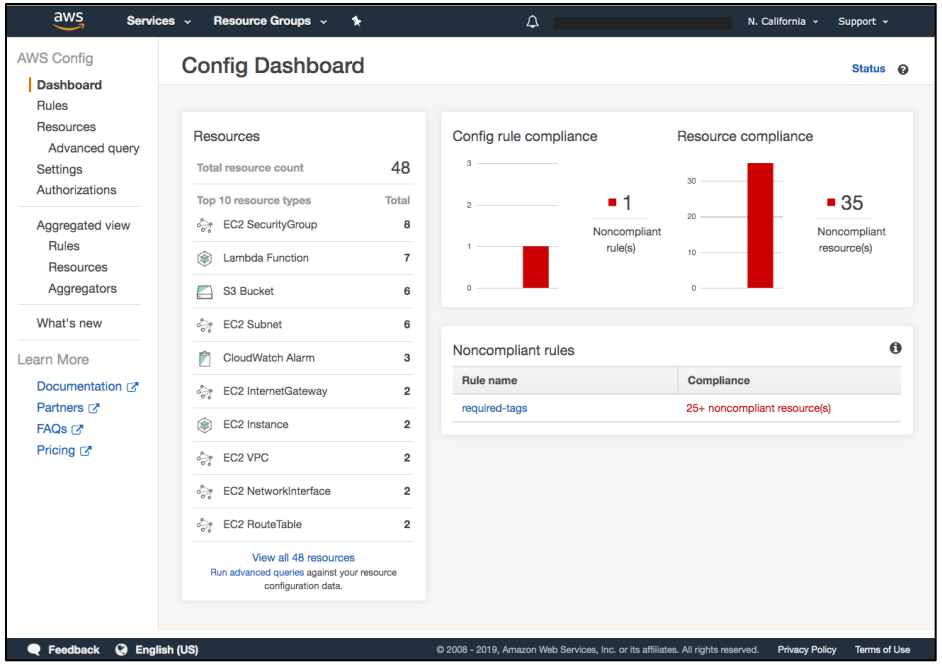


AWS Flow Logs



AWS Config

- **Assess and audit configurations of AWS resources**
- Continuous **monitoring** of configurations
- Automatically evaluate **recorded** vs **desired** configurations
- Review configuration **changes**
- View detailed configuration **history**
- Simplify **compliance auditing, security analysis** and change management



AWS Config
- Dashboard
- Rules
- Resources
  - Advanced query
- Settings
- Authorizations

- Aggregated view
  - Rules
  - Resources
  - Aggregators

- What's new

Learn More
- Documentation
- Partners
- FAQs
- Pricing

Config Dashboard

Status

Resources

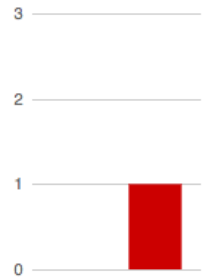
Total resource count	48

Top 10 resource types	Total
EC2 SecurityGroup	8
Lambda Function	7
S3 Bucket	6
EC2 Subnet	6
CloudWatch Alarm	3
EC2 InternetGateway	2
EC2 Instance	2
EC2 VPC	2
EC2 NetworkInterface	2
EC2 RouteTable	2

View all 48 resources

Run advanced queries against your resource configuration data.

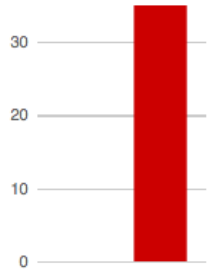
Config rule compliance



1

Noncompliant rule(s)

Resource compliance

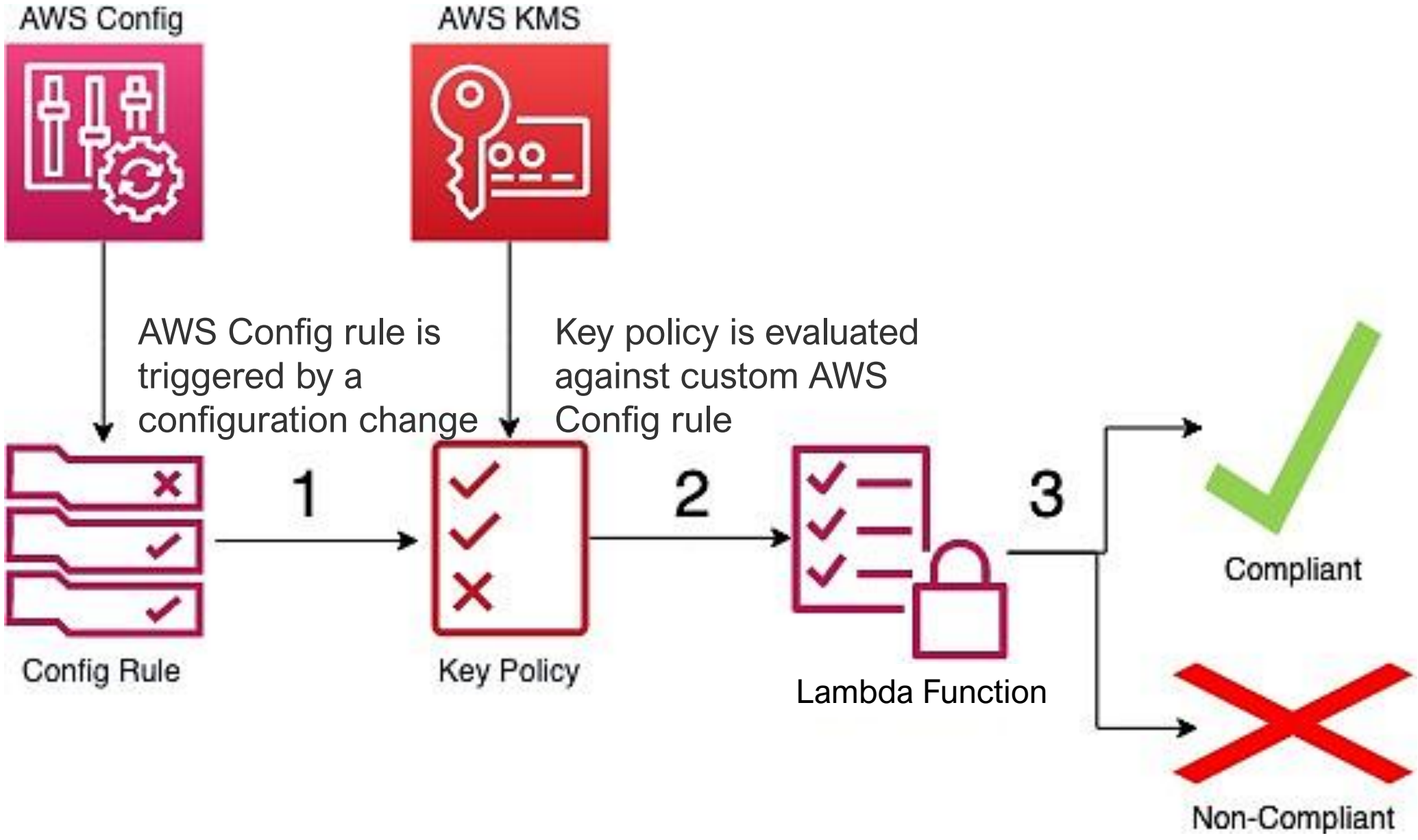


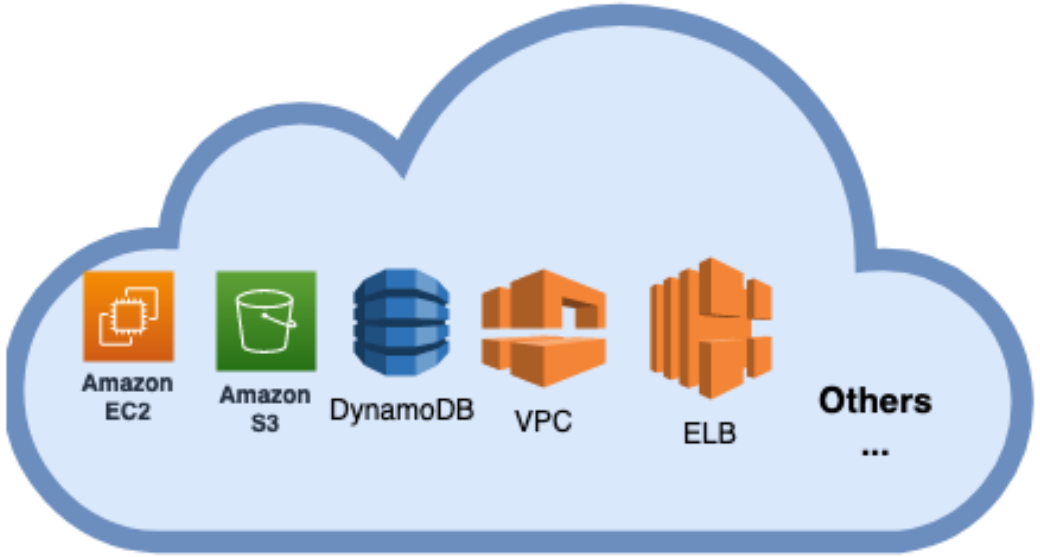
35

Noncompliant resource(s)

Noncompliant rules

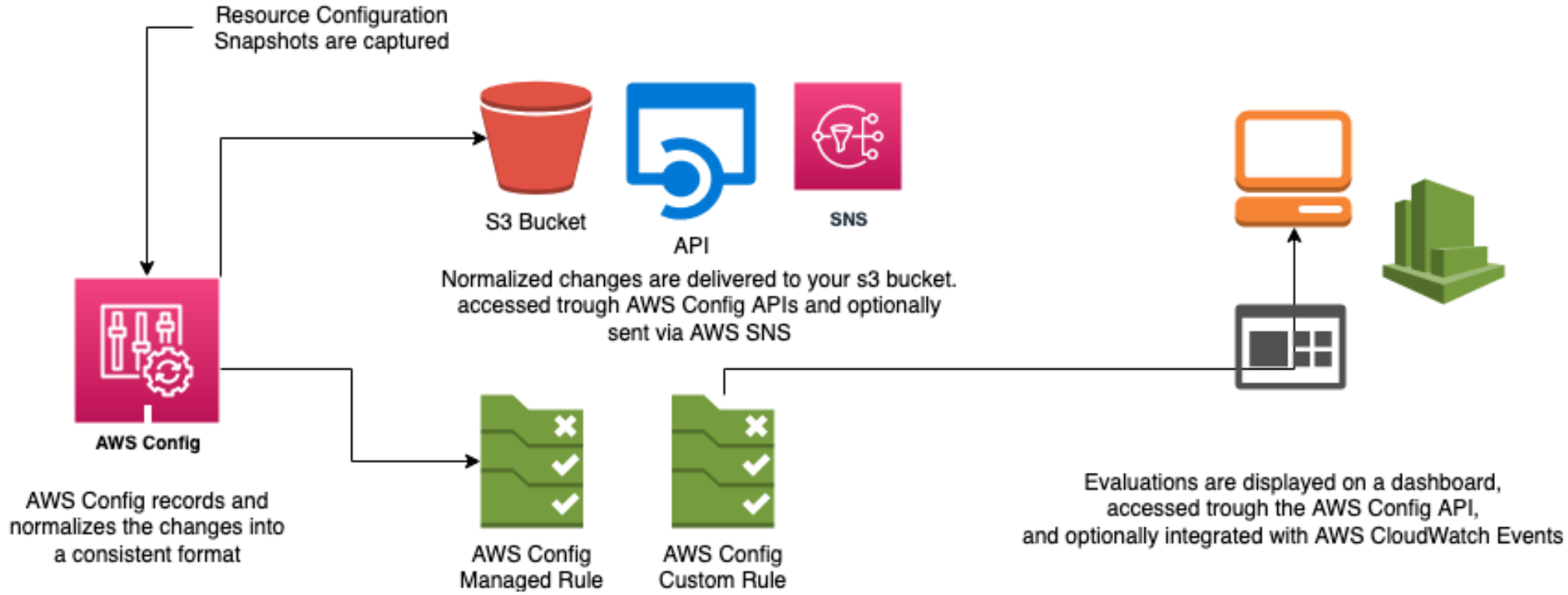
Rule name	Compliance
required-tags	25+ noncompliant resource(s)


AWS Config
AWS KMS
AWS Config rule is triggered by a configuration change
Key policy is evaluated against custom AWS Config rule
1
2
3
Config Rule
Key Policy
Lambda Function
Compliant
Non-Compliant



Example Scenario

Developers and operations teams sometimes need to adjust security groups while prototyping and troubleshooting infrastructure issues, and may forget to revert those changes



Additional Security Services

Amazon Macie

Use ML to discover, classify and protect sensitive data on S3

Protect personally identifiable information (PII), know when it moves, generate alerts

---

Amazon Inspector

Automated **security assessment** service

Define standards and best practices for applications and **validate adherence to** these **standards**

Amazon GuardDuty

Intelligent **threat detection** and continuous monitoring for **malicious activity** and unauthorized behaviour to protect AWS accounts and workloads

Analyzes **events** from CloudTrail, VPC Flow Logs, …

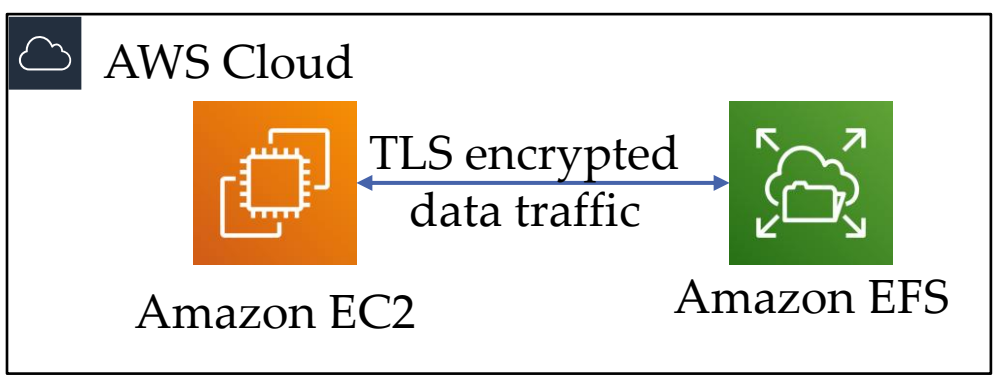
Cloud Data Security

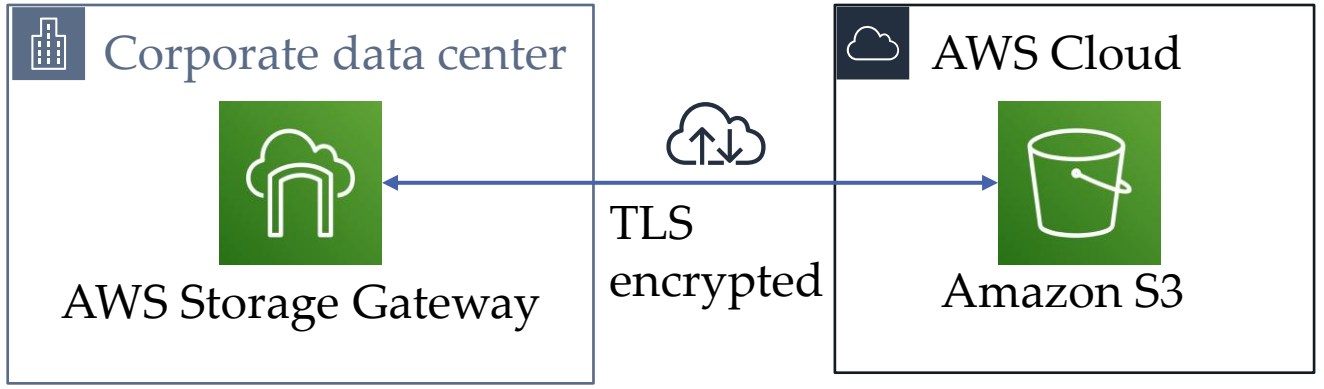
Encryption of Data at Rest

- AWS supports encryption of **data at rest**
  - **Symmetric** keys
  - AES-256
- Only those with **secret key** can decode
- **AWS KMS** manage secret keys
- **Envelope** encryption
  - KMS key to generate and en(de)crypt data keys
- Service supported by AWS KMS (**SSE-KMS**)
  - S3, EBS, EFS, RDS, **…**

Encryption of Data in Transit

- AWS supports encryption of data in transit
  - **TLS 1.2**
  - **AWS Certificate Manager** to manage, deploy, and renew TLS certs
- HTTPS creates a secure tunnel
  - Uses TLS for bidirectional exchange of data





Amazon S3

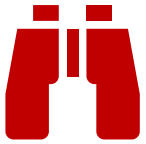
- **Object storage** service
- Stores **unlimited** amounts of unstructured data
- Data as **objects** in **buckets**
- **5 TB** maximum size of a single object
- All objects have REST globally **unique URL**
- Object are **immutable**
  - To modify change outside S3 and reupload
- All objects have key, version ID, **value**, metadata, and subresources

Amazon S3 Benefits

Durability

- Ensures data is not lost
- S3 Standard provides 11 9s (or 99.999999999%) of durability

Availability



- Access data when needed
- S3 Standard class provides four 9s (or 99.99%) availability

Scalability

- Virtually unlimited capacity
- Any single object of 5 TB or less

Security

- Fine-grained access control

Performance

- Latency measured in millisec
- Supported by many design patterns

Amazon S3 Common Usage Patterns

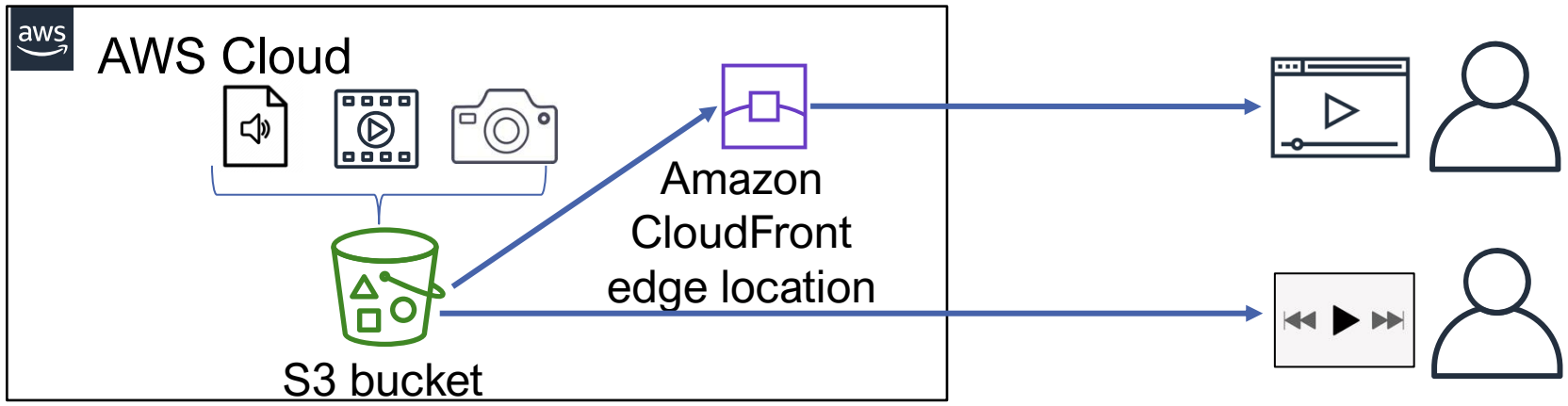
What problems can you solve by using Amazon S3?

S3 Use Case 1: Store and Distribute Web Content and Media

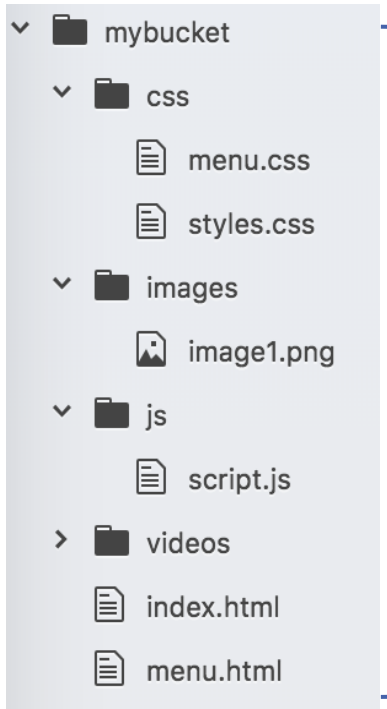
Build a redundant, scalable and highly available infrastructure that hosts media uploads and downloads.

`https://<bucket-name>.s3.amazonaws.com`

`https://<bucket-name>.s3.amazonaws.com/video.mp4`

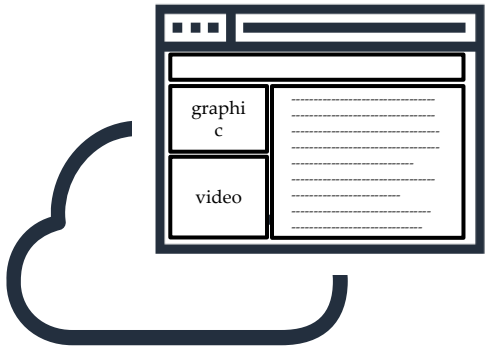


S3 Use Case 2: Host Static Websites



Example objects stored in the bucket configured for website hosting

S3 bucket with public read access



http://<bucket-name>.s3-website.<Region>.amazonaws.com

End user

- Supports static content including HTML files, images, videos and client-side scripts
- No Web server

Securing S3 Buckets and Objects

- Buckets and objects are private and protected **by default**
  - Explicitly **grant** access
- When use cases must **share** Amazon S3 data
  - Manage and **control** data access
  - Follow the principle of **least privilege**

Options for Controlling Access to S3

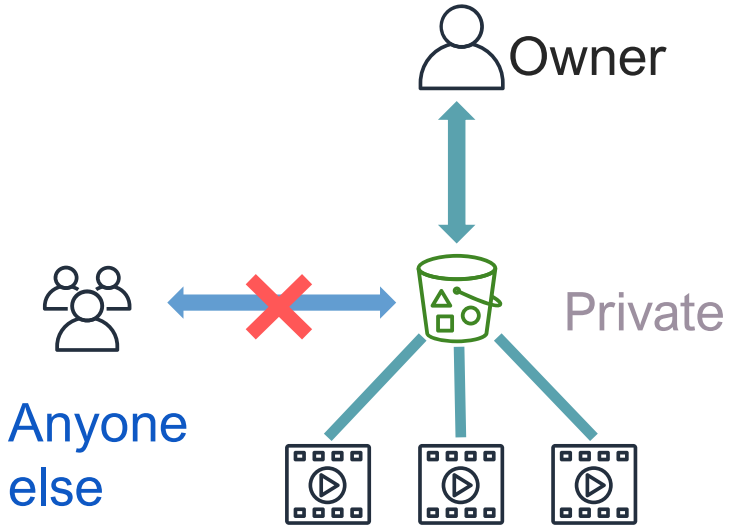
- Block **Public Access** feature: default
- **IAM policies**: IAM users or roles
- **Bucket policies**: define access to specific object or bucket
- **ACLs**: legacy mechanism
- **S3 Access Points**: configure access permissions to each application
- **Presigned URLs**: grant access to others with temporary URLs

- AWS **Trusted Advisor** bucket permission check

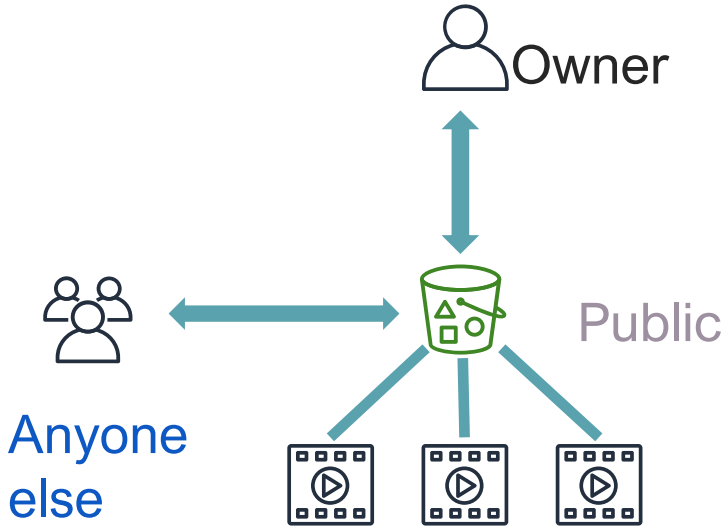
Three Access Scenarios

Configure security settings for your use case on the bucket and objects

Default S3 security settings

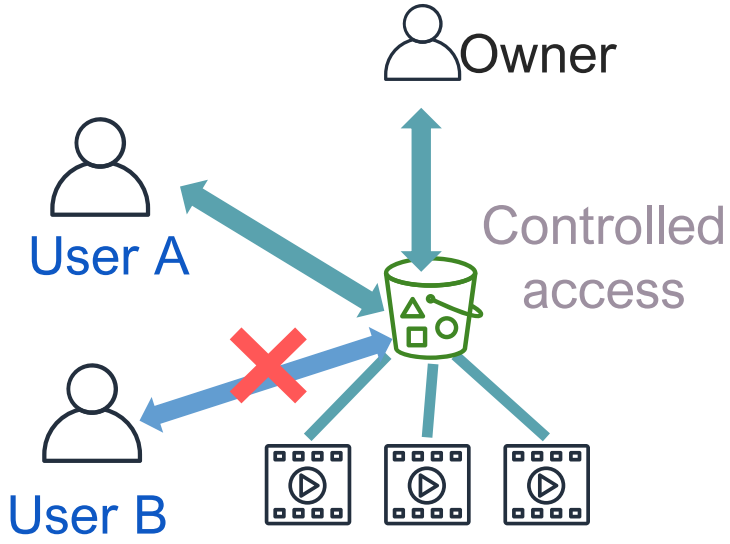


Public access S3 security settings



Most use cases do not require public access

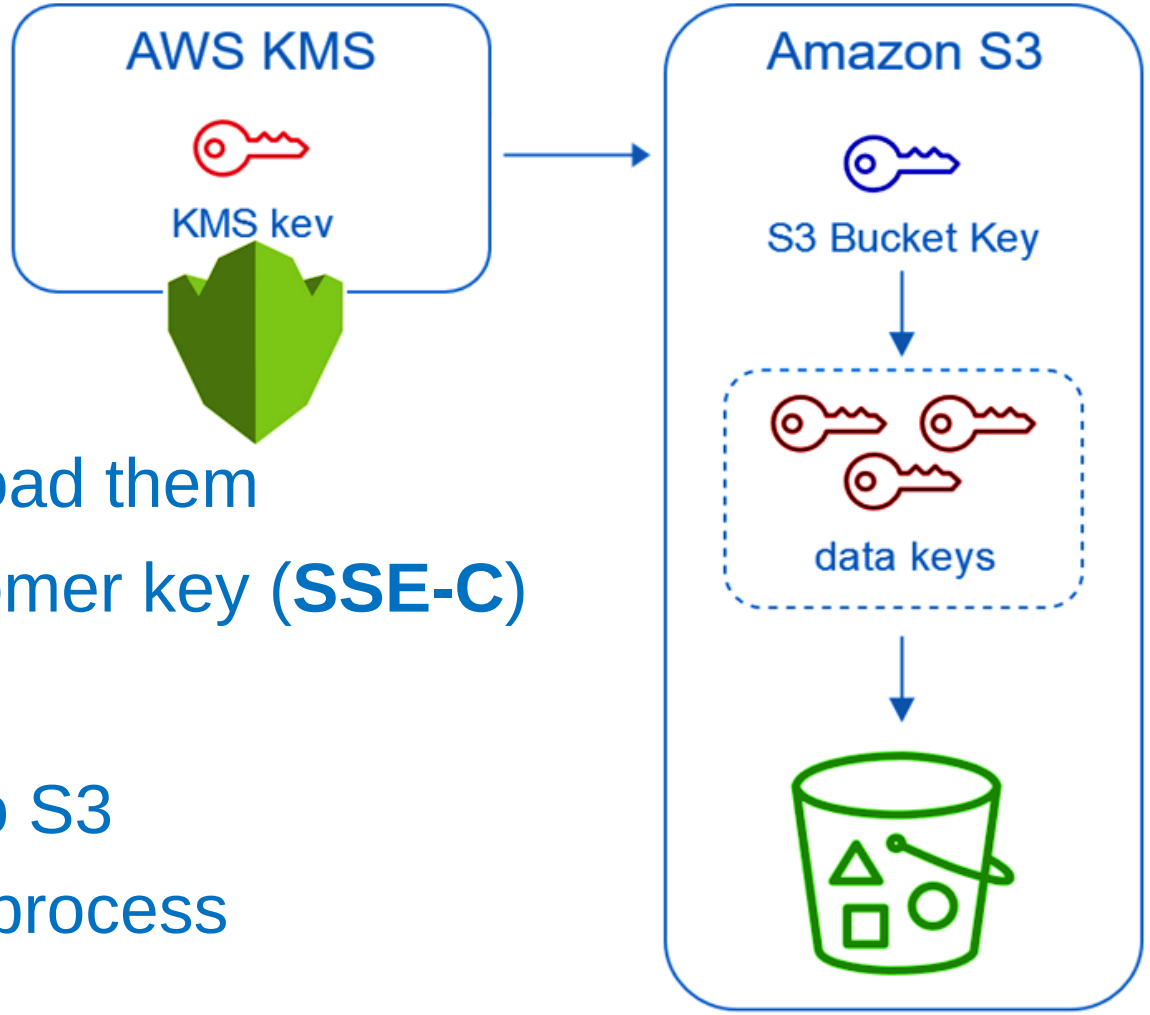
Access policy applied to S3 security settings



Use controls discussed previously

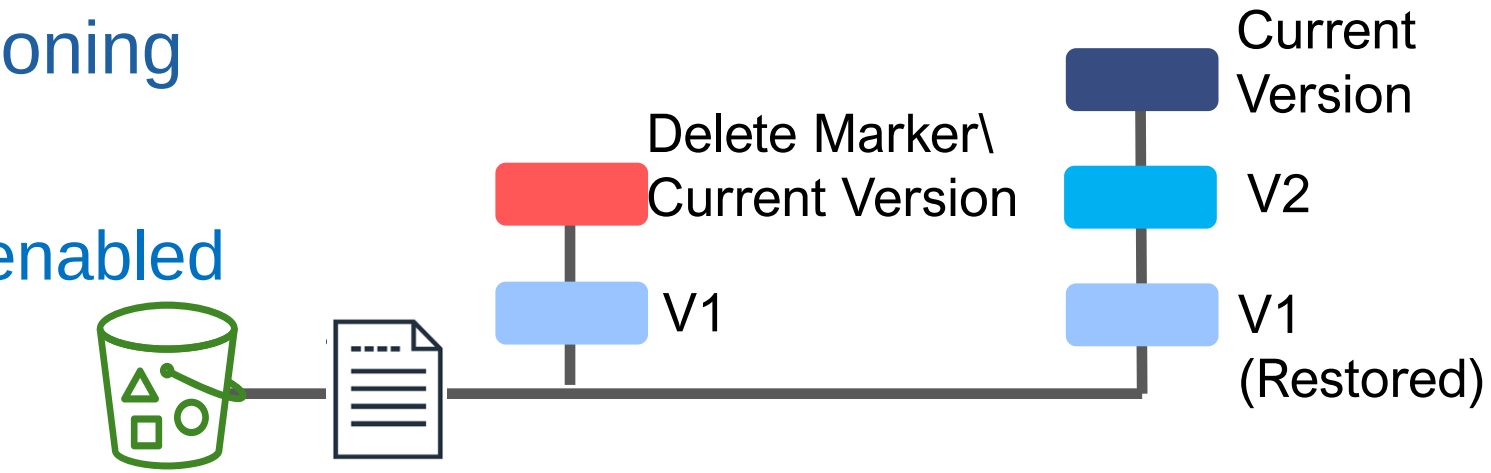
Encrypting Objects in S3

- **Server-side** encryption
  - On the bucket, selecting Default **encryption** option (**SSE-S3**)
  - **S3 encrypts** objects before it saves them and decrypts when you download them
  - Use **AWS KMS** (**SSE-KMS**) or customer key (**SSE-C**)
- **Client-side** encryption
  - Encrypt data on client and **upload** to S3
  - **Customer** manages the encryption process



S3 Versioning

- Protects against accidental **overwrites** and **deletes**
- No performance penalty
- Generates **new version** with every upload
- Enables **retrieval** of deleted objects or **rollback** to previous versions
- You **cannot disable** versioning
- States of S3 bucket
  1. Default: Versioning **not** enabled
  2. Versioning-**enabled**
  3. Versioning-**suspended**



Amazon Database Options

Relational databases

Amazon RDS

Amazon Redshift

Amazon Aurora

Non-relational databases

Amazon DynamoDB

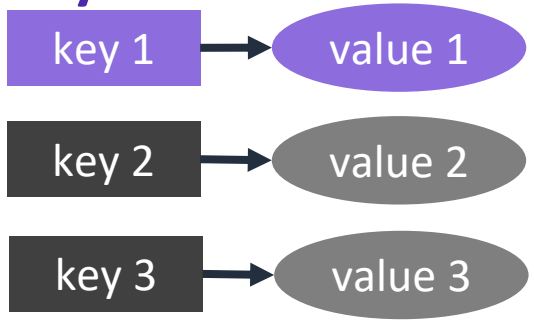
Amazon ElastiCache

Amazon Neptune

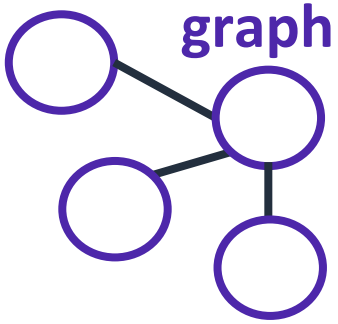
We focus on these two

ID	Name	Description	Price
10001	AAAA	Description of AAAA	100
10002	BBBB	Description of BBBB	200
10003	CCCC	Description of CCCC	200
.	.	.	.
.	.	.	.
10999	XXYZ	Description of XXYZ	500

key-value



graph

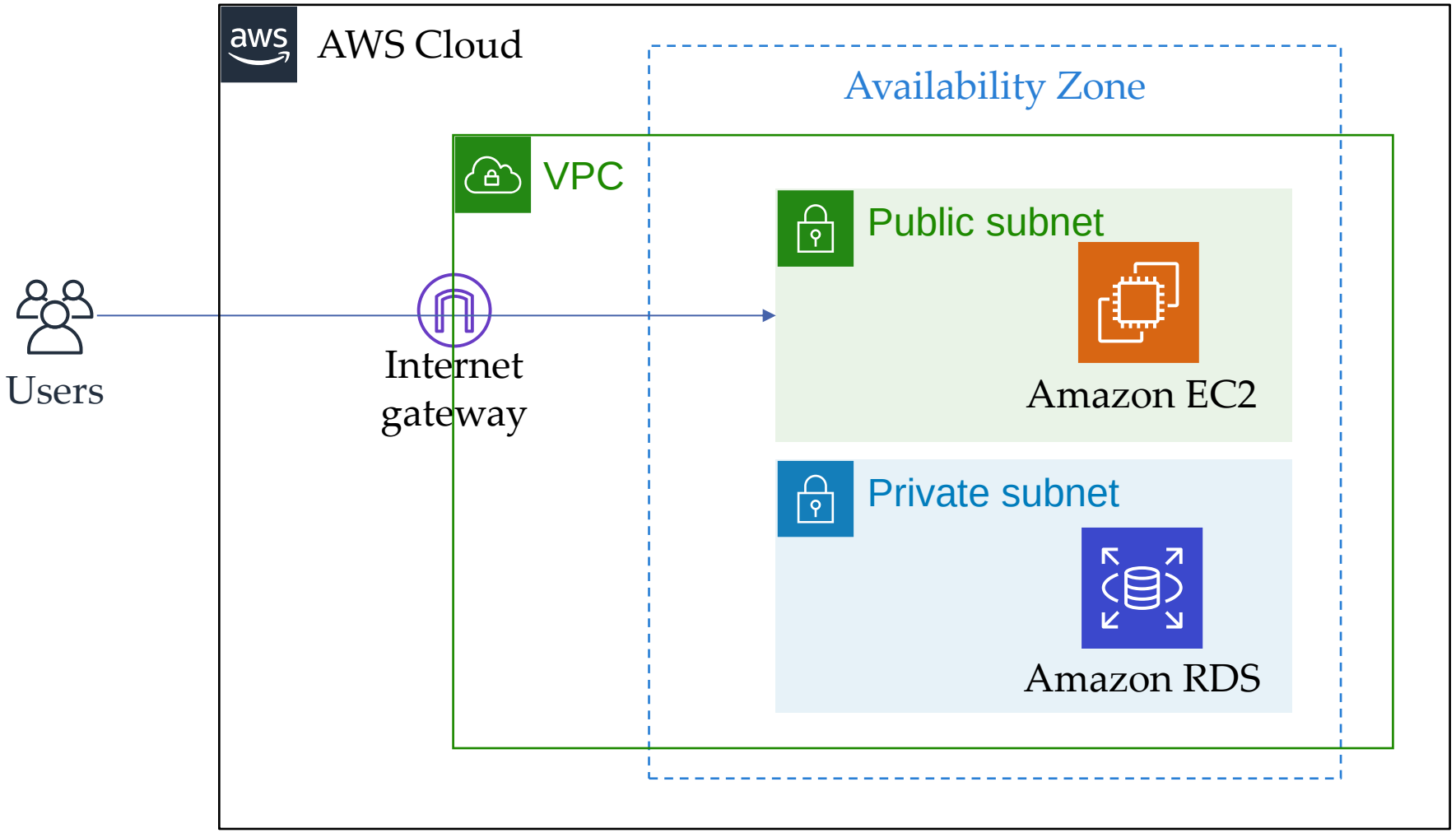


Amazon RDS

Database types supported

- Microsoft SQL Server
- Oracle
- MySQL
- PostgreSQL
- Aurora
- MariaDB

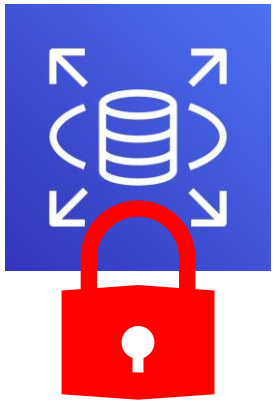
Amazon RDS in Virtual Private Cloud (VPC)



Securing RDS Databases

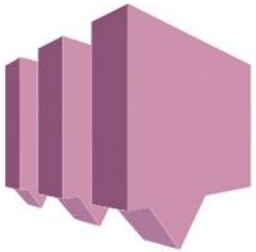
- Run the RDS instance in **VPC**
  - Provides service **isolation** and **IP firewall** protection
- Use AWS **IAM policies** for authentication and access
  - **Permissions** determine who is allowed to manage RDS resources
- Use **security groups** to control what **IP addresses** or **EC2 instances** can connect to your database
  - By default, network access is disabled
  - Open TCP port where database is accessible
- Use **TLS** for encryption in transit

Securing RDS Databases (cont)



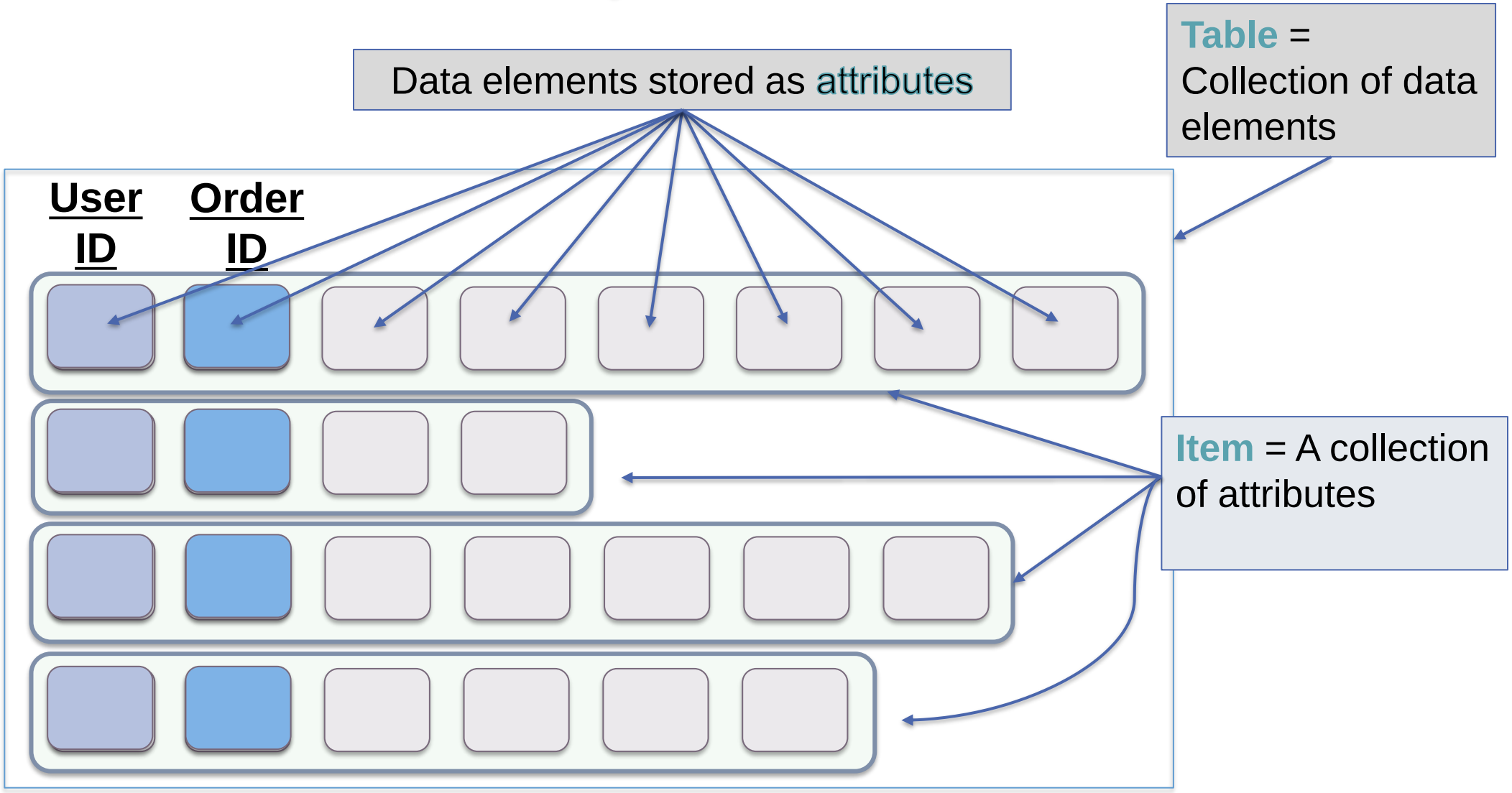
- **RDS encryption** on DB instances and snapshots
  - Secure data at rest
  - Transparent to client applications
- Use security features of **DB engine**

  e.g. password complexity, who can login to database
- Configure **event notifications** to alert for important RDS events
  - e.g. shutdown, SG changed
  - email, text, HTTP



amazon
SNS

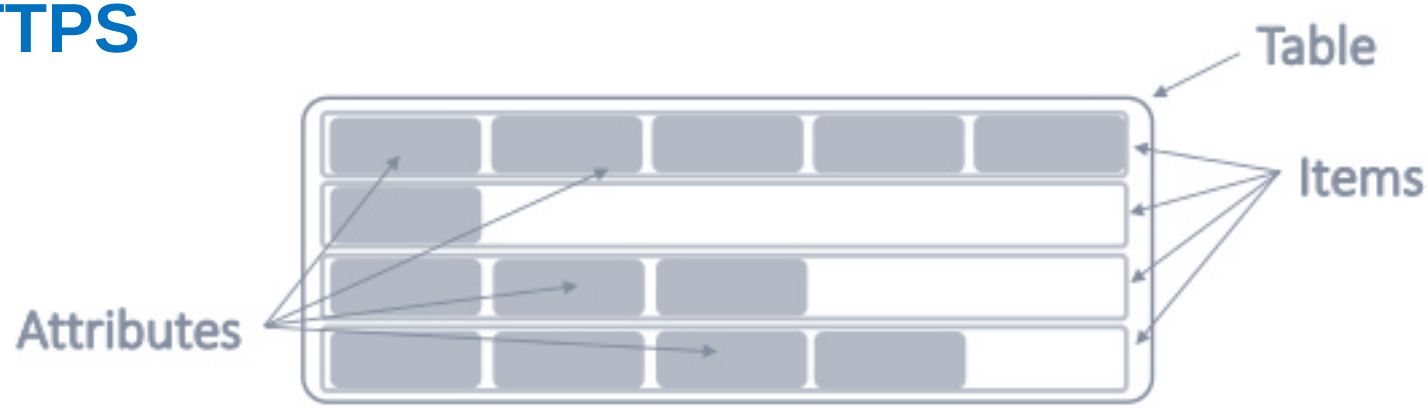
DynamoDB


Data elements stored as attributes
Table = Collection of data elements
User ID
Order ID
Item = A collection of attributes

DynamoDB Security

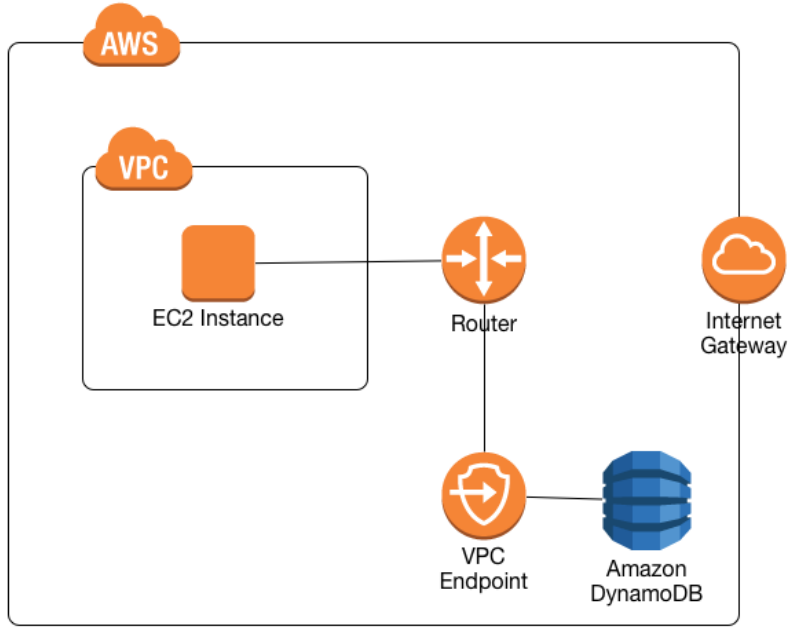
Security provided by **default**

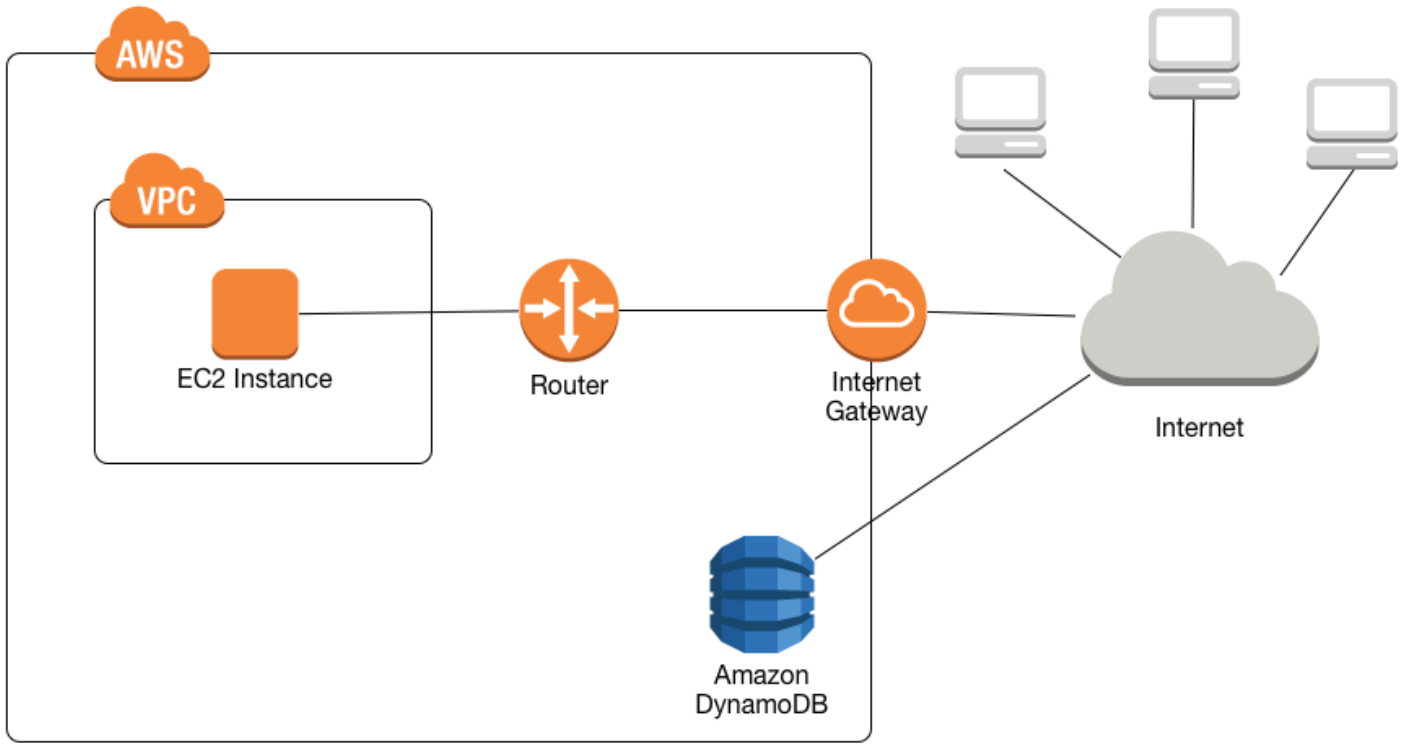
- **Encryption at rest**
  - **All data** stored in tables, indexes, and backups
- **Encryption in transit**
  - All communications to and from DynamoDB and other AWS resources use **HTTPS**

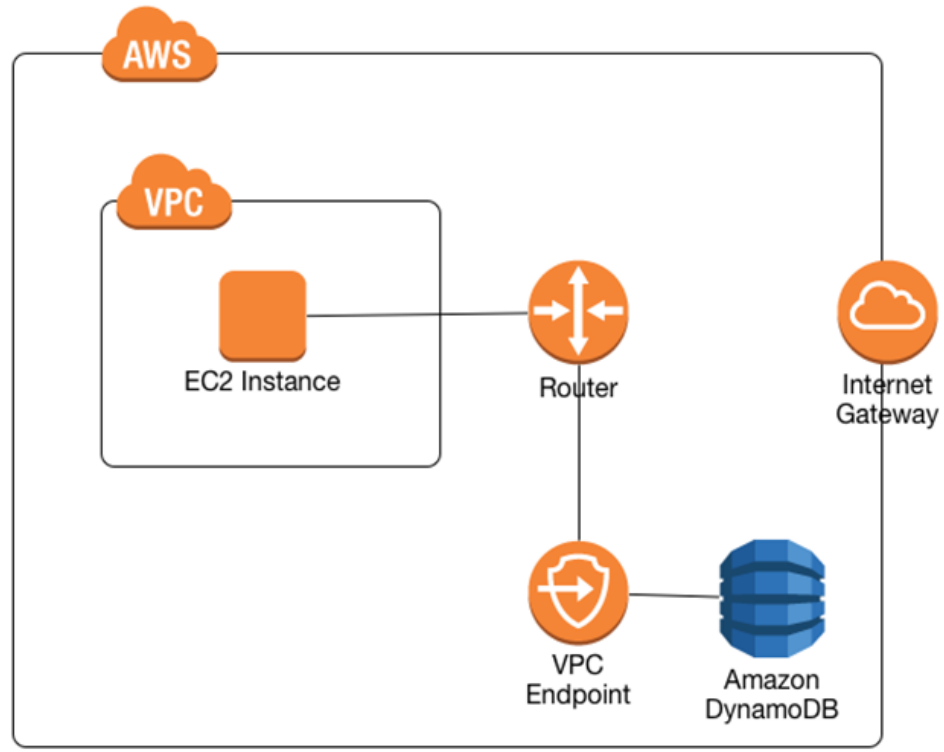


Further DynamoDB Security Recommendations

- Use **IAM roles** to authenticate
- Use **IAM policies**
  - Fine-grained access permissions
  - Define access at **table, item or attribute level**
  - **Grant least privilege**
- Configure **VPC endpoints**
  - Prevents traffic from **traversing the internet**
  - **VPC endpoint policies** allow you to control API access to a table
- Consider **client-side encryption**

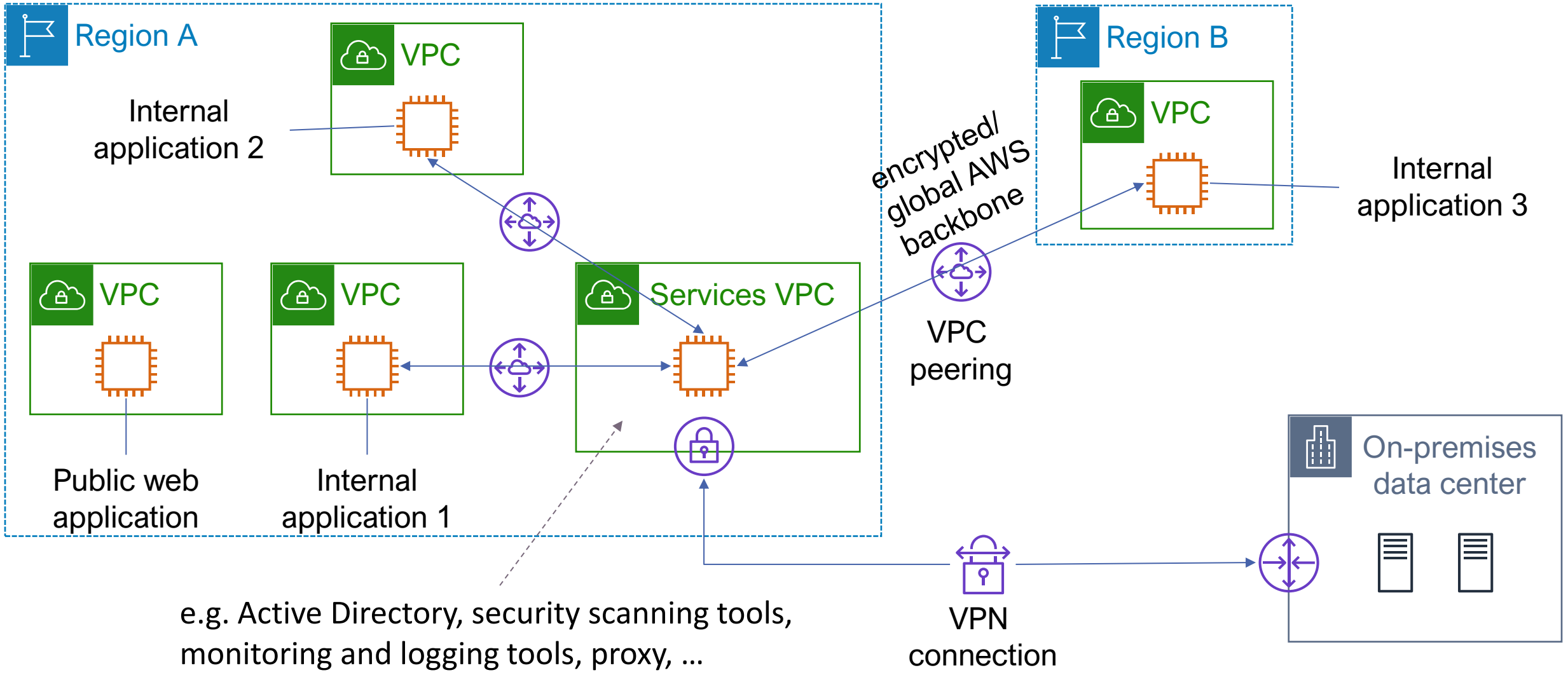



AWS
VPC
EC2 Instance
Router
Internet Gateway
Internet
Amazon DynamoDB

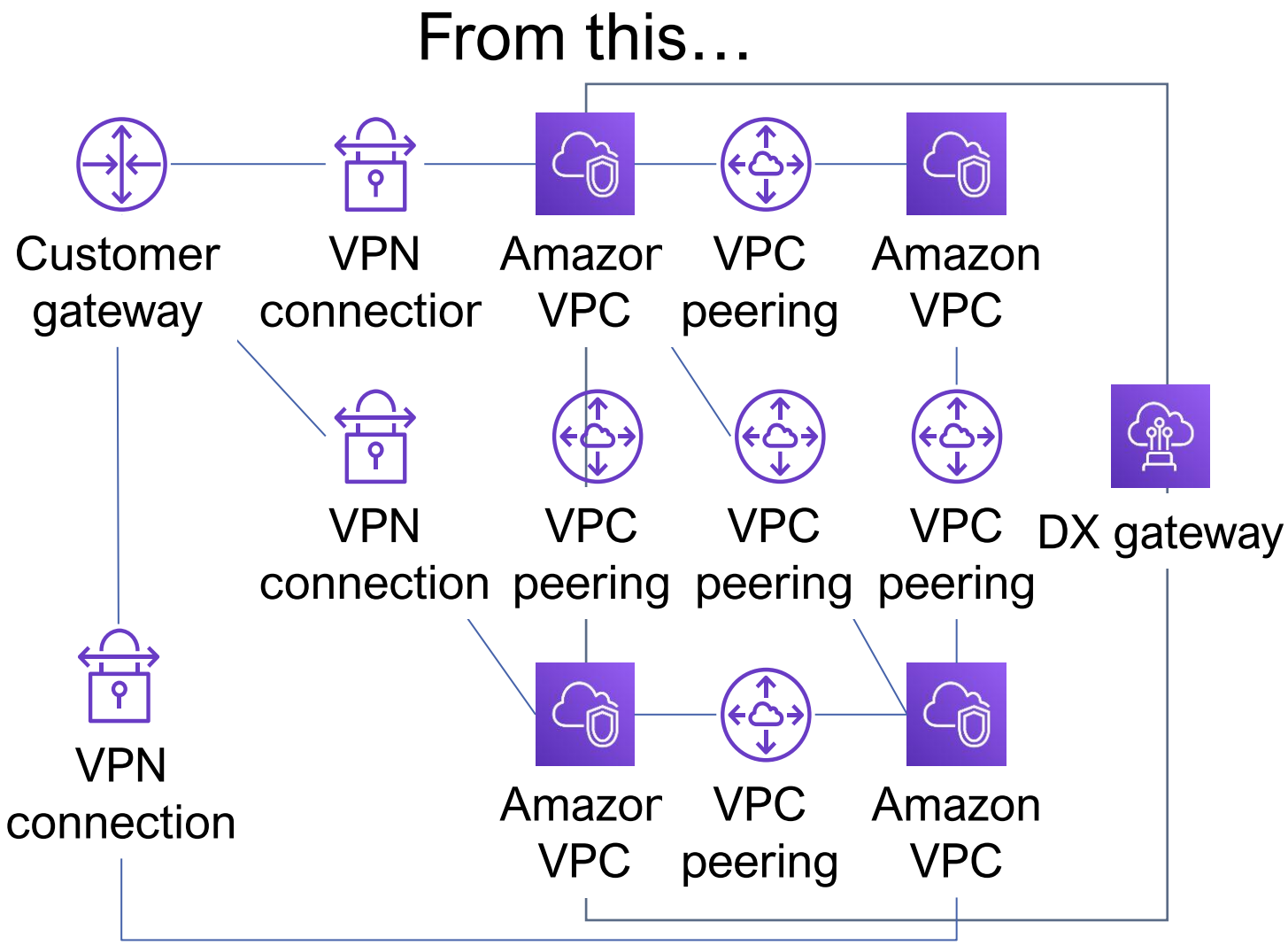

AWS
VPC
EC2 Instance
Router
Internet Gateway
VPC Endpoint
Amazon DynamoDB

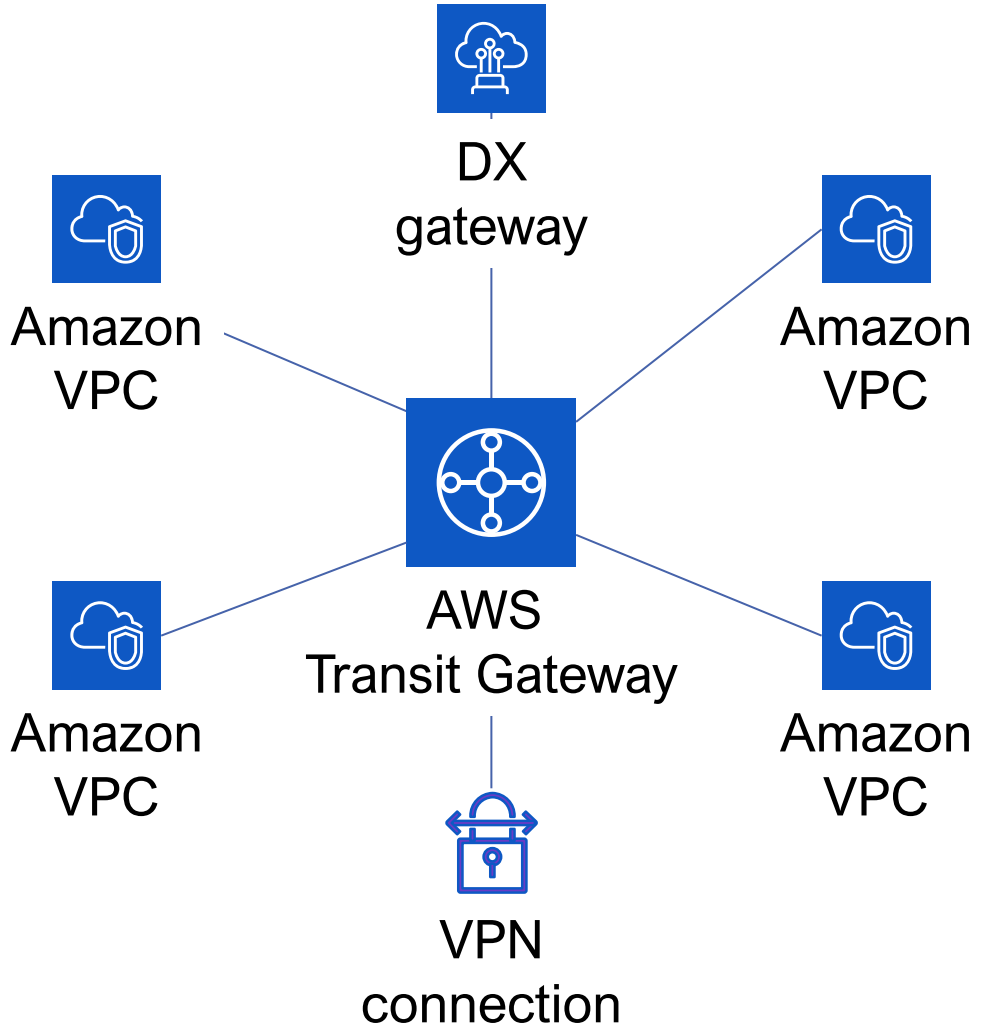
Connecting Networks and Services

Example VPC Peering for Shared Resources



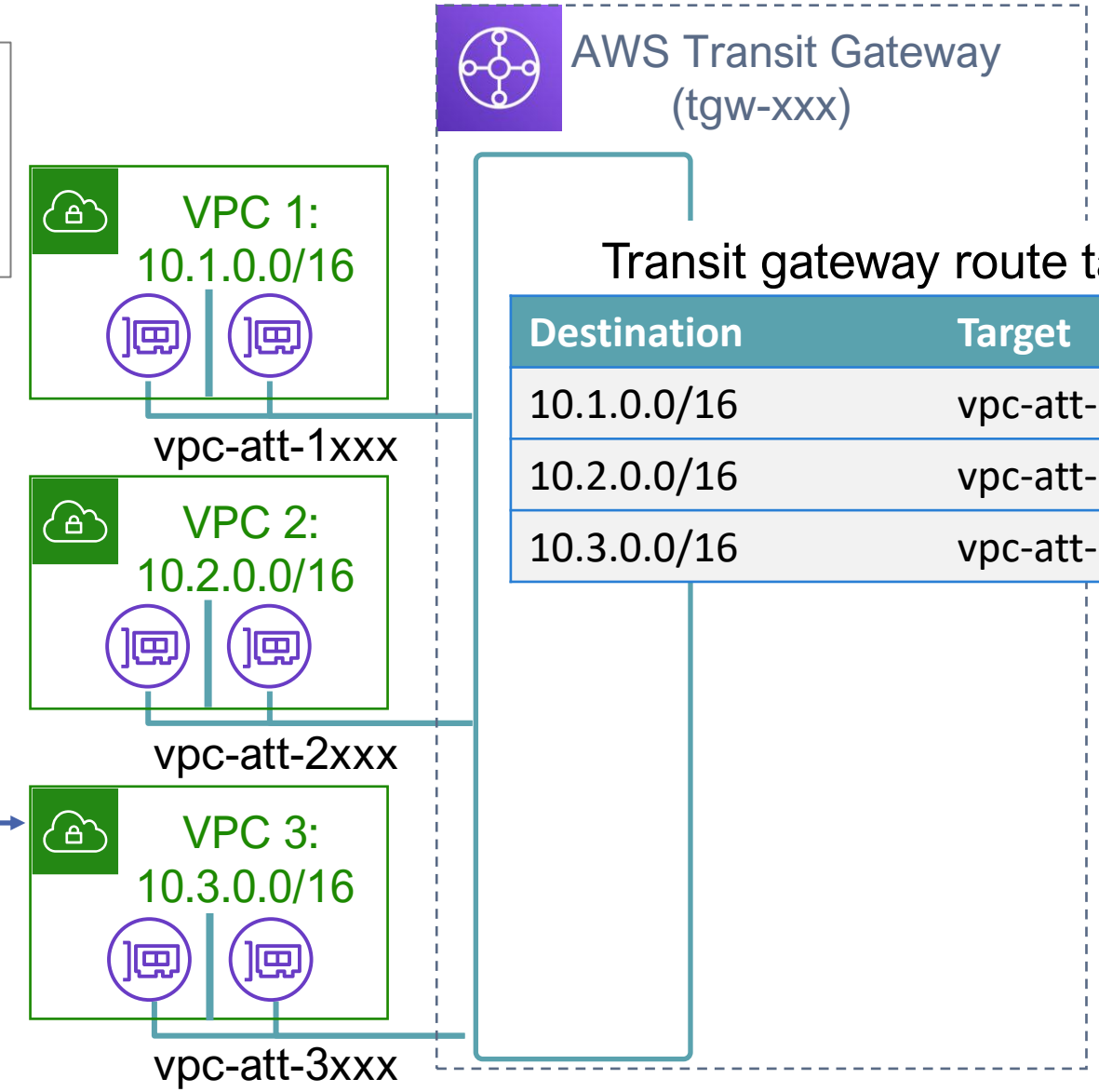
Scaling Networks Across Multiple VPCs





Using AWS Transit Gateway to Achieve VPC Isolation

Scenario: We want VPN access but isolated VPC connectivity



Transit gateway route table

Destination	Target
10.1.0.0/16	vpc-att-1xxx
10.2.0.0/16	vpc-att-2xxx
10.3.0.0/16	vpc-att-3xxx

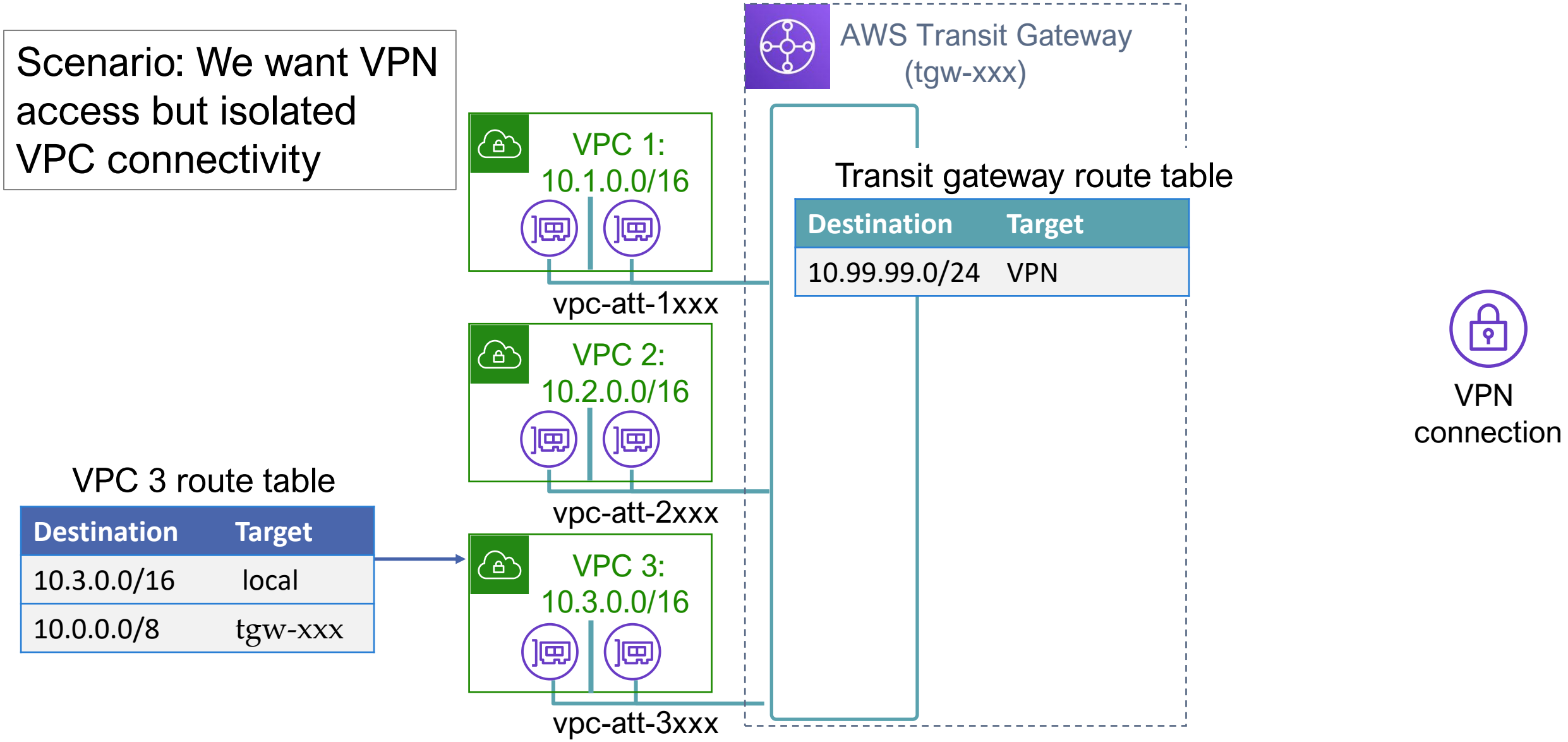
VPN connection

VPC 3 route table

Destination	Target
10.3.0.0/16	local
10.0.0.0/8	tgw-xxx

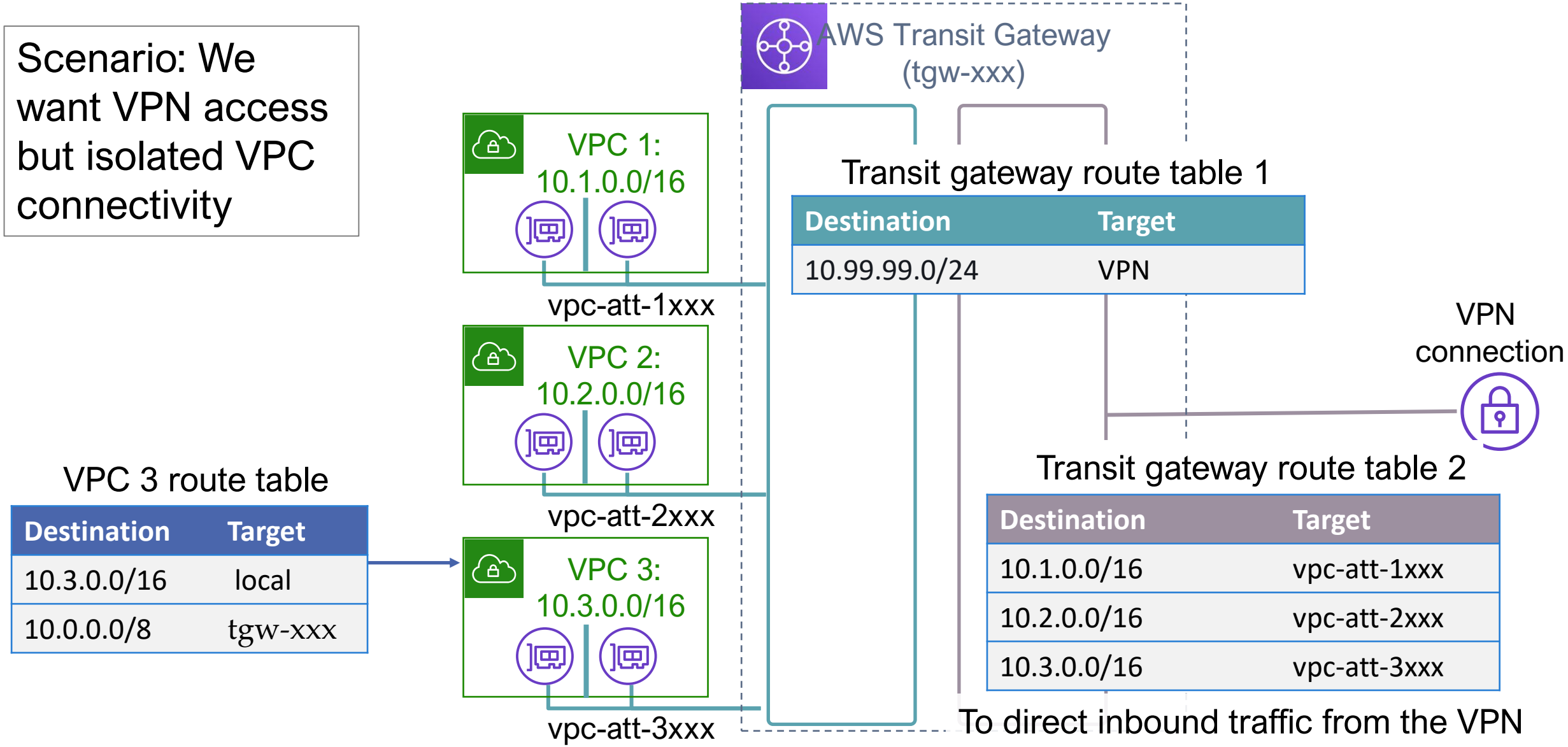
Using AWS Transit Gateway to Achieve VPC Isolation (cont)

Scenario: We want VPN access but isolated VPC connectivity



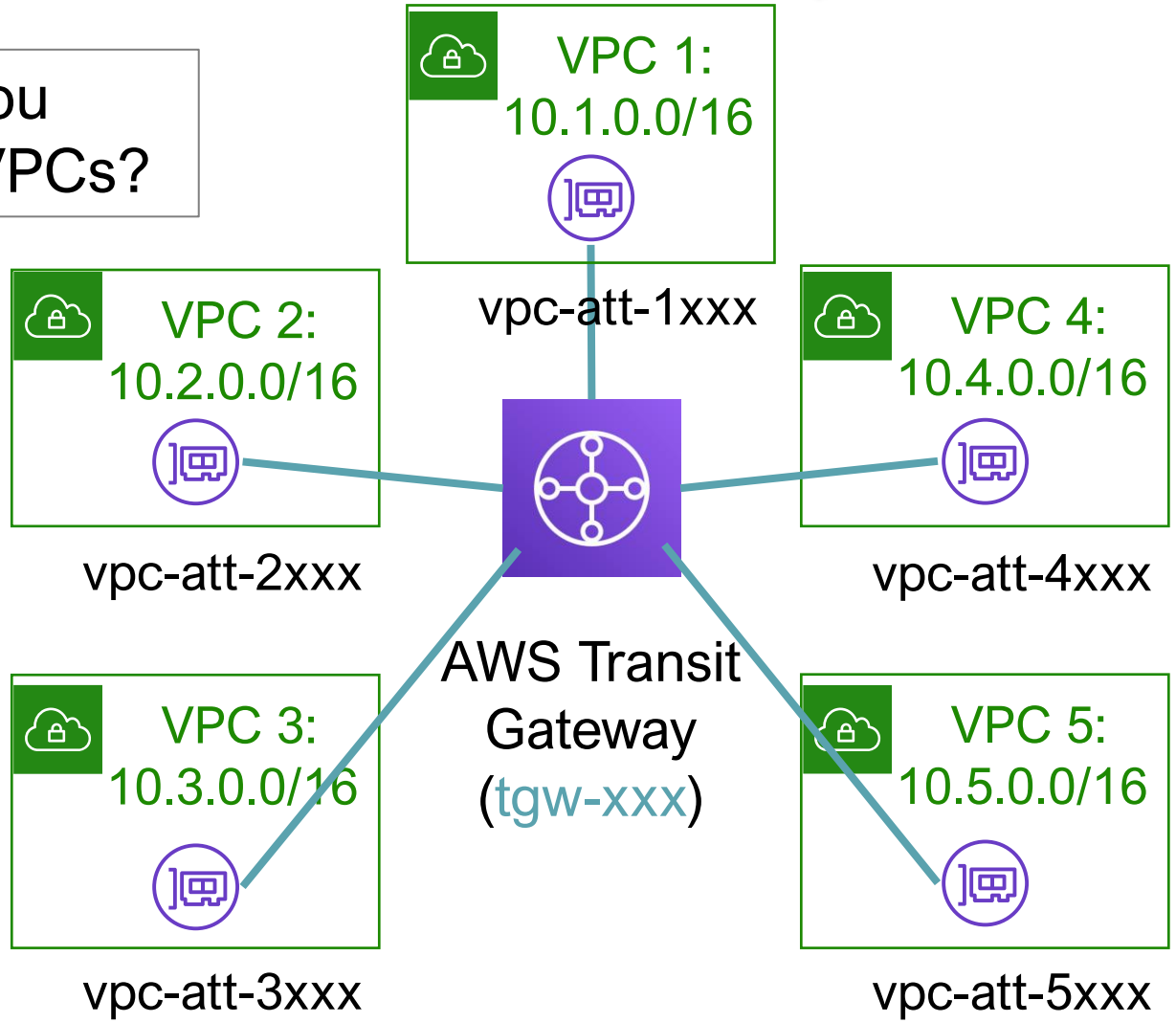
Using AWS Transit Gateway to Achieve VPC Isolation (cont)

Scenario: We want VPN access but isolated VPC connectivity



AWS Transit Gateway: Question

Scenario: How do you connect these five VPCs?



VPC # route table

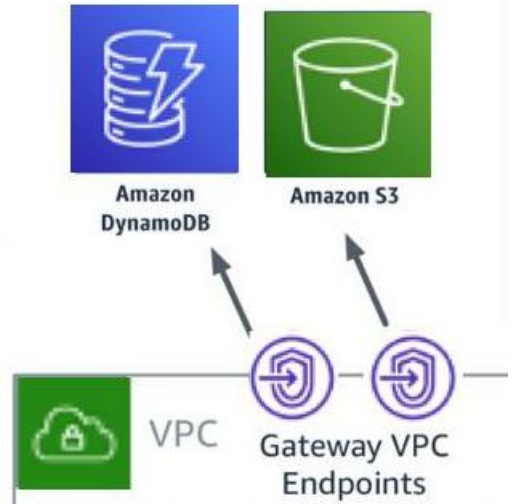
Destination	Target
10.#.0.0/16	local
?	?

Transit gateway route table

Destination	Target
?	?

VPC Endpoints

- Allow **privately connecting** VPC to supported AWS services
- Enable traffic between VPC and other services **without leaving** Amazon network
- **Do not** require internet gateway, VPN, NAT devices, or firewall proxies
- Horizontally scaled, redundant, highly available **virtual devices**



Two Types of VPC Endpoints

1. Gateway endpoint

Specify it as a **target** for a route in route table for traffic destined to a **supported** AWS service

- S3
- DynamoDB

2. Interface endpoint

Elastic network interface with **private IP address** that serves as an entry point for traffic destined to a supported service

Examples

- CloudWatch
- EC2 API
- Elastic Load Balancing
- Config, …

VPC endpoint

to privately connect AWS services
without Internet gateway/NAT gateway

Where to sit?

Inside a subnet

Interface endpoint

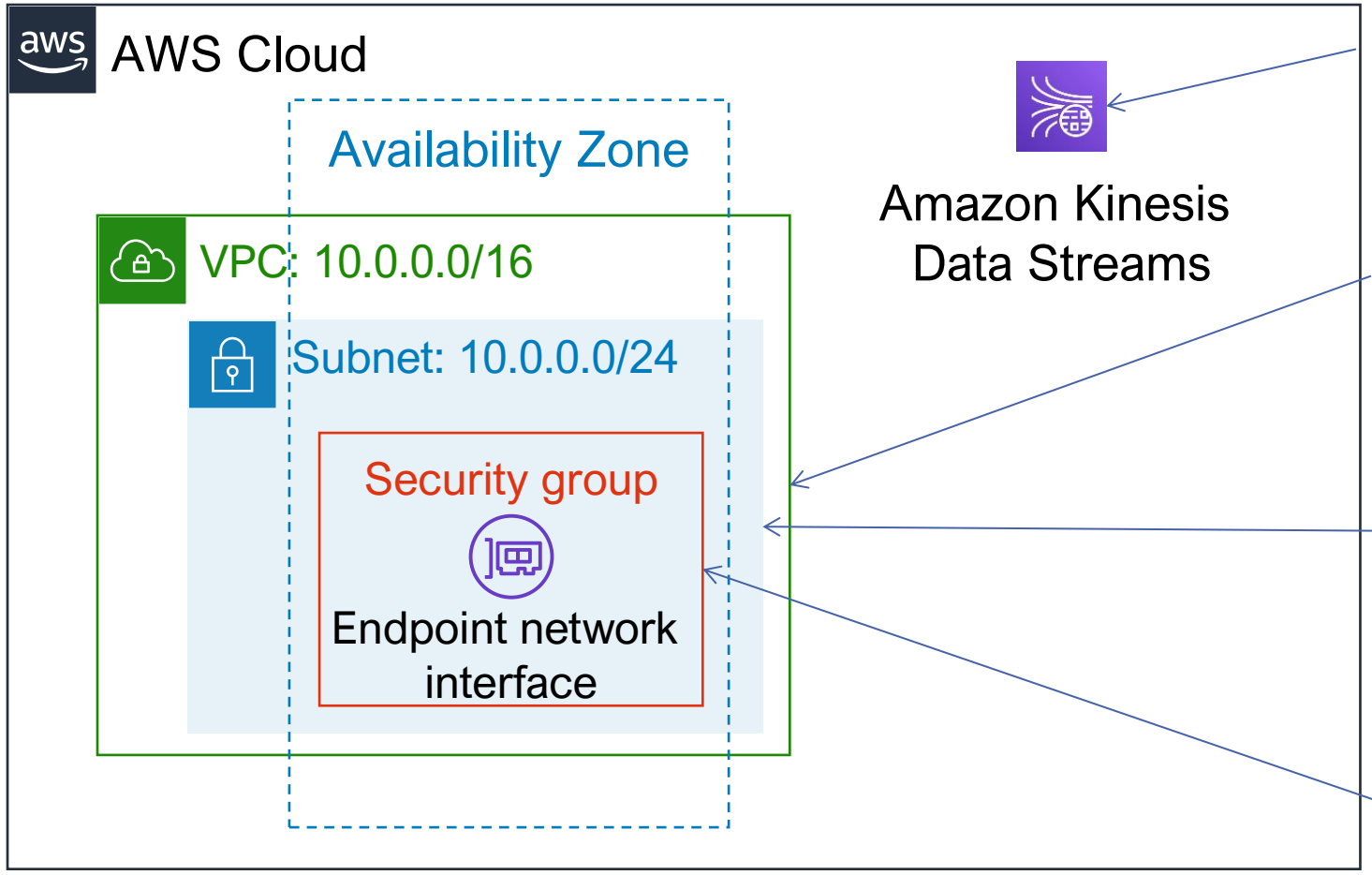
associated with a security group

Inside a VPC

Gateway endpoint

associated with a route table

Setting up an Interface Endpoint



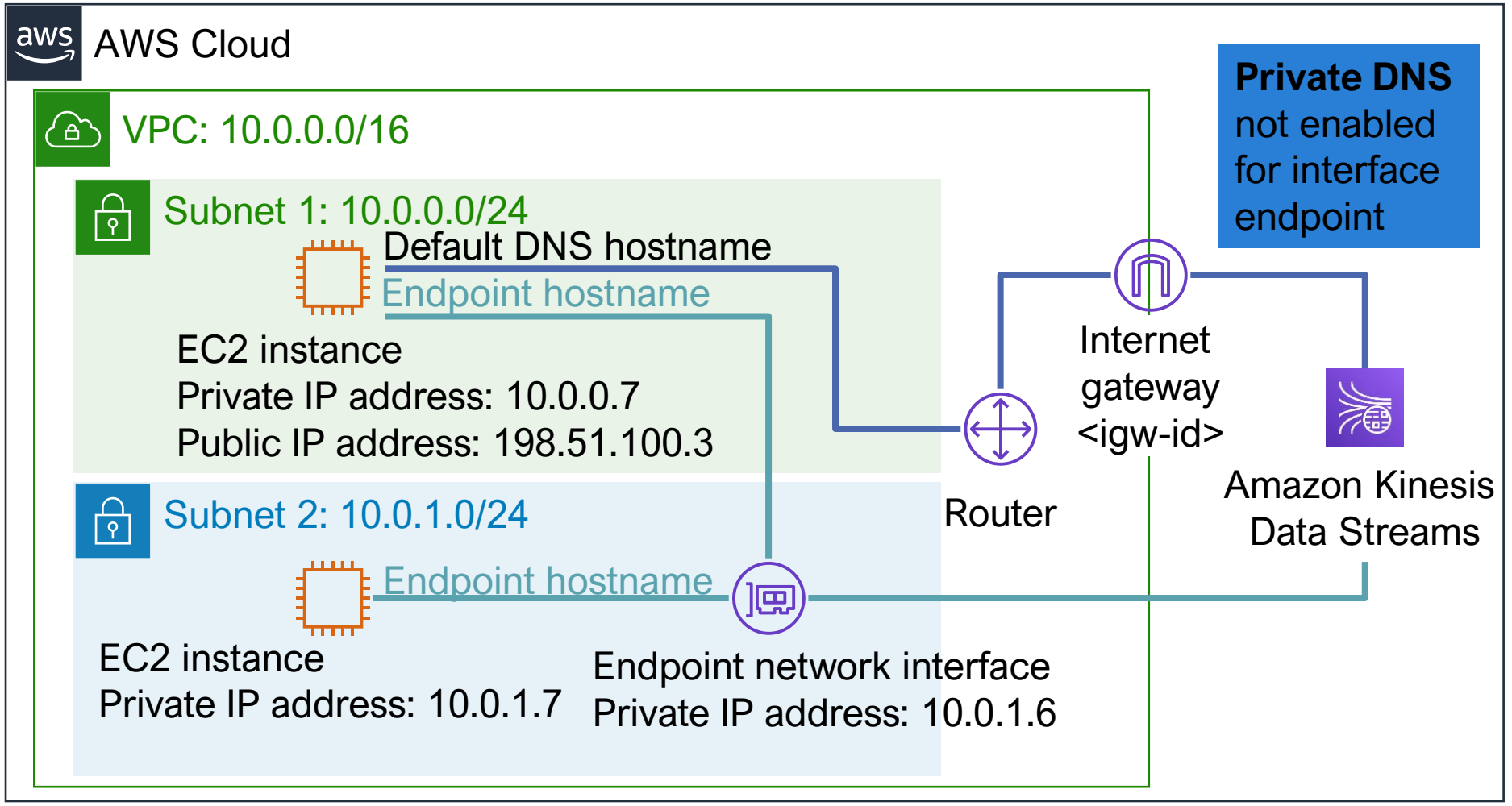
1. Specify the **AWS service**, endpoint service, or AWS Marketplace service you want to connect to

2. Choose the **VPC** where you want to create the interface endpoint

3. Choose a **subnet** in your VPC that will use the interface endpoint.

4. Specify the **security groups** to associate with the network interface

Example of Using VPC Endpoints



Subnet 1 route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-id

Internet

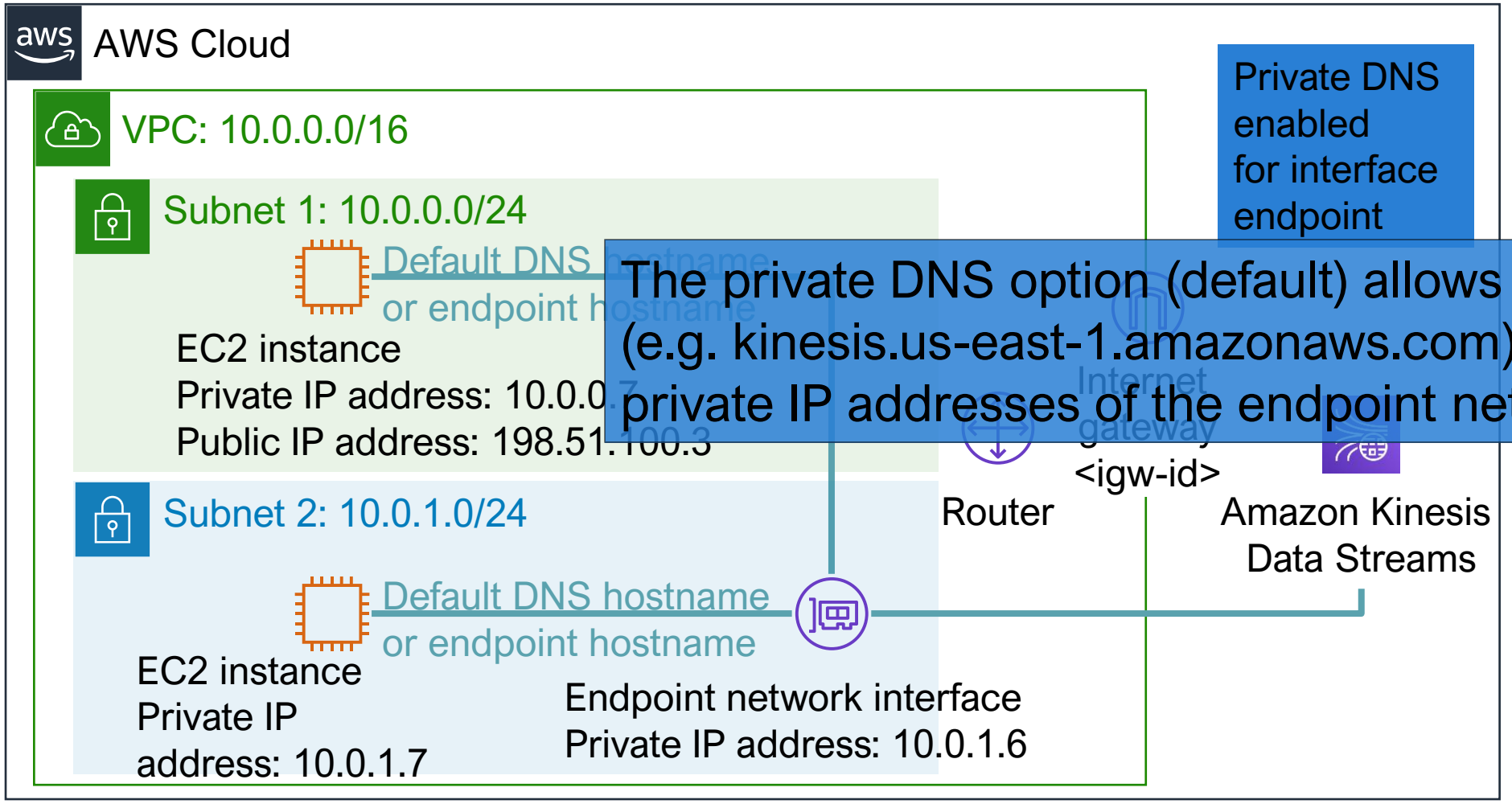
Subnet 2 route table

Destination	Target
10.0.0.0/16	local

Default DNS hostname: kinesis.us-east-1.amazonaws.com

Endpoint-specific DNS hostname: vpce-123-ab-kinesis.us-east-1.vpce.amazonaws.com

Example of Using VPC Endpoints (cont)



Private DNS enabled for interface endpoint

The private DNS option (default) allows the default DNS (e.g. kinesis.us-east-1.amazonaws.com) to resolve to the private IP addresses of the endpoint network interfaces

Subnet 1 route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-id

Subnet 2 route table

Destination	Target
10.0.0.0/16	local

Default DNS hostname: kinesis.us-east-1.amazonaws.com

Endpoint-specific DNS hostname: vpce-123-ab-kinesis.us-east-1.vpce.amazonaws.com