Cloud-native design

After completing this episode, you should be able to:

* Compare monolithic application design with decoupled, microservice application design.
* Explain the role of Application Programming Interfaces (APIs) in application design.
* Explain how multi-tier architectures and event-driven architectures support cloud-native network design.

Key point 1

Cloud-native design decouples system components to function as microservices.

Key point 2

Purpose-built Amazon Web Services (AWS) services can be selected to support each individual microservice.

Key point 3

Event-driven architecture relies on events to trigger actions and communication between microservices and other resources.

Key point 4

A multi-tier architecture places resources in subnets according to the types of security protections that the resources need. An example is the placement of databases in a private subnet with no access to the internet, while web servers might reside in a public subnet with direct access from the internet. Specifically, a three-tier architecture defines the presentation (client) tier, the logic (server) tier, and the data (database) tier.

Key point 5

The Well-Architected Framework defines standards for AWS deployments related to operational excellence, security, reliability, performance, cost optimization, and sustainability.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* What is an Event-Driven Architecture? <https://aws.amazon.com/event-driven-architecture>
* Three-tier architecture overview: <https://docs.aws.amazon.com/whitepapers/latest/serverless-multi-tier-architectures-api-gateway-lambda/three-tier-architecture-overview>
* AWS Well-Architected Framework links:
  + AWS Well-Architected: <https://aws.amazon.com/architecture/well-architected>
  + AWS Well-Architected Framework: <https://docs.aws.amazon.com/wellarchitected/latest/framework/welcome>
  + AWS Well-Architected Tool: <https://aws.amazon.com/well-architected-tool>
* Purpose-built AWS services:
  + AWS Cloud Databases: <https://aws.amazon.com/products/databases>
  + Web Hosting Services on AWS: <https://aws.amazon.com/websites>
  + Discover, deploy, and manage software that runs on AWS: <https://aws.amazon.com/marketplace>

|  |  |
| --- | --- |
| * **Started on** | Sunday, 13 July 2025, 7:22 AM |
| **State** | Finished |
| **Completed on** | Sunday, 13 July 2025, 7:27 AM |
| **Time taken** | 4 mins 57 secs |
| **Grade** | **100.00** out of 100.00 |

Top of Form

Question **1**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

Which of the following is the best description of the AWS Well-Architeted Tool?

Select one:

a. It evaluates the architecture you create for your software product

b. It is a tutorial that prescribes the software architecture for your product

c. It provides guidelines to help create the architecture for your software product Correct

d. It creates the architecture for your software product based on your inputs

Feedback

Your answer is correct.

Rationale: Of the listed options, the best description of the AWS Well-Architected Tool is that it provides guidelines to help create the architecture for your software product.

The correct answer is: It provides guidelines to help create the architecture for your software product

Question **2**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

Monolithic design is the basis for decoupled architecture in cloud.

Select one:

True

False Correct

Feedback

Rationale: Monolithic design is not the basis for the decoupled architecture in cloud.

The correct answer is 'False'.

Question **3**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

Which of the following is not a characteristic of cloud native design?

Select one:

a. APIs

b. Microservices

c. Decoupled architecture

d. Monolithic design Correct

Feedback

Your answer is correct.

Rationale: Of the listed options, the characteristics of a cloud native design include APIs, Elasticity, and Microservices. Monolithic design is not included as a characteristic of a cloud native design.

The correct answer is: Monolithic design

Question **4**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

Selecting a service to host your database in cloud is based on which of the following?

Select one:

a. Monolithic-design architecture

b. Purpose-driven architecture Correct

c. Service provider-driven architecture

d. Event-driven architecture

Feedback

Your answer is correct.

Rationale: Of the listed options, selecting a service to host your database in cloud is based on the purpose-driven architecture.

The correct answer is: Purpose-driven architecture

Question **5**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

A multi-tier architecture can be viewed as three layers: the presentation layer, the logic layer, and the data layer. Which of the following is true for this architecture?

Select one:

a. All the three layers have direct access to the internet

b. The logic layer and the data layer have direct access to the internet

c. Only the presentation layer has direct access to the internet Correct

d. The presentation layer and the logic layer have direct access to the internet

Feedback

Your answer is correct.

Rationale: In a multi-tier architecture, only the presentation layer has direct access to the internet.

The correct answer is: Only the presentation layer has direct access to the internet

Bottom of Form

Managing multiple AWS accounts

After completing this episode, you should be able to:

* Deploy access controls across multiple Amazon Web Services (AWS) accounts.
* Explain the roles of Control Tower and service control policies (SCPs) in managing account access from other accounts.
* Design role-based access control (RBAC) across accounts.

Key point 1

AWS Organizations is free to use, and the Organizations management account is charged for resources used by users and roles within those accounts. You can create new users within the account, or invite existing accounts to join the Organization.

Key point 2

SCPs filter, or remove/limit, permissions, never add permissions. Identity-based or resource-based policies are still required to grant permissions to resources or services. Effective permissions exist in the overlap between SCP restrictions and identity or resource policy permissions. SCP effects are inherited by everything below where the SCP is applied.

Key point 3

AWS IAM Identity Center provides access controls for organization accounts. It previously was called *AWS SSO*. AWS Control Tower incorporates functions from Organizations, Identity Center, CloudFormation, CloudTrail, and other services into a single service in addition to more robust features. Control Tower includes the following components: Landing Zone, Account Factory, Guardrails, and Dashboard.

Key point 4

AWS Resource Access Manager (RAM) allows users to share resources across accounts, organizations, organizational units (OUs), and in some cases, Identity Access Management (IAM) users or roles. AWS License Manager is used for managing software licenses and to enforce licensing rules to reflect the terms of the company's agreement with the vendor.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* Service Control Policies:
  + Service control policies (SCPs): <https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_scps>
  + SCP Syntax: <https://docs.aws.amazon.com/organizations/latest/userguide/orgs_manage_policies_scps_syntax>
* AWS IAM Identity Center Features: <https://aws.amazon.com/iam/identity-center/features>
* AWS Control Tower Features: <https://aws.amazon.com/controltower/features>
* AWS Service Catalog: <https://aws.amazon.com/servicecatalog>
* AWS Resource Access Manager (RAM): <https://aws.amazon.com/ram>
* AWS License Manager: <https://aws.amazon.com/license-manager>

Identity concepts

After completing this episode, you should be able to:

* Explain the importance of the AWS shared responsibility model in securing cloud resources.
* Describe the principle of least privilege.
* Compare various AWS access types.
* Compare resource policies, IAM roles, and IAM policies.

Key point 1

Security in Amazon Web Services (AWS) is structured by the shared responsibility model, in which AWS is responsible for some layers of security while the customer is responsible for the other layers. The dividing line of responsibility varies by the service used.

Key point 2

Cloud security works in layers, with various security tools working together to provide thorough coverage of all resources and services. Security layers are applied to resources, network spaces, and access controls.

Key point 3

AWS resources can be accessed through the Management Console, the command-line interface (CLI), CloudShell, and various software developer kit (SDK) tools. You can use identity federation to provide authentication through other identity providers.

Key point 4

The CLI and SDKs use AWS Service Endpoints to direct application programming interface (API) messaging.

Key point 5

Users represent humans or services with access to your AWS account. Users can be collected in groups for easier permission management. You can't "sign into" a group. Roles can define temporary permissions and are the best-practice method of giving permissions to resources and, in some cases, to users. You can use policies to group permissions for assignment. Resource-based policies apply to specific resources, while identity-based policies apply to users, groups, and roles.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* AWS service endpoints: <https://docs.aws.amazon.com/general/latest/gr/rande>
* AWS Identity and Access Management: <https://aws.amazon.com/iam>
* What is IAM? https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction
* IAM roles: <https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles>
* Policies and permissions in IAM: <https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies>
* Internetwork traffic privacy in Amazon VPC: <https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Security>

Securing AWS accounts

After completing this episode, you should be able to:

* Explain federated access and single sign-on (SSO).
* Compare root users and Identity Access Management (IAM) users.
* Configure multi-factor authentication (MFA) requirements.
* Design an authorization model that includes IAM users, groups, roles, and policies.

Key point 1

Best practice is to create an IAM user with full administrative access and other user accounts with more limited access, and then never again use the root account for anything. Never create programmatic access keys for the root account.

Key point 2

Always enable MFA on all Amazon Web Services (AWS) accounts.

Key point 3

Within an IAM role, the *trust policy* defines who can use the role while the *permissions policy* defines what they can do with the role.

Key point 4

You can create up to 5,000 users within your account. These users represent specific entities and require long-term access. In contrast, a role represents unspecified entities and provides short-term access or emergency access. Roles rely on Security Token Service (STS)) for authorization. Identity providers can be used through Web Identity Federation (OIDC), SAML 2.0, or Amazon Cognito.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* AWS security credentials: <https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html#access-keys-and-secret-access-keys>
* Setting up Active Directory to federate access to AMS IAM roles: <https://docs.aws.amazon.com/managedservices/latest/onboardingguide/set-up-ad-to-federate-iam-roles>
* AWS Directory Service:
  + What is AWS Directory Service? <https://docs.aws.amazon.com/directoryservice/latest/admin-guide/what_is>
  + AWS Directory Service: <https://aws.amazon.com/directoryservice>
* Identity providers and federation: <https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers>
* Amazon Cognito: <https://aws.amazon.com/cognito>

|  |  |
| --- | --- |
| **Started on** | Thursday, 17 July 2025, 8:07 AM |
| **State** | Finished |
| **Completed on** | Thursday, 17 July 2025, 8:07 AM |
| **Time taken** | 31 secs |
| **Grade** | **100.00** out of 100.00 |

Top of Form

Question **1**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

Which of the following are components of the URL of an AWS service endpoint?

Select one:

a. Protocol

b. Service-code

c. AWS region

d. amazonaws.com

e. All of the above Correct

Feedback

Your answer is correct.

Rationale: The URL of an AWS service endpoint includes all listed options, the prototcol, the service-code, the AWS region, and the phrase amazonaws.com.

The correct answer is: All of the above

Question **2**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

In the shared responsibility model with AWS, which of the following cloud deployment model gives the user the maximum responsibility?

Select one:

a. Infrastructure as a Service (IaaS) Correct

b. Platform as a Service (PaaS)

c. Software as a Service (SaaS)

d. All the models require same levels of responsibility

Feedback

Your answer is correct.

Rationale: Of the listed options, the Infrastructure as a Service (IaaS) cloud deployment model gives the user the maximum responsibility.

The correct answer is: Infrastructure as a Service (IaaS)

Question **3**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

A user account is configured such that it requires the user to type-in the first password to clear stage 1 authentication then the second password to clear stage 2 authentication. This is an example of multi-factor authentication (MFA).

Select one:

True

False Correct

Feedback

Rationale: A user account requiring the user to type in two passwords in two different authentication stages is not multi-factor authentication (MFA).

The correct answer is 'False'.

Question **4**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

When you log into a mobile app using your Facebook or Google login details, you are using which of the following types of Identity Provider (IdP)?

Select one:

a. Amazon Congnito

b. Security Assertion Markup Language 2.0 (SAML 2.0)

c. AWS Directory Service

d. Web Identity Federation Correct

Feedback

Your answer is correct.

Rationale: When you log into a mobile app using your Facebook or Google login details, you are using the Web Identity Federation type of Identity Provider (IdP).

The correct answer is: Web Identity Federation

Question **5**

Correct

Mark 20.00 out of 20.00

Flag question

Question text

Which of the following governs the effective permissions available to an AWS user account?

Select one:

a. Service Control Policies

b. Identity Policies

c. Resource Policies

d. An overlap of all three policies listed Correct

Feedback

Your answer is correct.

Rationale: The effective permissions on an AWS account are an overlap of all three: Service Control Policies, Identity Policies, and Resource Policies.

The correct answer is: An overlap of all three policies listed

Networking concepts

After completing this episode, you should be able to:

* Describe the open systems intercommunication (OSI) model.
* Compare regions and availability zones in the Amazon Web Services (AWS) global infrastructure.
* Identify commonly-used network protocols and their ports.
* Explain how route tables support routing in AWS.
* Describe basic load balancing and proxy concepts.

Key point 1

Network protocols function as various layers in the OSI model. Many of these protocols are identified by specific port numbers. Secure versions of protocols use different ports than their non-secure versions. For example, HTTP uses port 80 while HTTPS uses port 443.

Key point 2

Routing connects devices across networks. If a device doesn't know where to find a packet's destination device, the packet is sent to the default gateway for that network. Gateway devices (i.e., routers) use route tables to find destination networks.

Key point 3

AWS offers many regions throughout the world. Each region contains multiple Availability Zones (AZs). Edge Locations place AWS infrastructure closer to users.

Key point 4

Load balancers are used to distribute traffic across nodes in a cluster, which supports cloud scalability. Common load-balancing algorithms include round robin, least connections, least time, and weighted, which is a form of round robin.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

Bottom of Form

Network segmentation and addressing

After completing this episode, you should be able to:

* Explain the role of IP addressing in networks.
* Use subnet tiers to segment network spaces.
* Describe the organization of the AWS Solutions Architect exam objectives.
* Explain how Domain Name System (DNS) and Dynamic Host Configuration Protocol (DHCP) network services support IP addressing.

1. Describe network address translation (NAT).

Key point 1

Addressing at various layers of the Open Systems Interconnection (OSI) model include media access control (MAC) addresses at the Data Link layer, IP addresses at the Network layer, and Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) ports at the Transport layer. Public IP addresses can be used for communication on the Internet, while private IP addresses cannot. Private IP address ranges include 10.0.0.0 through 10.255.255.255, 172.16.0.0 through 172.31.255.255, and 192.168.0.0 through 192.168.255.255.

Key point 2

The DNS is used to resolve website addresses to IP addresses. In AWS, Route 53 can be used to provide this service. DHCP is used to assign IP addresses within a network space. DHCP configurations are provided by DHCP option sets in AWS, and DHCP option sets can also be used to assign DNS servers to instances based on the virtual private cloud (VPC).

Key point 3

Subnetting is the process of dividing IP address spaces for subnets in a network space. Internet gateways can be used to provide internet connectivity to public subnets, and NAT gateways can be used for internet connectivity to private subnets.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

VPC deep dive

After completing this episode, you should be able to:

* Design IP address spaces and subnets in Amazon Web Services (AWS).
* Manage route tables in virtual private clouds (VPCs) to control traffic into and out of network spaces.
* Design network topologies to support various system architectures.

Key point 1

Resources in a public subnet must have a route to an internet gateway and a public IP address to access the internet. The public IP address is assigned to an instance's virtual network interface, not within the operating system (OS). Resources in a private subnet must have a route to a network address translation (NAT) gateway, which should have its own public IP address and a route to an internet gateway for the private subnet resources to have access to the Internet.

Key point 2

Each VPC has a main route table that subnets can use. Every subnet must have exactly one route table associated with it. Each route table can be associated with multiple subnets. When evaluating traffic, the most specific route applies.

Key point 3

You can use VPC peering connections or a transit gateway to connect VPCs across regions. VPC peering is a one-to-one connection that isn't transitive. A transit gateway is more scalable to support multiple connections to VPCs and other networks.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* Amazon Virtual Private Cloud Documentation: <https://docs.aws.amazon.com/vpc>
* Configure route tables: <https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Route_Tables>
* AWS Transit Gateway: <https://aws.amazon.com/transit-gateway>
* Hybrid Cloud with AWS: <https://aws.amazon.com/hybrid>

External network connections

After completing this episode, you should be able to:

* Compare external network connection options for performance, security, and costs.
* Configure virtual private network (VPN) resources in Amazon Web Services (AWS).
* Explain options for AWS Direct Connect.
* Explain the benefits of AWS PrivateLink.
* Configure virtual private cloud (VPC) peering.

Key point 1

External network connection options include the internet, VPN connections, and Direct Connect. You can establish a VPN as a client VPN to a single user or as a site-to-site VPN, which is always on and connects to the on-premises network and not just to a single user. A VPN connection requires a virtual VPN gateway on the AWS side and a physical customer gateway on the customer network.

Key point 2

A Direct Connect connection is established through a colocation facility. To get a Direct Connect connection, you first configure the DX connection in your AWS account, download the LOA+CFA (letter of authorization + connecting facility assignment), and take that documentation to a colocation facility where you have a connection through your ISP or through a partner service. A Direct Connect connection is not inherently redundant.

Key point 3

VPC endpoints provide access to AWS services from within a VPC. Gateway endpoints are highly available and connect to S3 or DynamoDB. Interface endpoints are not inherently redundant and can be used to connect to many other AWS services using PrivateLink. PrivateLink can be used to keep your traffic within the AWS infrastructure and off the internet. It can also be used to connect to AWS resources owned by other customers.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* AWS Direct Connect: <https://aws.amazon.com/directconnect>
* AWS Direct Connect Resiliency Recommendations: <https://aws.amazon.com/directconnect/resiliency-recommendation/?nc=sn&loc=4&dn=2>
* PrivateLink:
  + What is AWS PrivateLink?: <https://docs.aws.amazon.com/vpc/latest/privatelink/what-is-privatelink>
  + Access AWS services through AWS PrivateLink: <https://docs.aws.amazon.com/vpc/latest/privatelink/privatelink-access-aws-services>
  + Control access to VPC endpoints using endpoint policies: <https://docs.aws.amazon.com/vpc/latest/privatelink/vpc-endpoints-access>

A deep dive on gateways

After completing this episode, you should be able to:

* Explain the purpose of network address translation (NAT) gateways.
* Compare NAT gateways and NAT instances.
* Explain the purpose of Transit Gateway.
* Describe the role of internet gateways in network segmentation.
* Identify virtual private network (VPN) gateway types.

Key point 1

Internet Gateways (IGWs) are inherently redundant and highly available. Create a route in a public subnet's route table to send internet traffic to the IGW. An IGW is automatically included in your default virtual private cloud (VPC) in a region. However, you'll have to create one for nondefault VPCs. With IPv4 traffic, the IGW also provides NAT services.

Key point 2

NAT gateways provide a route to the internet for resources in private subnets while protecting those resources from incoming traffic. A NAT gateway resides in an Availability Zone is a single point of failure. For IPv6 traffic, an egress-only IGW can support outgoing traffic only to provide protected access from private subnet resources to the internet in lieu of a NAT gateway.

Key point 3

A Transit Gateway can connect VPCs and on-prem networks through VPNs or Direct Connect. It can also peer with another Transit Gateway. Routes propagate across attachments through Border Gateway Protocol (BGP). You only need one Transit Gateway per region.

Key point 4

For VPNs, a virtual private gateway is attached to a VPC and requires a private autonomous system number (ASN). A customer gateway resource is also created in AWS to represent the physical device in the customer's on-premises network.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* NAT gateways: <https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-gateway>
* NAT instances: <https://docs.aws.amazon.com/vpc/latest/userguide/VPC_NAT_Instance>
* Transit Gateway:
  + What is a transit gateway? <https://docs.aws.amazon.com/vpc/latest/tgw/what-is-transit-gateway>
  + Hybrid Cloud with AWS: <https://aws.amazon.com/hybrid>
* How AWS Site-to-Site VPN works: <https://docs.aws.amazon.com/vpn/latest/s2svpn/how_it_works>

Route 53 deep dive

After completing this episode, you should be able to:

* Explain how Domain Name System (DNS) works.
* Compare DNS record types.
* Configure Amazon Route 53 to support AWS-hosted DNS records.

Key point 1

You can use Route 53 to register a domain, create hosted zones and DNS records, set health checks and failover, and set a DNS resolver for your virtual private clouds (VPCs). Hosted zones can be public for routing internet traffic, private for routing VPC traffic, or split-view/split-horizon for routing both public and private traffic. To set up a split horizon hosted zone, create both a public zone and a private zone with the same name for both, then associate the private zone with at least one VPC.

Key point 2

Common types of DNS records include A, AAAA, Canonical Name (CNAME), mail exchange (MX), pointer record (PTR), and start of authority (SOA). In Amazon Web Services (AWS), you often need to use an alias DNS record in place of a CNAME record because a CNAME record can't be posted to the zone apex. Alias records are the AWS solution to this limitation.

Key point 3

Commonly used Route 53 routing policies include simple routing, failover routing, geolocation routing, geoproximity routing, latency routing, IP-based routing, multivalue answer routing, and weighted routing. You can use health checks to check endpoint health or check group resources, CloudWatch data stream status, and recover status after failover.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* What is Amazon Route 53? <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/Welcome>
* Routing traffic to your resources: <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/dns-routing-traffic-to-resources>
* Choosing between alias and non-alias records: <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resource-record-sets-choosing-alias-non-alias>
* Solving DNS zone apex challenges with third-party DNS providers using AWS: <https://aws.amazon.com/blogs/networking-and-content-delivery/solving-dns-zone-apex-challenges-with-third-party-dns-providers-using-aws>
* Choosing a routing policy: <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/routing-policy>
* Creating Amazon Route 53 health checks and configuring DNS failover: <https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/dns-failover>

Network security

After completing this episode, you should be able to:

* Identify common threat vectors external to Amazon Web Services (AWS).
* Configure network traffic controls for protocols and ports.
* Compare AWS services used to ensure network traffic security.
* Integrate security into virtual private cloud (VPC) design.

Key point 1

Network access control lists (ACLs) provide subnet-level protection while security groups provide instance-level protection. NACLs are stateless and include allow or deny rules, where the highest priority match is applied to traffic. Security groups are stateful and include only allow rules with an implicit deny. All rules are evaluated to determine how to filter traffic.

Key point 2

Network Firewall provides stateful and stateless firewall options, which also include intrusion protection system (IPS) features. Network Firewall can be distributed to work within a VPC or centralized to work through Transit Gateway.

Key point 3

AWS Shield provides built-in distributed denial-of-service (DDoS) protection. The Standard version is free to all users and is always on. The Advanced version must be configured by the user, adds layer-7 protection, and includes cost protection, web application firewall (WAF) integration, real-time visibility, and health-based detection. Shield Advanced costs $3,000/month/organization plus data transfer and usage fees.

Key point 4

WAFv2 provides web ACLs to block common web-based exploits, such as SQL injection, XSS attacks, and HTTP flood. The WAF stands between the internet and web app resources such as CloudFront (global protection), Application load balancers (regional protection), and API Gateway (regional protection).

Key point 5

AWS Firewall Manager provides management oversight to WAF, Shield, security groups, Network Firewall, Domain Name System (DNS) Firewall, and Palo Alto next-generation firewalls (NGFWs) to handle policies across accounts within an Organization and organize policies hierarchically.

Key point 6

Amazon GuardDuty examines logs from S3, CloudTrail, DNS, Elastic Block Storage (EBS) volumes, Amazon Elastic Kubernetes Service (Amazon EKS), and VPC. It provides threat intelligence to identify privilege escalation, exposed credentials, malicious IP addresses, and malware.

Key point 7

Amazon Inspector continuously scans new Amazon Elastic Compute Cloud (EC2) instances and containers using eligible images from ECR for vulnerabilities. For network assessment, no agent is needed. For host assessment, the instance must be running the AWS Systems Manager Agent (SSM Agent).

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* Internetwork traffic privacy in Amazon VPC: https://docs.aws.amazon.com/vpc/latest/userguide/VPC\_Security
* AWS Network Firewall: <https://aws.amazon.com/network-firewall>
* AWS Shield: <https://aws.amazon.com/shield>
* Security Automations for AWS WAF: <https://aws.amazon.com/solutions/implementations/aws-waf-security-automations>
* AWS Firewall Manager: <https://aws.amazon.com/firewall-manager>
* AWS Firewall Manager: <https://docs.aws.amazon.com/waf/latest/developerguide/fms-chapter.html>
* What is Amazon GuardDuty? <https://docs.aws.amazon.com/guardduty/latest/ug/what-is-guardduty>
* Amazon Inspector: <https://aws.amazon.com/inspector/?nc=sn&loc=1>
* CVE: [https://cve.mitre.org](https://cve.mitre.org/)

Network resiliency and scalability

After completing this episode, you should be able to:

* Explain basic disaster recovery (DR) and business continuity (BC) concepts.
* Describe DR strategies in Amazon Web Services (AWS).
* Use AWS regions and availability zones to ensure high availability (HA) of resources and connections.
* Explain how to incorporate automation in designing resilient and scalable infrastructure.

Key point 1

Services designed to support network scalability include Transit Gateway (up to 5,000 attachments), load balancing (direct traffic to auto-scaling groups), CloudFront (distribute content close to point of use), and thoughtful IP address space design that avoids overlap in Classless Inter-Domain Routing (CIDR) blocks for subnets and virtual private clouds (VPCs).

Key point 2

Eliminate single points of failure by using multiple Availability Zones and multiple regions, where appropriate. A good compromise might be to back up data across regions but keep functionality in one region.

Key point 3

Failover strategies include backup and restore, pilot light, warm standby, and multi-site active/active. Automated recovery tools in AWS include AWS Systems Manager (manage applications and infrastructure), AWS Config (assess, audit, and evaluate resource configurations), and AWS CloudFormation (automate deployment through Infrastructure as Code).

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* Scalability: <https://docs.aws.amazon.com/whitepapers/latest/hybrid-connectivity/scalability>
* Disaster Recovery of Workloads on AWS: Recovery in the Cloud: <https://docs.aws.amazon.com/whitepapers/latest/disaster-recovery-workloads-on-aws/disaster-recovery-workloads-on-aws>
* Business Continuity Plan (BCP): <https://docs.aws.amazon.com/whitepapers/latest/disaster-recovery-workloads-on-aws/business-continuity-plan-bcp>
* AWS Systems Manager Automation: <https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-automation>
* AWS Config: <https://aws.amazon.com/config>
* AWS CloudFormation: <https://aws.amazon.com/cloudformation>
* Using redundant Site-to-Site VPN connections to provide failover: <https://docs.aws.amazon.com/vpn/latest/s2svpn/vpn-redundant-connection>

Load balancing

After completing this episode, you should be able to:

* Explain how load balancing works.
* Compare Amazon Web Services (AWS) load balancing services and configuration options.
* Choose an appropriate load-balancing strategy for a given scenario.

Key point 1

Elastic Load Balancers can be Application Load Balancers, Network Load Balancers, or Gateway Load Balancers. Network Load Balancers function at layer 4, target IP addresses/instances/other load balancers, and provide high-speed performance.

Key point 2

Application Load Balancers function at layer 7, can route to specific and multiple ports on a target, can route to Lambda, and can handle user authentication processes.

Key point 3

Gateway Load Balancers direct traffic to a virtual private cloud (VPC) endpoint. It's a transparent device, sometimes called a *bump in the line*.

Key point 4

Load Balancer architecture includes listeners, rules, actions, target groups, and health checks.

Additional resources

If additional resources are used during the episode, they can be obtained using the download link on the overview episode (e.g. diagrams, no PowerPoints).

External resources

You can reference the following external resources for supplementary tools and information:

* Load Balancer types:
  + Elastic Load Balancing: <https://aws.amazon.com/elasticloadbalancing>
  + Elastic Load Balancing features: <https://aws.amazon.com/elasticloadbalancing/features>
  + Data protection in Elastic Load Balancing: <https://docs.aws.amazon.com/elasticloadbalancing/latest/userguide/data-protection>
  + Elastic Load Balancing pricing: <https://aws.amazon.com/elasticloadbalancing/pricing>