Chapter 12
Security on AWS

THE AWS CERTIFIED SOLUTIONS ARCHITECT EXAM TOPICS COVERED IN THIS CHAPTER MAY INCLUDE, BUT ARE NOT LIMITED TO, THE FOLLOWING:

Domain 3.0: Data Security

✓ **3.1 Recognize and implement secure practices for optimum cloud deployment and maintenance.**

Content may include the following:

- AWS shared responsibility model
- AWS platform compliance
- AWS security attributes (customer workloads down to physical layer)
- AWS administration and security services
- AWS Identity and Access Management (IAM)
- Amazon Virtual Private Cloud (Amazon VPC)
- AWS CloudTrail
- Ingress vs. egress filtering, and which AWS services and features fit
- Core Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Simple Storage Service (Amazon S3) security feature sets
- Incorporating common conventional security products (Firewall, Virtual Private Network [VPN])
- Denial of Service (DoS) mitigation
- Encryption solutions (e.g., key services)
- Complex access controls (building sophisticated security groups, Access Control Lists [ACLs], etc.)

Introduction

Cloud security is the first priority at AWS. All AWS customers benefit from a data center and network architecture that is built to satisfy the requirements of the most security-sensitive organizations. AWS and its partners offer tools and features to help you meet your security objectives around visibility, auditability, controllability, and agility. This means that you can have the security you need, but without the capital outlay and at a much lower operational overhead than in an on-premises or a traditional data center environment. This chapter will cover the relevant security topics that are within scope of the AWS Certified Solutions Architect – Associate exam.

Shared Responsibility Model

Before we go into the details of how AWS secures its resources, we should talk about how security in the cloud is slightly different than security in your on-premises data centers. When you move computer systems and data to the cloud, security responsibilities become shared between you and your cloud service provider. In this case, AWS is responsible for securing the underlying infrastructure that supports the cloud, and you're responsible for anything you put on the cloud or connect to the cloud. This shared responsibility model can reduce your operational burden in many ways, and in some cases it may even improve your default security posture without additional action on your part. Figure 12.1 illustrates AWS responsibilities versus those of the customer. Essentially, AWS is responsible for security *of* the cloud, and customers are responsible for security *in* the cloud.

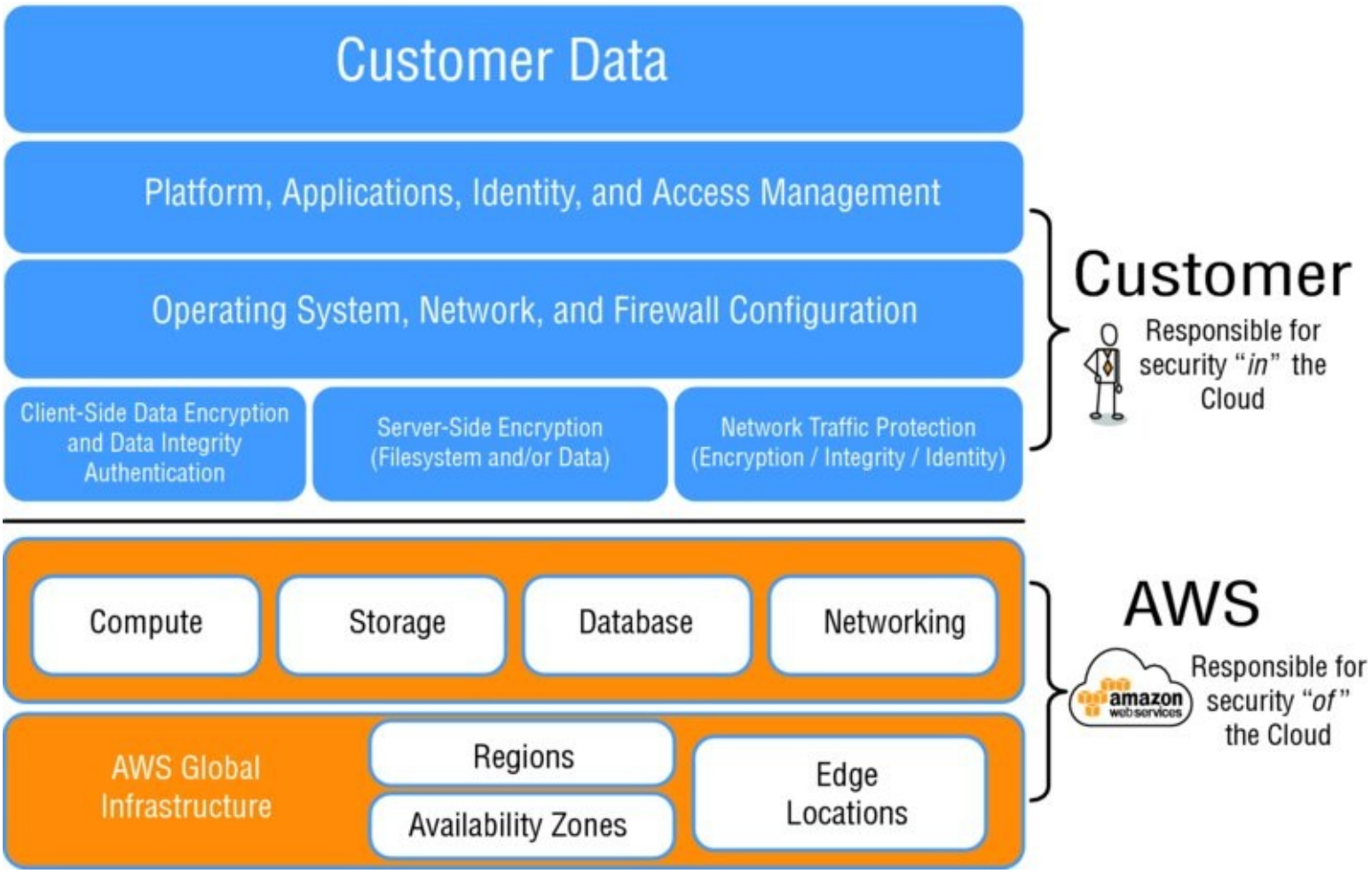

FIGURE 12.1 The shared responsibility model

AWS Compliance Program

AWS compliance enables customers to understand the robust controls in place at AWS to maintain security and data protection in the cloud. As you build systems on top of AWS Cloud infrastructure, you share compliance responsibilities with AWS. By tying together governance-focused, audit-friendly service features with applicable compliance or audit standards, AWS compliance enablers build on traditional programs, helping you to establish and operate in an AWS security control environment. The IT infrastructure that AWS provides is designed and managed in alignment with security best practices and a variety of IT security standards, including (at the time of this writing):

- Service Organization Control (SOC) 1/Statement on Standards for Attestation Engagements (SSAE)16/International Standards for Assurance Engagements No. 3402 (ISAE) 3402 (formerly Statement on Auditing Standards [SAS] 70)
- SOC 2
- SOC 3
- Federal Information Security Management Act (FISMA), Department of Defense (DoD) Information Assurance Certification and Accreditation Process (DIACAP), and Federal Risk and Authorization Management Program (FedRAMP)
- DoD Cloud Computing Security Requirements Guide (SRG) Levels 2 and 4
- Payment Card Industry Data Security Standard (PCI DSS) Level 1
- International Organization for Standardization (ISO) 9001 and ISO 27001
- International Traffic in Arms Regulations (ITAR)
- Federal Information Processing Standard (FIPS) 140-2

In addition, the flexibility and control that the AWS platform provides allows customers to deploy solutions that meet several industry-specific standards, including:

- Criminal Justice Information Services (CJIS)
- Cloud Security Alliance (CSA)
- Family Educational Rights and Privacy Act (FERPA)
- Health Insurance Portability and Accountability Act (HIPAA)
- Motion Picture Association of America (MPAA)

AWS provides a wide range of information regarding its IT control environment to customers through whitepapers, reports, certifications, accreditations, and other third-party attestations. To aid in preparation for your AWS Certified Solutions Architect Associate exam, see Chapter 13, “AWS Risk and Compliance.” More information is available in the “AWS Risk and Compliance” whitepaper available on the AWS website.

AWS Global Infrastructure Security

AWS operates the global cloud infrastructure that you use to provision a variety of basic computing resources such as processing and storage. The AWS global infrastructure includes the facilities, network, hardware, and operational software (for example, host operating system and virtualization software) that support the provisioning and use of these resources. The AWS global infrastructure is designed and managed according to security best practices as well as a variety of security compliance standards. As an AWS customer, you can be assured that you're building web architectures on top of some of the most secure computing infrastructure in the world.

Physical and Environmental Security

AWS data centers are state of the art, using innovative architectural and engineering approaches. Amazon has many years of experience in designing, constructing, and operating large-scale data centers. This experience has been applied to the AWS platform and infrastructure. AWS data centers are housed in nondescript facilities. Physical access is strictly controlled both at the perimeter and at building ingress points by professional security staff using video surveillance, intrusion detection systems, and other electronic means. Authorized staff must pass two-factor authentication a minimum of two times to access data center floors. All visitors and contractors are required to present identification and are signed in and continually escorted by authorized staff.

AWS only provides data center access and information to employees and contractors who have a legitimate business need for such privileges. When an employee no longer has a business need for these privileges, his or her access is immediately revoked, even if they continue to be an employee of Amazon or AWS. All physical access to data centers by AWS employees is logged and audited routinely.

Fire Detection and Suppression

AWS data centers have automatic fire detection and suppression equipment to reduce risk. The fire detection system uses smoke detection sensors in all data center environments, mechanical and electrical infrastructure spaces, chiller rooms and generator equipment rooms. These areas are protected by wet-pipe, double-interlocked pre-action, or gaseous sprinkler systems.

Power

AWS data center electrical power systems are designed to be fully redundant and maintainable without impact to operations, 24 hours a day, and 7 days a week. Uninterruptible Power Supply (UPS) units provide backup power in the event of an electrical failure for critical and essential loads in the facility. AWS data centers use generators to provide backup power for the entire facility.

Climate and Temperature

Climate control is required to maintain a constant operating temperature for servers and other hardware, which prevents overheating and reduces the possibility of service outages.

AWS data centers are built to maintain atmospheric conditions at optimal levels. Personnel and systems monitor and control temperature and humidity at appropriate levels.

Management

AWS monitors electrical, mechanical, and life support systems and equipment so that any issues are immediately identified. AWS staff performs preventative maintenance to maintain the continued operability of equipment.

Storage Device Decommissioning

When a storage device has reached the end of its useful life, AWS procedures include a decommissioning process that is designed to prevent customer data from being exposed to unauthorized individuals.

Business Continuity Management

Amazon's infrastructure has a high level of availability and provides customers with the features to deploy a resilient IT architecture. AWS has designed its systems to tolerate system or hardware failures with minimal customer impact. Data center Business Continuity Management at AWS is under the direction of the Amazon Infrastructure Group.

Availability

Data centers are built in clusters in various global regions. All data centers are online and serving customers; no data center is "cold." In case of failure, automated processes move data traffic away from the affected area. Core applications are deployed in an N+1 configuration, so that in the event of a data center failure, there is sufficient capacity to enable traffic to be load-balanced to the remaining sites.

AWS provides its customers with the flexibility to place instances and store data within multiple geographic regions and also across multiple Availability Zones within each region. Each Availability Zone is designed as an independent failure zone. This means that Availability Zones are physically separated within a typical metropolitan region and are located in lower risk flood plains (specific flood zone categorization varies by region). In addition to having discrete UPS and on-site backup generation facilities, they are each fed via different grids from independent utilities to further reduce single points of failure. Availability Zones are all redundantly connected to multiple tier-1 transit providers. Figure 12.2 illustrates how AWS regions are comprised of Availability Zones.

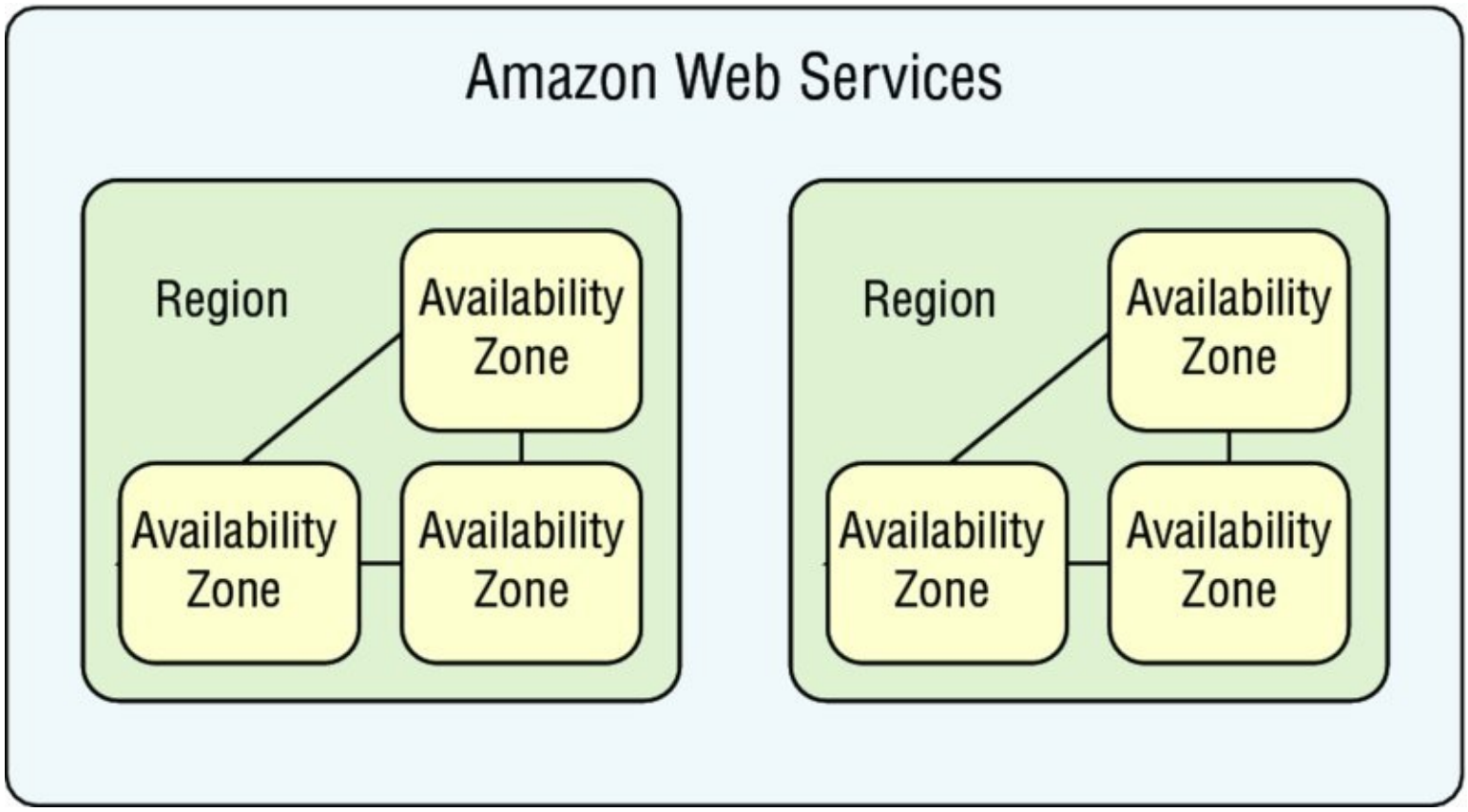

FIGURE 12.2 Amazon Web Services regions


You should architect your AWS usage to take advantage of multiple regions and Availability Zones. Distributing applications across multiple Availability Zones provides the ability to remain resilient in the face of most failure modes, including natural disasters or system failures.

Incident Response

The Amazon Incident Management team employs industry-standard diagnostic procedures to drive resolution during business-impacting events. Staff operators provide 24 × 7 × 365 coverage to detect incidents and to manage the impact and resolution.

Communication

AWS has implemented various methods of internal communication at a global level to help employees understand their individual roles and responsibilities and to communicate significant events in a timely manner. These methods include orientation and training programs for newly hired employees, regular management meetings for updates on business performance and other matters, and electronics means such as video conferencing, electronic mail messages, and the posting of information via the Amazon intranet.

AWS has also implemented various methods of external communication to support its customer base and the community. Mechanisms are in place to allow the customer support team to be notified of operational issues that impact the customer experience. A Service Health Dashboard is available and maintained by the customer support team to alert customers to any issues that may be of broad impact. The AWS Security Center is available to

provide you with security and compliance details about AWS. Customers can also subscribe to AWS Support offerings that include direct communication with the customer support team and proactive alerts to any customer-impacting issues.

Network Security

The AWS network has been architected to permit you to select the level of security and resiliency appropriate for your workload. To enable you to build geographically dispersed, fault-tolerant web architectures with cloud resources, AWS has implemented a world-class network infrastructure that is carefully monitored and managed.

Secure Network Architecture

Network devices, including firewall and other boundary devices, are in place to monitor and control communications at the external boundary of the network and at key internal boundaries within the network. These boundary devices employ rule sets, *access control lists (ACLs)*, and configurations to enforce the flow of information to specific information system services.

ACLs, or traffic flow policies, are established on each managed interface, which manage and enforce the flow of traffic. ACL policies are approved by Amazon Information Security. These policies are automatically pushed to ensure these managed interfaces enforce the most up-to-date ACLs.

Secure Access Points

AWS has strategically placed a limited number of access points to the cloud to allow for a more comprehensive monitoring of inbound and outbound communications and network traffic. These customer access points are called Application Programming Interface (API) endpoints, and they permit secure HTTP access (HTTPS), which allows you to establish a secure communication session with your storage or compute instances within AWS. To support customers with Federal Information Processing Standard (FIPS) cryptographic requirements, the Secure Sockets Layer (SSL)-terminating load balancers in AWS GovCloud (US) are FIPS 140-2 compliant.

In addition, AWS has implemented network devices that are dedicated to managing interfacing communications with Internet Service Providers (ISPs). AWS employs a redundant connection to more than one communication service at each Internet-facing edge of the AWS network. These connections each have dedicated network devices.

Transmission Protection

You can connect to an AWS access point via HTTP or HTTPS using SSL, a cryptographic protocol that is designed to protect against eavesdropping, tampering, and message forgery. For customers who require additional layers of network security, AWS offers the Amazon Virtual Private Cloud (Amazon VPC) (as referenced in Chapter 4, “Amazon Virtual Private Cloud (Amazon VPC),” which provides a private subnet within the AWS Cloud and the ability to use an IPsec Virtual Private Network (VPN) device to provide an encrypted tunnel between the Amazon VPC and your data center.

Network Monitoring and Protection

The AWS network provides significant protection against traditional network security issues, and you can implement further protection. The following are a few examples:

Distributed Denial of Service (DDoS) Attacks AWS API endpoints are hosted on a large, Internet-scale, world-class infrastructure that benefits from the same engineering expertise that has built Amazon into the world's largest online retailer. Proprietary DDoS mitigation techniques are used. Additionally, AWS networks are multi-homed across a number of providers to achieve Internet access diversity.

Man in the Middle (MITM) Attacks All of the AWS APIs are available via SSL-protected endpoints that provide server authentication. Amazon Elastic Compute Cloud (Amazon EC2) AMIs automatically generate new Secure Shell (SSH) host certificates on first boot and log them to the instance's console. You can then use the secure APIs to call the console and access the host certificates before logging into the instance for the first time. AWS encourages you to use SSL for all of your interactions.

IP Spoofing Amazon EC2 instances cannot send spoofed network traffic. The AWS-controlled, host-based firewall infrastructure will not permit an instance to send traffic with a source IP or Machine Access Control (MAC) address other than its own.

Port Scanning Unauthorized port scans by Amazon EC2 customers are a violation of the AWS Acceptable Use Policy. Violations of the AWS Acceptable Use Policy are taken seriously, and every reported violation is investigated. Customers can report suspected abuse via the contacts available on the AWS website. When unauthorized port scanning is detected by AWS, it is stopped and blocked. Port scans of Amazon EC2 instances are generally ineffective because, by default, all inbound ports on Amazon EC2 instances are closed and are only opened by the customer. Strict management of security groups can further mitigate the threat of port scans. If you configure the security group to allow traffic from any source to a specific port, that specific port will be vulnerable to a port scan. In these cases, you must use appropriate security measures to protect listening services that may be essential to their application from being discovered by an unauthorized port scan. For example, a web server must clearly have port 80 (HTTP) open to the world, and the administrator of this server is responsible for the security of the HTTP server software, such as Apache. You may request permission to conduct vulnerability scans as required to meet your specific compliance requirements. These scans must be limited to your own instances and must not violate the AWS Acceptable Use Policy. Advanced approval for these types of scans can be initiated by submitting a request via the AWS website.

Packet Sniffing by Other Tenants While you can place your interfaces into promiscuous mode, the hypervisor will not deliver any traffic to them that is not addressed to them. Even two virtual instances that are owned by the same customer located on the same physical host cannot listen to each other's traffic. While Amazon EC2 does provide ample protection against one customer inadvertently or maliciously attempting to view another customer's data, as a standard practice you should *encrypt* sensitive traffic.

It is not possible for a virtual instance running in promiscuous mode to receive or "sniff" traffic that is intended for a different virtual instance.

 Attacks such as Address Resolution Protocol (ARP) cache poisoning do not work within Amazon EC2 and Amazon VPC.

AWS Account Security Features

AWS provides a variety of tools and features that you can use to keep your *AWS account* and resources safe from unauthorized use. This includes *credentials* for access control, HTTPS endpoints for encrypted data transmission, the creation of separate AWS Identity and Access Management (IAM) user accounts, and user activity logging for security monitoring. You can take advantage of all of these security tools no matter which AWS services you select.

AWS Credentials

To help ensure that only authorized users and processes access your AWS account and resources, AWS uses several types of credentials for *authentication*. These include passwords, cryptographic keys, digital signatures, and certificates. AWS also provides the option of requiring *Multi-Factor Authentication (MFA)* to log in to your AWS Account or *IAM user* accounts. Table 12.1 highlights the various AWS credentials and their uses.

TABLE 12.1 AWS Credentials

Credential Type	Use	Description
Passwords	AWS root account or IAM user account login to the AWS Management Console	A string of characters used to log in to your AWS account or IAM account. AWS passwords must be a minimum of 6 characters and may be up to 128 characters.
Multi-Factor Authentication (MFA)	AWS root account or IAM user account login to the AWS Management Console	A six-digit, single-use code that is required in addition to your password to log in to your AWS account or IAM user account.
Access Keys	Digitally-signed requests to AWS APIs (using the AWS Software Development Kit [SDK], Command Line Interface [CLI], or REST/Query APIs)	Includes an access key ID and a secret access key. You use access keys to sign programmatic requests digitally that you make to AWS.
Key Pairs	SSH login to Amazon EC2 instances Amazon CloudFront-signed URLs	A key pair is required to connect to an Amazon EC2 instance launched from a public AMI. The keys that Amazon EC2 uses are 1024-bit SSH-2 RSA keys. You can have a key pair generated automatically for you when you launch the instance, or you can upload your own.
X.509 Certificates	Digitally signed SOAP requests to AWS APIs SSL server certificates for HTTPS	X.509 certificates are only used to sign SOAP-based requests (currently used only with Amazon Simple Storage Service [Amazon S3]). You can have AWS create an X.509 certificate and private key that you can download, or you can upload your own certificate by using the Security Credentials page.

For security reasons, if your credentials have been lost or forgotten, you cannot recover them or re-download them. However, you can create new credentials and then disable or delete the old set of credentials. In fact, AWS recommends that you change (rotate) your *access keys* and certificates on a regular basis. To help you do this without potential impact to your application's availability, AWS supports multiple concurrent access keys and certificates. With this feature, you can rotate keys and certificates into and out of operation on a regular basis without any downtime to your application. This can help to mitigate risk from lost or compromised access keys or certificates.

The AWS IAM API enables you to rotate the access keys of your AWS account and also for IAM user accounts.

Passwords

Passwords are required to access your AWS Account, individual IAM user accounts, AWS

Discussion Forums, and the AWS Support Center. You specify the password when you first create the account, and you can change it at any time by going to the Security Credentials page. AWS passwords can be up to 128 characters long and contain special characters, giving you the ability to create very strong passwords.

You can set a password policy for your IAM user accounts to ensure that strong passwords are used and that they are changed often. A password policy is a set of rules that define the type of password an IAM user can set.

AWS Multi-Factor Authentication (AWS MFA)

AWS MFA is an additional layer of security for accessing AWS Cloud services. When you enable this optional feature, you will need to provide a six-digit, single-use code in addition to your standard user name and password credentials before access is granted to your AWS account settings or AWS Cloud services and resources. You get this single-use code from an authentication device that you keep in your physical possession. This is MFA because more than one authentication factor is checked before access is granted: a password (something you know) and the precise code from your authentication device (something you have). You can enable MFA devices for your AWS account and for the users you have created under your AWS account with AWS IAM. In addition, you can add MFA protection for access across AWS accounts, for when you want to allow a user you've created under one AWS account to use an *IAM role* to access resources under another AWS account. You can require the user to use MFA before assuming the role as an additional layer of security.

AWS MFA supports the use of both hardware tokens and virtual MFA devices. Virtual MFA devices use the same protocols as the physical MFA devices, but can run on any mobile hardware device, including a smart phone. A virtual MFA device uses a software application that generates six-digit authentication codes that are compatible with the Time-Based One-Time Password (TOTP) standard, as described in RFC 6238. Most virtual MFA applications allow you to host more than one virtual MFA device, which makes them more convenient than hardware MFA devices. However, you should be aware that because a virtual MFA may be run on a less secure device such as a smart phone, a virtual MFA might not provide the same level of security as a hardware MFA device.

You can also enforce MFA authentication for AWS Cloud service APIs in order to provide an extra layer of protection over powerful or privileged actions such as terminating Amazon EC2 instances or reading sensitive data stored in Amazon S3. You do this by adding an MFA requirement to an IAM access policy. You can attach these access policies to IAM users, *IAM groups*, or resources that support ACLs like Amazon S3 buckets, Amazon Simple Queue Service (Amazon SQS) queues, and Amazon Simple Notification Service (Amazon SNS) topics.

Access Keys

Access keys are created by AWS IAM and delivered as a pair: the *Access Key ID (AKI)* and the *Secret Access Key (SAK)*. AWS requires that all API requests be signed by the SAK; that is, they must include a digital signature that AWS can use to verify the identity of the requestor. You calculate the digital signature using a cryptographic hash function. If you use any of the AWS SDKs to generate requests, the digital signature calculation is done for you.

Not only does the signing process help protect message integrity by preventing tampering with the request while it is in transit, but it also helps protect against potential replay attacks. A request must reach AWS within 15 minutes of the timestamp in the request. Otherwise, AWS denies the request.

The most recent version of the digital signature calculation process at the time of this writing is *Signature Version 4*, which calculates the signature using the *Hashed Message Authentication Mode (HMAC)*-Secure Hash Algorithm (SHA)-256 protocol. Version 4 provides an additional measure of protection over previous versions by requiring that you sign the message using a key that is derived from your SAK instead of using the SAK itself. In addition, you derive the signing key based on credential scope, which facilitates cryptographic isolation of the signing key.

Because access keys can be misused if they fall into the wrong hands, AWS encourages you to save them in a safe place and to not embed them in your code. For customers with large fleets of elastically scaling Amazon EC2 instances, the use of IAM roles can be a more secure and convenient way to manage the distribution of access keys.

IAM roles provide temporary credentials, which not only get automatically loaded to the target instance, but are also automatically rotated multiple times a day.

Amazon EC2 uses an Instance Profile as a container for an IAM role. When you create an IAM role using the AWS Management Console, the console creates an instance profile automatically and gives it the same name as the role to which it corresponds. If you use the AWS CLI, API, or an AWS SDK to create a role, you create the role and instance profile as separate actions, and you might give them different names. To launch an instance with an IAM role, you specify the name of its instance profile. When you launch an instance using the Amazon EC2 console, you can select a role to associate with the instance; however, the list that's displayed is actually a list of instance profile names.

Key pairs

Amazon EC2 supports RSA 2048 SSH keys for gaining first access to an Amazon EC2 instance. On a Linux instance, access is granted through showing possession of the SSH private key. On a Windows instance, access is granted by showing possession of the SSH private key in order to decrypt the administrator password. The public key is embedded in your instance, and you use the private key to sign in securely without a password. After you create your own AMIs, you can choose other mechanisms to log in to your new instances securely. You can have a *key pair* generated automatically for you when you launch the instance or you can upload your own. Save the private key in a safe place on your system and record the location where you saved it.

For Amazon CloudFront, you use key pairs to create signed URLs for private content, such as when you want to distribute restricted content that someone paid for. You create Amazon CloudFront key pairs by using the Security Credentials page. Amazon CloudFront key pairs can be created only by the root account and cannot be created by IAM users.

X.509 Certificates

X.509 certificates are used to sign SOAP-based requests. X.509 certificates contain a public key that is associated with a private key. When you create a request, you create a digital signature with your private key and then include that signature in the request, along with your certificate. AWS verifies that you're the sender by decrypting the signature with the public key that is in your certificate. AWS also verifies that the certificate that you sent matches the certificate that you uploaded to AWS.

For your AWS account, you can have AWS create an X.509 certificate and private key that you can download, or you can upload your own certificate by using the Security Credentials page. For IAM users, you must create the X.509 certificate (signing certificate) by using third-party software. In contrast to root account credentials, AWS cannot create an X.509 certificate for IAM users. After you create the certificate, you attach it to an IAM user by using IAM.

In addition to SOAP requests, X.509 certificates are used as SSL/*Transport Layer Security (TLS)* server certificates for customers who want to use HTTPS to encrypt their transmissions. To use them for HTTPS, you can use an open-source tool like OpenSSL to create a unique private key. You'll need the private key to create the Certificate Signing Request (CSR) that you submit to a Certificate Authority (CA) to obtain the server certificate. You'll then use the AWS CLI to upload the certificate, private key, and certificate chain to IAM.

You will also need an X.509 certificate to create a customized Linux AMI for Amazon EC2 instances. The certificate is only required to create an instance-backed AMI (as opposed to an Amazon Elastic Block Store [Amazon EBS]-backed AMI). You can have AWS create an X.509 certificate and private key that you can download, or you can upload your own certificate by using the Security Credentials page.

AWS CloudTrail

AWS CloudTrail is a web service that records API calls made on your account and delivers log files to your Amazon S3 bucket. AWS CloudTrail's benefit is visibility into account activity by recording API calls made on your account. AWS CloudTrail records the following information about each API call:

- The name of the API
- The identity of the caller
- The time of the API call
- The request parameters
- The response elements returned by the AWS Cloud service

This information helps you to track changes made to your AWS resources and to troubleshoot operational issues. AWS CloudTrail makes it easier to ensure compliance with internal policies and regulatory standards.

AWS CloudTrail supports log file integrity, which means you can prove to third parties (for example, auditors) that the log file sent by AWS CloudTrail has not been altered. Validated log files are invaluable in security and forensic investigations. This feature is built using

industry standard algorithms: SHA-256 for hashing and SHA-256 with RSA for digital signing. This makes it computationally unfeasible to modify, delete, or forge AWS CloudTrail log files without detection.

AWS Cloud Service-Specific Security

Not only is security built into every layer of the AWS infrastructure, but also into each of the services available on that infrastructure. AWS Cloud services are architected to work efficiently and securely with all AWS networks and platforms. Each service provides additional security features to enable you to protect sensitive data and applications.

Compute Services

AWS provides a variety of cloud-based computing services that include a wide selection of compute instances that can scale up and down automatically to meet the needs of your application or enterprise.

Amazon Elastic Compute Cloud (Amazon EC2) Security

Amazon EC2 is a key component in Amazon's Infrastructure as a Service (IaaS), providing resizable computing capacity using server instances in AWS data centers. Amazon EC2 is designed to make web-scale computing easier by enabling you to obtain and configure capacity with minimal friction. You create and launch instances, which are collections of platform hardware and software.

Multiple Levels of Security Security within Amazon EC2 is provided on multiple levels: the operating system (OS) of the host platform, the virtual instance OS or guest OS, a firewall, and signed API calls. Each of these items builds on the capabilities of the others. The goal is to prevent data contained within Amazon EC2 from being intercepted by unauthorized systems or users and to make Amazon EC2 instances themselves as secure as possible without sacrificing the flexibility in configuration that customers demand.

The Hypervisor Amazon EC2 currently uses a highly customized version of the Xen hypervisor, taking advantage of paravirtualization (in the case of Linux guests). Because paravirtualized guests rely on the hypervisor to provide support for operations that normally require privileged access, the guest OS has no elevated access to the CPU. The CPU provides four separate privilege modes: 0–3, called rings. Ring 0 is the most privileged and 3 the least. The host OS executes in Ring 0. However, instead of executing in Ring 0 as most OSs do, the guest OS runs in lesser-privileged Ring 1, and applications in the least privileged in Ring 3. This explicit virtualization of the physical resources leads to a clear separation between guest and hypervisor, resulting in additional security separation between the two.

Instance Isolation Different instances running on the same physical machine are isolated from each other via the Xen hypervisor. Amazon is active in the Xen community, which provides AWS with awareness of the latest developments. In addition, the AWS firewall resides within the hypervisor layer, between the physical network interface and the instance's virtual interface. All packets must pass through this layer; thus, an instance's neighbors have no more access to that instance than any other host on the Internet and can be treated as if they are on separate physical hosts. The physical RAM is separated using similar mechanisms. Customer instances have no access to raw disk devices, but instead are presented with virtualized disks. The AWS proprietary disk virtualization layer automatically resets every block of storage used by the customer, so that one customer's data is never

unintentionally exposed to another customer. In addition, memory allocated to guests is scrubbed (set to zero) by the hypervisor when it is unallocated to a guest. The memory is not returned to the pool of free memory available for new allocations until the memory scrubbing is completed. Figure 12.3 depicts instance isolation within Amazon EC2.

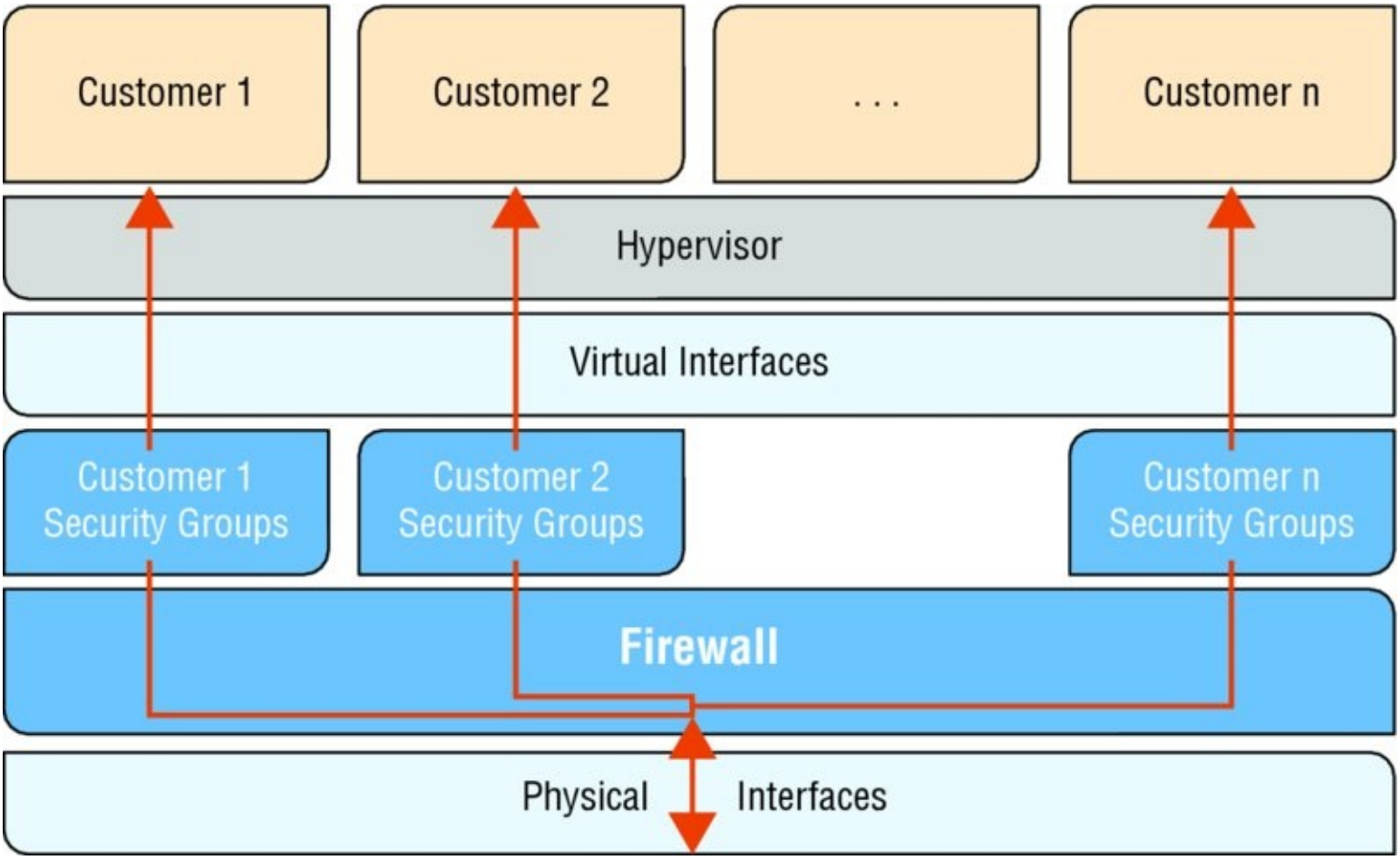

FIGURE 12.3 Amazon EC2 multiple layers of security

Host Operating System Administrators with a business need to access the management plane are required to use MFA to gain access to purpose-built administration hosts. These administrative hosts are systems that are specifically designed, built, configured, and hardened to protect the management plane of the cloud. All such access is logged and audited. When an employee no longer has a business need to access the management plane, the privileges and access to these hosts and relevant systems can be revoked.

Guest Operating System Virtual instances are completely controlled by you, the customer. You have full root access or administrative control over accounts, services, and applications. AWS does not have any access rights to your instances or the guest OS. AWS recommends a base set of security best practices to include disabling password-only access to your guests, and using some form of MFA to gain access to your instances (or at a minimum certificate-based SSH Version 2 access). Additionally, you should employ a privilege escalation mechanism with logging on a per-user basis. For example, if the guest OS is Linux, after hardening, your instance you should use certificate-based SSHv2 to access the virtual instance, disable remote root login, use command-line logging, and use `sudo` for privilege escalation. You should generate your own key pairs in order to guarantee that they are unique and not shared with other customers or with AWS. AWS also supports the use of the SSH network protocol to enable you to log in securely to your UNIX/Linux Amazon EC2 instances.

Authentication for SSH used with AWS is via a public/private key pair to reduce the risk of unauthorized access to your instance. You can also connect remotely to your Windows instances using Remote Desktop Protocol (RDP) by using an RDP certificate generated for your instance. You also control the updating and patching of your guest OS, including security updates. Amazon-provided Windows and Linux-based AMIs are updated regularly with the latest patches, so if you do not need to preserve data or customizations on your running Amazon AMI instances, you can simply relaunch new instances with the latest updated AMI. In addition, updates are provided for the Amazon Linux AMI via the Amazon Linux yum repositories.

Firewall Amazon EC2 provides a mandatory inbound firewall that is configured in a default deny-all mode; Amazon EC2 customers must explicitly open the ports needed to allow inbound traffic. The traffic may be restricted by protocol, by service port, and by source IP address (individual IP or Classless Inter-Domain Routing [CIDR] block).

The firewall can be configured in groups, permitting different classes of instances to have different rules. Consider, for example, the case of a traditional three-tiered web application. The group for the web servers would have port 80 (HTTP) and/or port 443 (HTTPS) open to the Internet. The group for the application servers would have port 8000 (application specific) accessible only to the web server group. The group for the database servers would have port 3306 (MySQL) open only to the application server group. All three groups would permit administrative access on port 22 (SSH), but only from the customer's corporate network. Highly secure applications can be deployed using this approach, which is also depicted in Figure 12.4.

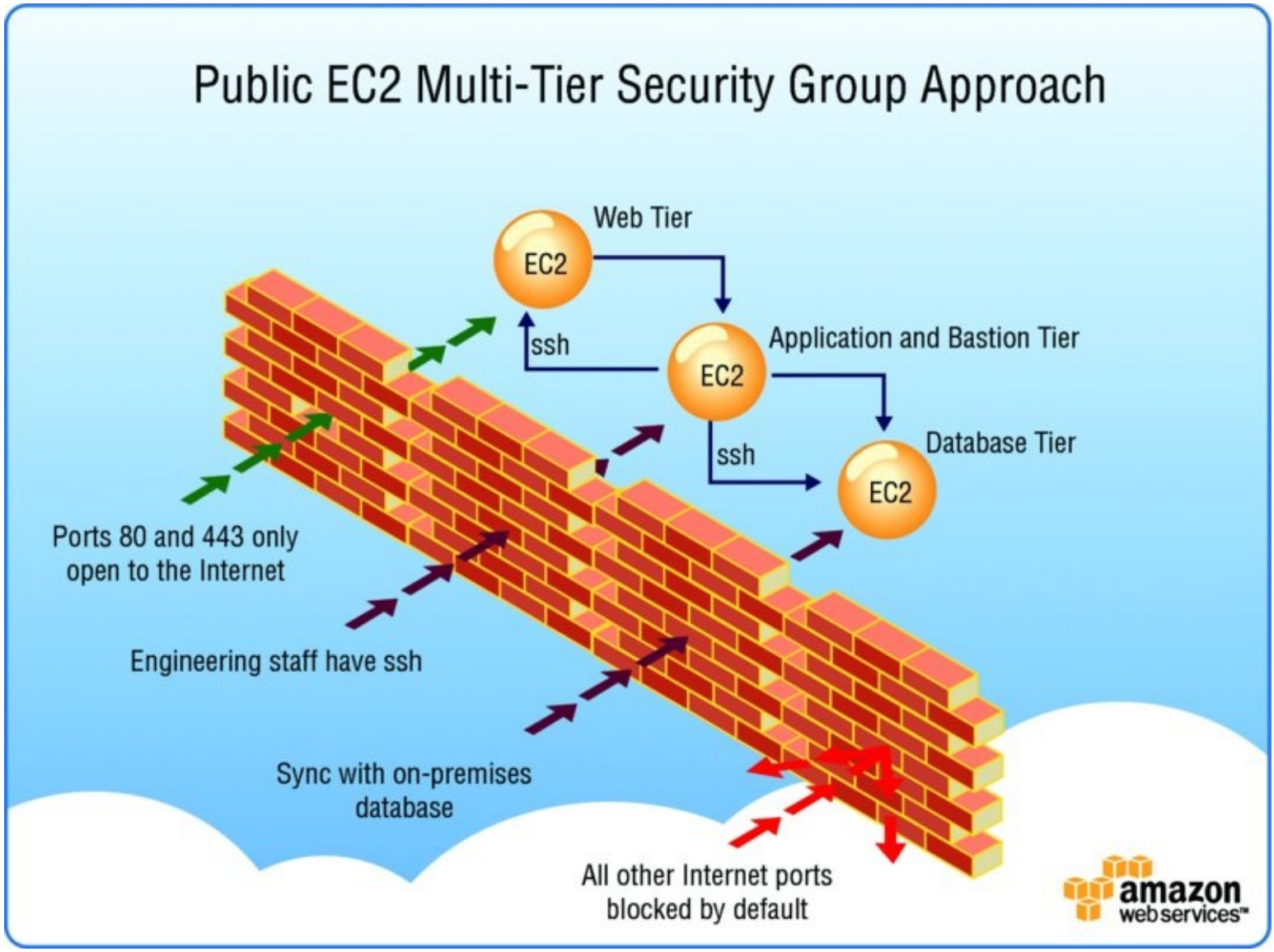

FIGURE 12.4 Amazon EC2 security group firewall

The level of security afforded by the firewall is a function of which ports you open and for what duration and purpose. Well-informed traffic management and security design are still required on a per-instance basis. AWS further encourages you to apply additional per-instance filters with host-based firewalls such as IPtables or the Windows Firewall and VPNs. This can restrict both inbound and outbound traffic.


The default state is to deny all incoming traffic, and you should carefully plan what you will open when building and securing your applications.

API Access API calls to launch and terminate instances, change firewall parameters, and perform other functions are all signed by your Amazon Secret Access Key, which could be either the AWS account's Secret Access Key or the Secret Access key of a user created with AWS IAM. Without access to your Secret Access Key, Amazon EC2 API calls cannot be made on your behalf. API calls can also be encrypted with SSL to maintain confidentiality. AWS recommends always using SSL-protected API endpoints.

Amazon Elastic Block Storage (Amazon EBS) Security *Amazon EBS* allows you to create storage volumes from 1 GB to 16 TB that can be mounted as devices by Amazon EC2

instances. Storage volumes behave like raw, unformatted block devices, with user-supplied device names and a block device interface. You can create a file system on top of Amazon EBS volumes or use them in any other way you would use a block device (like a hard drive). Amazon EBS volume access is restricted to the AWS account that created the volume and to the users under the AWS account created with AWS IAM (if the user has been granted access to the EBS operations). All other AWS accounts and users are denied the permission to view or access the volume.

Data stored in Amazon EBS volumes is redundantly stored in multiple physical locations as part of normal operation of those services and at no additional charge. However, Amazon EBS replication is stored within the same Availability Zone, not across multiple zones; therefore, it is highly recommended that you conduct regular snapshots to Amazon S3 for long-term data durability. For customers who have architected complex transactional databases using Amazon EBS, it is recommended that backups to Amazon S3 be performed through the database management system so that distributed transactions and logs can be checkpointed. AWS does not automatically perform backups of data that are maintained on virtual disks attached to running instances on Amazon EC2.

You can make Amazon EBS volume snapshots publicly available to other AWS accounts to use as the basis for creating duplicate volumes. Sharing Amazon EBS volume snapshots does not provide other AWS accounts with the permission to alter or delete the original snapshot, as that right is explicitly reserved for the AWS account that created the volume. An Amazon EBS snapshot is a block-level view of an entire Amazon EBS volume. Note that data that is not visible through the filesystem on the volume, such as files that have been deleted, may be present in the Amazon EBS snapshot. If you want to create shared snapshots, you should do so carefully. If a volume has held sensitive data or has had files deleted from it, you should create a new Amazon EBS volume to share. The data to be contained in the shared snapshot should be copied to the new volume, and the snapshot created from the new volume.

Amazon EBS volumes are presented to you as raw unformatted *block devices* that have been wiped prior to being made available for use. Wiping occurs immediately before reuse so that you can be assured that the wipe process is completed. If you have procedures requiring that all data be wiped via a specific method, you have the ability to do so on Amazon EBS. You should conduct a specialized wipe procedure prior to deleting the volume for compliance with your established requirements.

Encryption of sensitive data is generally a good security practice, and AWS provides the ability to encrypt Amazon EBS volumes and their snapshots with Advanced Encryption Standard (AES)-256. The encryption occurs on the servers that host the Amazon EC2 instances, providing encryption of data as it moves between Amazon EC2 instances and Amazon EBS storage. In order to be able to do this efficiently and with low latency, the Amazon EBS encryption feature is only available on Amazon EC2's more powerful instance types.

Networking

AWS provides a range of networking services that enable you to create a logically isolated network that you define, establish a private network connection to the AWS Cloud, use a highly available and scalable Domain Name System (DNS) service, and deliver content to your end users with low latency at high data transfer speeds with a content delivery web

service.

Elastic Load Balancing Security

Elastic Load Balancing is used to manage traffic on a fleet of Amazon EC2 instances, distributing traffic to instances across all Availability Zones within a region. Elastic Load Balancing has all of the advantages of an on-premises load balancer, plus several security benefits:

- Takes over the encryption and decryption work from the Amazon EC2 instances and manages it centrally on the load balancer.
- Offers clients a single point of contact, and can also serve as the first line of defense against attacks on your network.
- When used in an Amazon VPC, supports creation and management of security groups associated with your Elastic Load Balancing to provide additional networking and security options.
- Supports end-to-end traffic encryption using TLS (previously SSL) on those networks that use secure HTTP (HTTPS) connections. When TLS is used, the TLS server certificate used to terminate client connections can be managed centrally on the load balancer, instead of on every individual instance.

HTTPS/TLS uses a long-term secret key to generate a short-term session key to be used between the server and the browser to create the encrypted message. Elastic Load Balancing configures your load balancer with a pre-defined cipher set that is used for TLS negotiation when a connection is established between a client and your load balancer. The pre-defined cipher set provides compatibility with a broad range of clients and uses strong cryptographic algorithms. However, some customers may have requirements for allowing only specific ciphers and protocols (for example, Payment Card Industry Data Security Standard [PCI DSS], Sarbanes-Oxley Act [SOX]) from clients to ensure that standards are met. In these cases, Elastic Load Balancing provides options for selecting different configurations for TLS protocols and ciphers. You can choose to enable or disable the ciphers depending on your specific requirements.

To help ensure the use of newer and stronger cipher suites when establishing a secure connection, you can configure the load balancer to have the final say in the cipher suite selection during the client-server negotiation. When the Server Order Preference option is selected, the load balancer will select a cipher suite based on the server's prioritization of cipher suites instead of the client's. This gives you more control over the level of security that clients use to connect to your load balancer.

For even greater communication privacy, Elastic Load Balancing allows the use of Perfect Forward Secrecy, which uses session keys that are ephemeral and not stored anywhere. This prevents the decoding of captured data, even if the secret long-term key itself is compromised.

Elastic Load Balancing allows you to identify the originating IP address of a client connecting to your servers, whether you're using HTTPS or TCP load balancing. Typically, client connection information, such as IP address and port, is lost when requests are proxied through a load balancer. This is because the load balancer sends requests to the server on

behalf of the client, making your load balancer appear as though it is the requesting client. Having the originating client IP address is useful if you need more information about visitors to your applications in order to gather connection statistics, analyze traffic logs, or manage whitelists of IP addresses.

Elastic Load Balancing access logs contain information about each HTTP and TCP request processed by your load balancer. This includes the IP address and port of the requesting client, the back-end IP address of the instance that processed the request, the size of the request and response, and the actual request line from the client (for example, `GET http://www.example.com: 80/HTTP/1.1`). All requests sent to the load balancer are logged, including requests that never make it to back-end instances.

Amazon Virtual Private Cloud (Amazon VPC) Security

Normally, each Amazon EC2 instance you launch is randomly assigned a public IP address in the Amazon EC2 address space. *Amazon VPC* enables you to create an isolated portion of the AWS Cloud and launch Amazon EC2 instances that have private (RFC 1918) addresses in the range of your choice (for example, 10.0.0.0/16). You can define subnets within your Amazon VPC, grouping similar kinds of instances based on IP address range and then set up routing and security to control the flow of traffic in and out of the instances and subnets.

Security features within Amazon VPC include security groups, *network ACLs*, routing tables, and external gateways. Each of these items is complementary to providing a secure, isolated network that can be extended through selective enabling of direct Internet access or private connectivity to another network. Amazon EC2 instances running within an Amazon VPC inherit all of the benefits described below related to the guest OS and protection against packet sniffing. Note, however, that you must create security groups specifically for your Amazon VPC; any Amazon EC2 security groups you have created will not work inside your Amazon VPC. In addition, Amazon VPC security groups have additional capabilities that Amazon EC2 security groups do not have, such as being able to change the security group after the instance is launched and being able to specify any protocol with a standard protocol number (as opposed to just TCP, User Datagram Protocol [UDP], or Internet Control Message Protocol [ICMP]).

Each Amazon VPC is a distinct, isolated network within the cloud; network traffic within each Amazon VPC is isolated from all other Amazon VPCs. At creation time, you select an IP address range for each Amazon VPC. You may create and attach an Internet gateway, virtual private gateway, or both to establish external connectivity, subject to the following controls.

API Access Calls to create and delete Amazon VPCs; change routing, security group, and network ACL parameters; and perform other functions are all signed by your Amazon Secret Access Key, which could be either the AWS account's Secret Access Key or the Secret Access key of a user created with AWS IAM. Without access to your Secret Access Key, Amazon VPC API calls cannot be made on your behalf. In addition, API calls can be encrypted with SSL to maintain confidentiality. AWS recommends always using SSL-protected API endpoints. AWS IAM also enables a customer to further control what APIs a newly created user has permissions to call.

Subnets and Route Tables You create one or more subnets within each Amazon VPC; each instance launched in the Amazon VPC is connected to one subnet. Traditional Layer 2

security attacks, including MAC spoofing and ARP spoofing, are blocked. Each subnet in an Amazon VPC is associated with a routing table, and all network traffic leaving the subnet is processed by the routing table to determine the destination.

Firewall (Security Groups) Like Amazon EC2, Amazon VPC supports a complete firewall solution, enabling filtering on both ingress and egress traffic from an instance. The default group enables inbound communication from other members of the same group and outbound communication to any destination. Traffic can be restricted by any IP protocol, by service port, and source/destination IP address (individual IP or CIDR block). The firewall isn't controlled through the guest OS; rather, it can be modified only through the invocation of Amazon VPC APIs. AWS supports the ability to grant granular access to different administrative functions on the instances and the firewall, therefore enabling you to implement additional security through separation of duties. The level of security afforded by the firewall is a function of which ports you open and for what duration and purpose. Well-informed traffic management and security design are still required on a per-instance basis. AWS further encourages you to apply additional per-instance filters with host-based firewalls such as IPtables or the Windows Firewall. Figure 12.5 illustrates an Amazon VPC with two types of subnets—public and private—and two network paths with two different networks—a customer data center and the Internet.

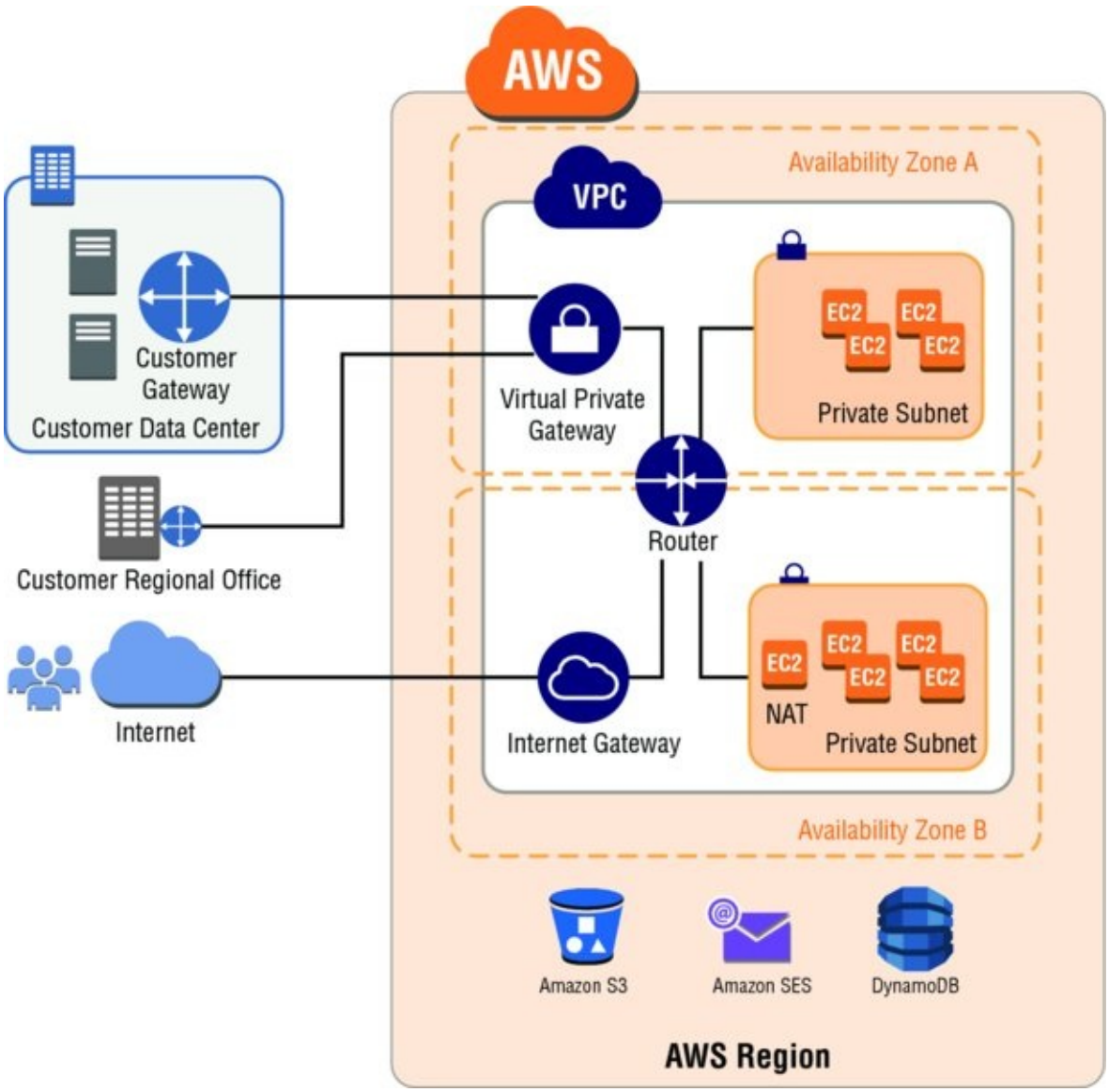

FIGURE 12.5 Amazon VPC network architecture

Network ACLs To add a further layer of security within Amazon VPC, you can configure network ACLs. These are stateless traffic filters that apply to all traffic inbound or outbound from a subnet within Amazon VPC. These ACLs can contain ordered rules to allow or deny traffic based on IP protocol, by service port, and source/destination IP address.

Like security groups, network ACLs are managed through Amazon VPC APIs, adding an additional layer of protection and enabling additional security through separation of duties. Figure 12.6 depicts how the security controls above interrelate to enable flexible network topologies while providing complete control over network traffic flows.

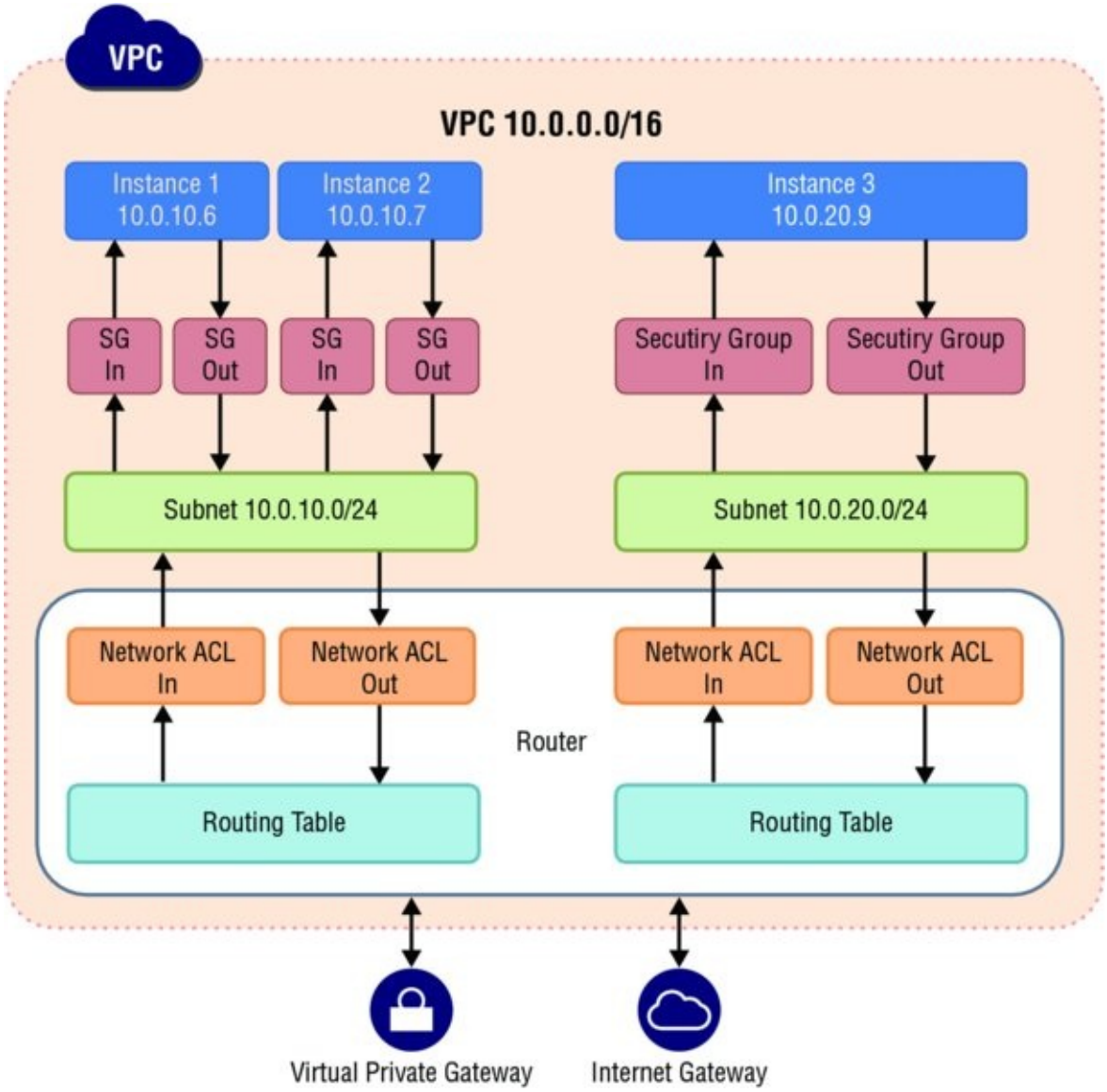

FIGURE 12.6 Flexible network architectures

Virtual Private Gateway A virtual private gateway enables private connectivity between the Amazon VPC and another network. Network traffic within each virtual private gateway is isolated from network traffic within all other virtual private gateways. You can establish VPN connections to the virtual private gateway from gateway devices at your premises. Each connection is secured by a preshared key in conjunction with the IP address of the customer gateway device.

Internet Gateway An Internet gateway may be attached to an Amazon VPC to enable direct connectivity to Amazon S3, other AWS services, and the Internet. Each instance desiring this access must either have an Elastic IP associated with it or route traffic through a Network

Address Translation (NAT) instance. Additionally, network routes are configured to direct traffic to the Internet gateway (see Figure 12.6). AWS provides reference NAT AMIs that you can extend to perform network logging, deep packet inspection, application layer filtering, or other security controls.

This access can only be modified through the invocation of Amazon VPC APIs. AWS supports the ability to grant granular access to different administrative functions on the instances and the Internet gateway, enabling you to implement additional security through separation of duties.

Dedicated Instances Within an Amazon VPC, you can launch Amazon EC2 instances that are physically isolated at the host hardware level (that is, they will run on single-tenant hardware). An Amazon VPC can be created with "dedicated" tenancy, so that all instances launched into the Amazon VPC will use this feature. Alternatively, an Amazon VPC may be created with "default" tenancy, but you can specify dedicated tenancy for particular instances launched into it.

Amazon CloudFront Security

Amazon CloudFront gives customers an easy way to distribute content to end users with low latency and high data transfer speeds. It delivers dynamic, static, and streaming content using a global network of edge locations. Requests for customers' objects are automatically routed to the nearest edge location, so content is delivered with the best possible performance. Amazon CloudFront is optimized to work with other AWS services like Amazon S3, Amazon EC2, Elastic Load Balancing, and Amazon Route 53. It also works seamlessly with any non-AWS origin server that stores the original, definitive versions of your files.

Amazon CloudFront requires that every request made to its control API is authenticated so only authorized users can create, modify, or delete their own Amazon CloudFront distributions. Requests are signed with an HMAC-SHA-1 signature calculated from the request and the user's private key. Additionally, the Amazon CloudFront control API is only accessible via SSL-enabled endpoints.

There is no guarantee of durability of data held in Amazon CloudFront edge locations. The service may sometimes remove objects from edge locations if those objects are not requested frequently. Durability is provided by Amazon S3, which works as the origin server for Amazon CloudFront by holding the original, definitive copies of objects delivered by Amazon CloudFront.

If you want control over who can download content from Amazon CloudFront, you can enable the service's private content feature. This feature has two components. The first controls how content is delivered from the Amazon CloudFront edge location to viewers on the Internet. The second controls how the Amazon CloudFront edge locations access objects in Amazon S3. Amazon CloudFront also supports geo restriction, which restricts access to your content based on the geographic location of your viewers.

To control access to the original copies of your objects in Amazon S3, Amazon CloudFront allows you to create one or more Origin Access Identities and associate these with your distributions. When an Origin Access Identity is associated with an Amazon CloudFront distribution, the distribution will use that identity to retrieve objects from Amazon S3. You can then use Amazon S3's ACL feature, which limits access to that Origin Access Identity so

the original copy of the object is not publicly readable.

To control who can download objects from Amazon CloudFront edge locations, the service uses a signed-URL verification system. To use this system, you first create a public-private key pair and upload the public key to your account via the AWS Management Console. You then configure your Amazon CloudFront distribution to indicate which accounts you would authorize to sign requests—you can indicate up to five AWS accounts that you trust to sign requests. As you receive requests, you will create policy documents indicating the conditions under which you want Amazon CloudFront to serve your content. These policy documents can specify the name of the object that is requested, the date and time of the request, and the source IP (or CIDR range) of the client making the request. You then calculate the SHA-1 hash of your policy document and sign this using your private key. Finally, you include both the encoded policy document and the signature as query string parameters when you reference your objects. When Amazon CloudFront receives a request, it will decode the signature using your public key. Amazon CloudFront will only serve requests that have a valid policy document and matching signature.

Note that private content is an optional feature that must be enabled when you set up your Amazon CloudFront distribution. Content delivered without this feature enabled will be publicly readable.

Amazon CloudFront provides the option to transfer content over an encrypted connection (HTTPS). By default, Amazon CloudFront will accept requests over both HTTP and HTTPS protocols. However, you can also configure Amazon CloudFront to require HTTPS for all requests or have Amazon CloudFront redirect HTTP requests to HTTPS. You can even configure Amazon CloudFront distributions to allow HTTP for some objects but require HTTPS for other objects.

Storage

AWS provides low-cost data storage with high durability and availability. AWS offers storage choices for backup, archiving, and disaster recovery, and also for block and object storage.

Amazon Simple Storage Service (Amazon S3) Security

Amazon S3 allows you to upload and retrieve data at any time, from anywhere on the web. Amazon S3 stores data as objects within buckets. An object can be any kind of file: a text file, a photo, a video, and more. When you add a file to Amazon S3, you have the option of including metadata with the file and setting permissions to control access to the file. For each bucket, you can control access to the bucket (who can create, delete, and list objects in the bucket), view access logs for the bucket and its objects, and choose the geographical region where Amazon S3 will store the bucket and its contents.

Data Access

Access to data stored in Amazon S3 is restricted by default; only bucket and object owners have access to the Amazon S3 resources they create. (Note that a bucket/object owner is the AWS account owner, not the user who created the bucket/object.) There are multiple ways to control access to buckets and objects:

IAM Policies AWS IAM enables organizations with many employees to create and manage

multiple users under a single AWS account. IAM policies are attached to the users, enabling centralized control of permissions for users under your AWS account to access buckets or objects. With IAM policies, you can only grant users within your own AWS account permission to access your Amazon S3 resources.

ACLs Within Amazon S3, you can use ACLs to give read or write access on buckets or objects to groups of users. With ACLs, you can only grant other AWS accounts (not specific users) access to your Amazon S3 resources.

Bucket Policies Bucket policies in Amazon S3 can be used to add or deny permissions across some or all of the objects within a single bucket. Policies can be attached to users, groups, or Amazon S3 buckets, enabling centralized management of permissions. With bucket policies, you can grant users within your AWS account or other AWS accounts access to your Amazon S3 resources.

Query String Authentication You can use a query string to express a request entirely in a URL. In this case, you use query parameters to provide request information, including the authentication information. Because the request signature is part of the URL, this type of URL is often referred to as a *pre-signed URL*. You can use pre-signed URLs to embed clickable links, which can be valid for up to seven days, in HTML.

You can further restrict access to specific resources based on certain conditions. For example, you can restrict access based on request time (Date Condition), whether the request was sent using SSL (Boolean Conditions), a requester's IP address (IP Address Condition), or the requester's client application (String Conditions). To identify these conditions, you use policy keys.

Amazon S3 also gives developers the option to use query string authentication, which allows them to share Amazon S3 objects through URLs that are valid for a predefined period of time. Query string authentication is useful for giving HTTP for browser access to resources that would normally require authentication. The signature in the query string secures the request.

Data Transfer

For maximum security, you can securely upload/download data to Amazon S3 via the SSL-encrypted endpoints. The encrypted endpoints are accessible from both the Internet and from within Amazon EC2, so that data is transferred securely both within AWS and to and from sources outside of AWS.

Data Storage

Amazon S3 provides multiple options for protecting data at rest. For customers who prefer to manage their own encryption, they can use a client encryption library like the Amazon S3 Encryption Client to encrypt data before uploading to Amazon S3. Alternatively, you can use Amazon S3 *Server Side Encryption (SSE)* if you prefer to have Amazon S3 manage the encryption process for you. Data is encrypted with a key generated by AWS or with a key you supply, depending on your requirements. With Amazon S3 SSE, you can encrypt data on upload simply by adding an additional request header when writing the object. Decryption happens automatically when data is retrieved. Note that metadata, which you can include with your object, is not encrypted.

 AWS recommends that customers not place sensitive information in Amazon S3 metadata.

Amazon S3 SSE uses one of the strongest block ciphers available: AES-256. With Amazon S3 SSE, every protected object is encrypted with a unique encryption key. This object key itself is then encrypted with a regularly rotated master key. Amazon S3 SSE provides additional security by storing the encrypted data and encryption keys in different hosts. Amazon S3 SSE also makes it possible for you to enforce encryption requirements. For example, you can create and apply bucket policies that require that only encrypted data can be uploaded to your buckets.

When an object is deleted from Amazon S3, removal of the mapping from the public name to the object starts immediately and is generally processed across the distributed system within several seconds. After the mapping is removed, there is no remote access to the deleted object. The underlying storage area is then reclaimed for use by the system.

Amazon S3 Standard is designed to provide 99.999999999 percent durability of objects over a given year. This durability level corresponds to an average annual expected loss of 0.000000001 percent of objects. For example, if you store 10,000 objects with Amazon S3, you can, on average, expect to incur a loss of a single object once every 10,000,000 years. In addition, Amazon S3 is designed to sustain the concurrent loss of data in two facilities.

Access Logs

An Amazon S3 bucket can be configured to log access to the bucket and objects within it. The access log contains details about each access request including request type, the requested resource, the requestor's IP, and the time and date of the request. When logging is enabled for a bucket, log records are periodically aggregated into log files and delivered to the specified Amazon S3 bucket.

Cross-Origin Resource Sharing (CORS)

AWS customers who use Amazon S3 to host static web pages or store objects used by other web pages can load content securely by configuring an Amazon S3 bucket to explicitly enable cross-origin requests. Modern browsers use the Same Origin policy to block JavaScript or HTML5 from allowing requests to load content from another site or domain as a way to help ensure that malicious content is not loaded from a less reputable source (such as during cross-site scripting attacks). With the *Cross-Origin Resource Sharing (CORS)*policy enabled, assets such as web fonts and images stored in an Amazon S3 bucket can be safely referenced by external web pages, style sheets, and HTML5 applications.

Amazon Glacier Security

Like Amazon S3, the *Amazon Glacier* service provides low-cost, secure, and durable storage. Where Amazon S3 is designed for rapid retrieval, however, Amazon Glacier is meant to be used as an archival service for data that is not accessed often and for which retrieval times of several hours are suitable.

Amazon Glacier stores files as archives within vaults. Archives can be any data such as a photo, video, or document, and can contain one or several files. You can store an unlimited number of archives in a single vault and can create up to 1,000 vaults per region. Each archive can contain up to 40 TB of data.

Data Transfer

For maximum security, you can securely upload/download data to Amazon Glacier via the SSL encrypted endpoints. The encrypted endpoints are accessible from both the Internet and from within Amazon EC2, so that data is transferred securely both within AWS and to and from sources outside of AWS.

Data Retrieval

Retrieving archives from Amazon Glacier requires the initiation of a retrieval job, which is generally completed in three to five hours. You can then access the data via HTTP `GET` requests. The data will remain available to you for 24 hours. You can retrieve an entire archive or several files from an archive. If you want to retrieve only a subset of an archive, you can use one retrieval request to specify the range of the archive that contains the files in which you are interested or you can initiate multiple retrieval requests, each with a range for one or more files.

You can also limit the number of vault inventory items retrieved by filtering on an archive creation date range or by setting a maximum items limit. Whichever method you choose, when you retrieve portions of your archive, you can use the supplied checksum to help ensure the integrity of the files provided that the range that is retrieved is aligned with the tree hash of the overall archive.

Data Storage

Amazon Glacier automatically encrypts the data using AES-256 and stores it durably in an immutable form. Amazon Glacier is designed to provide average annual durability of 99.999999999 percent for an archive. It stores each archive in multiple facilities and multiple devices. Unlike traditional systems, which can require laborious data verification and manual repair, Amazon Glacier performs regular, systematic data integrity checks and is built to be self-healing.

Data Access

Only your account can access your data in Amazon Glacier. To control access to your data in Amazon Glacier, you can use AWS IAM to specify which users within your account have rights to operations on a given vault.

AWS Storage Gateway Security

The *AWS Storage Gateway* service connects your on-premises software appliance with cloud-based storage to provide seamless and secure integration between your IT environment and AWS storage infrastructure. The service enables you to upload data securely to AWS scalable, reliable, and secure Amazon S3 storage service for cost-effective backup and rapid disaster recovery.

Data Transfer

Data is asynchronously transferred from your on-premises storage hardware to AWS over SSL.

Data Storage

The data is stored encrypted in Amazon S3 using AES 256, a symmetric key encryption standard using 256-bit encryption keys. The AWS Storage Gateway only uploads data that has changed, minimizing the amount of data sent over the Internet.

Database

AWS provides a number of database solutions for developers and businesses from managed relational and NoSQL database services, to in-memory caching as a service and petabyte-scale data warehouse service.

Amazon DynamoDB Security

Amazon DynamoDB is a managed NoSQL database service that provides fast and predictable performance with seamless scalability. Amazon DynamoDB enables you to offload the administrative burdens of operating and scaling distributed databases to AWS, so you don't have to worry about hardware provisioning, setup and configuration, replication, software patching, or cluster scaling.

You can create a database table that can store and retrieve any amount of data and serve any level of request traffic. Amazon DynamoDB automatically spreads the data and traffic for the table over a sufficient number of servers to handle the request capacity you specified and the amount of data stored, while maintaining consistent, fast performance. All data items are stored on Solid State Drives (SSDs) and are automatically replicated across multiple Availability Zones in a region to provide built-in high availability and data durability.

You can set up automatic backups using a special template in AWS Data Pipeline that was created just for copying Amazon DynamoDB tables. You can choose full or incremental backups to a table in the same region or a different region. You can use the copy for disaster recovery in the event that an error in your code damages the original table or to federate Amazon DynamoDB data across regions to support a multi-region application.

To control who can use the Amazon DynamoDB resources and API, you set up permissions in AWS IAM. In addition to controlling access at the resource-level with IAM, you can also control access at the database level—you can create database-level permissions that allow or deny access to items (rows) and attributes (columns) based on the needs of your application. These database-level permissions are called *fine-grained access controls*, and you create them using an IAM policy that specifies under what circumstances a user or application can access an Amazon DynamoDB table. The IAM policy can restrict access to individual items in a table, access to the attributes in those items, or both at the same time.

In addition to requiring database and user permissions, each request to the Amazon DynamoDB service must contain a valid HMAC-SHA-256 signature or the request is rejected. The AWS SDKs automatically sign your requests; however, if you want to write your own HTTP POST requests, you must provide the signature in the header of your request to Amazon DynamoDB. To calculate the signature, you must request temporary security credentials from

the AWS Security Token Service. Use the temporary security credentials to sign your requests to Amazon DynamoDB. Amazon DynamoDB is accessible via SSL-encrypted endpoints, and the encrypted endpoints are accessible from both the Internet and from within Amazon EC2.

Amazon Relational Database Service (Amazon RDS) Security

Amazon Relational Database Service (Amazon RDS) allows you to quickly create a relational Database Instance (DB Instance) and flexibly scale the associated compute resources and storage capacity to meet application demand. Amazon RDS manages the database instance on your behalf by performing backups, handling failover, and maintaining the database software. As of the time of this writing, Amazon RDS is available for MySQL, Oracle, Microsoft SQL Server, MariaDB, Amazon Aurora, and PostgreSQL database engines.

Amazon RDS has multiple features that enhance reliability for critical production databases, including DB security groups, permissions, SSL connections, automated backups, DB snapshots, and multiple Availability Zone (Multi-AZ) deployments. DB Instances can also be deployed in an Amazon VPC for additional network isolation.

Access Control When you first create a DB Instance within Amazon RDS, you will create a master user account, which is used only within the context of Amazon RDS to control access to your DB Instance(s). The master user account is a native database user account that allows you to log on to your DB Instance with all database privileges. You can specify the master user name and password you want associated with each DB Instance when you create the DB Instance. After you have created your DB Instance, you can connect to the database using the master user credentials. Subsequently, you can create additional user accounts so that you can restrict who can access your DB Instance.

You can control Amazon RDS DB Instance access via *DB security groups*, which are similar to Amazon EC2 security groups but not interchangeable. DB security groups act like a firewall controlling network access to your DB Instance. DB security groups default to deny all access mode, and customers must specifically authorize network ingress. There are two ways of doing this:

- Authorizing a network IP range
- Authorizing an existing Amazon EC2 security group

DB security groups only allow access to the database server port (all others are blocked) and can be updated without restarting the Amazon RDS DB Instance, which gives you seamless control of their database access.

Using AWS IAM, you can further control access to your Amazon RDS DB instances. AWS IAM enables you to control what Amazon RDS operations each individual AWS IAM user has permission to call.

Network Isolation For additional network access control, you can run your DB Instances in an Amazon VPC. Amazon VPC enables you to isolate your DB Instances by specifying the IP range you want to use and connect to your existing IT infrastructure through industry-standard encrypted IPsec VPN. Running Amazon RDS in a VPC enables you to have a DB instance within a private subnet. You can also set up a virtual private gateway that extends your corporate network into your VPC, and allows access to the RDS DB instance in that VPC.

For Multi-AZ deployments, defining a subnet for all Availability Zones in a region, will allow

Amazon RDS to create a new standby in another Availability Zone should the need arise. You can create DB subnet groups, which are collections of subnets that you may want to designate for your Amazon RDS DB Instances in an Amazon VPC. Each DB subnet group should have at least one subnet for every Availability Zone in a given region. In this case, when you create a DB Instance in an Amazon VPC, you select a DB subnet group; Amazon RDS then uses that DB subnet group and your preferred Availability Zone to select a subnet and an IP address within that subnet. Amazon RDS creates and associates an Elastic Network Interface to your DB Instance with that IP address.

DB Instances deployed within an Amazon VPC can be accessed from the Internet or from Amazon EC2 instances outside the Amazon VPC via VPN or bastion hosts that you can launch in your public subnet. To use a bastion host, you will need to set up a public subnet with an Amazon EC2 instance that acts as a SSH Bastion. This public subnet must have an Internet gateway and routing rules that allow traffic to be directed via the SSH host, which must then forward requests to the private IP address of your Amazon RDS DB Instance.

DB security groups can be used to help secure DB Instances within an Amazon VPC. In addition, network traffic entering and exiting each subnet can be allowed or denied via network ACLs. All network traffic entering or exiting your Amazon VPC via your IPsec VPN connection can be inspected by your on-premises security infrastructure, including network firewalls and intrusion detection systems.

Encryption You can encrypt connections between your application and your DB Instance using SSL. For MySQL and SQL Server, Amazon RDS creates an SSL certificate and installs the certificate on the DB Instance when the instance is provisioned. For MySQL, you launch the MySQL client using the `--ssl_ca` parameter to reference the public key in order to encrypt connections. For SQL Server, download the public key and import the certificate into your Windows operating system. Oracle RDS uses Oracle native network encryption with a DB Instance. You simply add the native network encryption option to an option group and associate that option group with the DB Instance. After an encrypted connection is established, data transferred between the DB Instance and your application will be encrypted during transfer. You can also require your DB Instance to accept only encrypted connections.

Amazon RDS supports Transparent Data Encryption (TDE) for SQL Server (SQL Server Enterprise Edition) and Oracle (part of the Oracle Advanced Security option available in Oracle Enterprise Edition). The TDE feature automatically encrypts data before it is written to storage and automatically decrypts data when it is read from storage. If you require your MySQL data to be encrypted while at rest in the database, your application must manage the encryption and decryption of data.

Note that SSL support within Amazon RDS is for encrypting the connection between your application and your DB Instance; it should not be relied on for authenticating the DB Instance itself. While SSL offers security benefits, be aware that SSL encryption is a compute intensive operation and will increase the latency of your database connection.

Automated Backups and DB Snapshots Amazon RDS provides two different methods for backing up and restoring your DB Instance(s): automated backups and Database Snapshots (DB Snapshots). Turned on by default, the automated backup feature of Amazon RDS enables point-in-time recovery for your DB Instance. Amazon RDS will back up your database and transaction logs and store both for a user-specified retention period. This allows

you to restore your DB Instance to any second during your retention period, up to the last five minutes. Your automatic backup retention period can be configured to up to 35 days.

DB Snapshots are user-initiated backups of your DB Instance. These full database backups are stored by Amazon RDS until you explicitly delete them. You can copy DB snapshots of any size and move them between any of AWS public regions, or copy the same snapshot to multiple regions simultaneously. You can then create a new DB Instance from a DB Snapshot whenever you desire.

During the backup window, storage I/O may be suspended while your data is being backed up. This I/O suspension typically lasts a few minutes. This I/O suspension is avoided with Multi-AZ DB deployments, because the backup is taken from the standby.

DB Instance Replication AWS Cloud computing resources are housed in highly available data center facilities in different regions of the world, and each region contains multiple distinct locations called Availability Zones. Each Availability Zone is engineered to be isolated from failures in other Availability Zones and provide inexpensive, low-latency network connectivity to other Availability Zones in the same region.

To architect for high availability of your Oracle, PostgreSQL, or MySQL databases, you can run your Amazon RDS DB Instance in several Availability Zones, an option called a *Multi-AZ deployment*. When you select this option, AWS automatically provisions and maintains a synchronous standby replica of your DB Instance in a different Availability Zone. The primary DB Instance is synchronously replicated across Availability Zones to the standby replica. In the event of DB Instance or Availability Zone failure, Amazon RDS will automatically failover to the standby so that database operations can resume quickly without administrative intervention.

For customers who use MySQL and need to scale beyond the capacity constraints of a single DB Instance for read-heavy database workloads, Amazon RDS provides a read replica option. After you create a read replica, database updates on the source DB Instance are replicated to the read replica using MySQL's native, asynchronous replication. You can create multiple read replicas for a given source DB instance and distribute your application's read traffic among them. Read replicas can be created with Multi-AZ deployments to gain read scaling benefits in addition to the enhanced database write availability and data durability provided by Multi-AZ deployments.

Automatic Software Patching Amazon RDS will make sure that the relational database software powering your deployment stays up-to-date with the latest patches. When necessary, patches are applied during a maintenance window that you can control. You can think of the Amazon RDS maintenance window as an opportunity to control when DB Instance modifications (such as scaling DB Instance class) and software patching occur, in the event either are requested or required. If a maintenance event is scheduled for a given week, it will be initiated and completed at some point during the 30-minute maintenance window you identify.

The only maintenance events that require Amazon RDS to take your DB Instance offline are scale compute operations (which generally take only a few minutes from start to finish) or required software patching. Required patching is automatically scheduled only for patches that are related to security and durability. Such patching occurs infrequently (typically once every few months) and should seldom require more than a fraction of your maintenance

window. If you do not specify a preferred weekly maintenance window when creating your DB Instance, a 30-minute default value is assigned. If you want to modify when maintenance is performed on your behalf, you can do so by modifying your DB Instance in the AWS Management Console or by using the `ModifyDBInstance` API. Each of your DB Instances can have different preferred maintenance windows, if you so choose.

Running your DB Instance in a Multi-AZ deployment can further reduce the impact of a maintenance event, as Amazon RDS will conduct maintenance via the following steps:

1. Perform maintenance on standby.
2. Promote standby to primary.
3. Perform maintenance on old primary, which becomes the new standby.

When an Amazon RDS DB Instance deletion API (`DeleteDBInstance`) is run, the DB Instance is marked for deletion. After the instance no longer indicates deleting status, it has been removed. At this point, the instance is no longer accessible, and unless a final snapshot copy was asked for, it cannot be restored and will not be listed by any of the tools or APIs.

Amazon Redshift Security

Amazon Redshift is a petabyte-scale SQL data warehouse service that runs on highly optimized and managed AWS compute and storage resources. The service has been architected not only to scale up or down rapidly, but also to improve query speeds significantly even on extremely large datasets. To increase performance, Amazon Redshift uses techniques such as columnar storage, data compression, and zone maps to reduce the amount of I/O needed to perform queries. It also has a Massively Parallel Processing (MPP) architecture, parallelizing and distributing SQL operations to take advantage of all available resources.

Cluster Access By default, clusters that you create are closed to everyone. Amazon Redshift enables you to configure firewall rules (security groups) to control network access to your data warehouse cluster. You can also run Amazon Redshift inside an Amazon VPC to isolate your data warehouse cluster in your own virtual network and connect it to your existing IT infrastructure using industry-standard encrypted IPsec VPN.

The AWS account that creates the cluster has full access to the cluster. Within your AWS account, you can use AWS IAM to create user accounts and manage permissions for those accounts. By using IAM, you can grant different users permission to perform only the cluster operations that are necessary for their work. Like all databases, you must grant permission in Amazon Redshift at the database level in addition to granting access at the resource level. Database users are named user accounts that can connect to a database and are authenticated when they log in to Amazon Redshift. In Amazon Redshift, you grant database user permissions on a per-cluster basis instead of on a per-table basis. However, users can see data only in the table rows that were generated by their own activities; rows generated by other users are not visible to them.

The user who creates a database object is its owner. By default, only a super user or the owner of an object can query, modify, or grant permissions on the object. For users to use an object, you must grant the necessary permissions to the user or the group that contains the user. In addition, only the owner of an object can modify or delete it.

Data Backups Amazon Redshift distributes your data across all compute nodes in a cluster. When you run a cluster with at least two compute nodes, data on each node will always be mirrored on disks on another node, reducing the risk of data loss. In addition, all data written to a node in your cluster is continuously backed up to Amazon S3 using snapshots. Amazon Redshift stores your snapshots for a user-defined period, which can be from 1 to 35 days. You can also take your own snapshots at any time; these snapshots leverage all existing system snapshots and are retained until you explicitly delete them.

Amazon Redshift continuously monitors the health of the cluster and automatically re-replicates data from failed drives and replaces nodes as necessary. All of this happens without any effort on your part, although you may see a slight performance degradation during the re-replication process.

You can use any system or user snapshot to restore your cluster using the AWS Management Console or the Amazon Redshift APIs. Your cluster is available as soon as the system metadata has been restored, and you can start running queries while user data is spooled down in the background.

Data Encryption When creating a cluster, you can choose to encrypt it in order to provide additional protection for your data at rest. When you enable encryption in your cluster, Amazon Redshift stores all data in user-created tables in an encrypted format using hardware-accelerated AES-256 block encryption keys. This includes all data written to disk and any backups.

Amazon Redshift uses a four-tier, key-based architecture for encryption. These keys consist of data encryption keys, a database key, a cluster key, and a master key.

- Data encryption keys encrypt data blocks in the cluster. Each data block is assigned a randomly-generated AES256 key. These keys are encrypted by using the database key for the cluster.
- The database key encrypts data encryption keys in the cluster. The database key is a randomly-generated AES-256 key. It is stored on disk in a separate network from the Amazon Redshift cluster and encrypted by a master key. Amazon Redshift passes the database key across a secure channel and keeps it in memory in the cluster.
- The cluster key encrypts the database key for the Amazon Redshift cluster. You can use either AWS or a Hardware Security Module (HSM) to store the cluster key. HSMs provide direct control of key generation and management and make key management separate and distinct from the application and the database.
- The master key encrypts the cluster key if it is stored in AWS. The master key encrypts the cluster-key-encrypted database key if the cluster key is stored in an HSM.

You can have Amazon Redshift rotate the encryption keys for your encrypted clusters at any time. As part of the rotation process, keys are also updated for all of the cluster's automatic and manual snapshots. Note that enabling encryption in your cluster will impact performance, even though it is hardware accelerated.

Encryption also applies to backups. When you're restoring from an encrypted snapshot, the new cluster will be encrypted as well.

To encrypt your table load data files when you upload them to Amazon S3, you can use

Amazon S3 server-side encryption. When you load the data from Amazon S3, the `COPY` command will decrypt the data as it loads the table.

Database Audit Logging Amazon Redshift logs all SQL operations, including connection attempts, queries, and changes to your database. You can access these logs using SQL queries against system tables or choose to have them downloaded to a secure Amazon S3 bucket. You can then use these audit logs to monitor your cluster for security and troubleshooting purposes.

Automatic Software Patching Amazon Redshift manages all the work of setting up, operating, and scaling your data warehouse, including provisioning capacity, monitoring the cluster, and applying patches and upgrades to the Amazon Redshift engine. Patches are applied only during specified maintenance windows.

SSL Connections To protect your data in transit within the AWS Cloud, Amazon Redshift uses hardware-accelerated SSL to communicate with Amazon S3 or Amazon DynamoDB for `COPY`, `UNLOAD`, backup, and restore operations. You can encrypt the connection between your client and the cluster by specifying SSL in the parameter group associated with the cluster. To have your clients also authenticate the Amazon Redshift server, you can install the public key (`.pem` file) for the SSL certificate on your client and use the key to connect to your clusters.

Amazon Redshift offers the newer, stronger cipher suites that use the Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) protocol. ECDHE allows SSL clients to provide Perfect Forward Secrecy between the client and the Amazon Redshift cluster. Perfect Forward Secrecy uses session keys that are ephemeral and not stored anywhere, which prevents the decoding of captured data by unauthorized third parties, even if the secret long-term key itself is compromised. You do not need to configure anything in Amazon Redshift to enable ECDHE; if you connect from an SQL client tool that uses ECDHE to encrypt communication between the client and server, Amazon Redshift will use the provided cipher list to make the appropriate connection.

Amazon ElastiCache Security

Amazon ElastiCache is a web service that makes it easy to set up, manage, and scale distributed in-memory cache environments in the cloud. The service improves the performance of web applications by allowing you to retrieve information from a fast, managed, in-memory caching system, instead of relying entirely on slower disk-based databases. It can be used to improve latency and throughput significantly for many read-heavy application workloads (such as social networking, gaming, media sharing, and Q and A portals) or compute-intensive workloads (such as a recommendation engine). Caching improves application performance by storing critical pieces of data in memory for low-latency access. Cached information may include the results of I/O-intensive database queries or the results of computationally-intensive calculations.

The Amazon ElastiCache service automates time-consuming management tasks for in-memory cache environments, such as patch management, failure detection, and recovery. It works in conjunction with other AWS Cloud services (such as Amazon EC2, Amazon CloudWatch, and Amazon SNS) to provide a secure, high-performance, and managed in-memory cache. For example, an application running in Amazon EC2 can securely access an Amazon ElastiCache cluster in the same region with very low latency.

Using the Amazon ElastiCache service, you create a *Cache Cluster*, which is a collection of one or more *Cache Nodes*, each running an instance of the Memcached service. A Cache Node is a fixed-size chunk of secure, network-attached RAM. Each Cache Node runs an instance of the Memcached service and has its own DNS name and port. Multiple types of Cache Nodes are supported, each with varying amounts of associated memory. A Cache Cluster can be set up with a specific number of Cache Nodes and a Cache Parameter Group that controls the properties for each Cache Node. All Cache Nodes within a Cache Cluster are designed to be of the same Node Type and have the same parameter and security group settings.

Data Access Amazon ElastiCache allows you to control access to your Cache Clusters using *Cache Security Groups*. A Cache Security Group acts like a firewall, controlling network access to your Cache Cluster. By default, network access is turned off to your Cache Clusters. If you want your applications to access your Cache Cluster, you must explicitly enable access from hosts in specific Amazon EC2 security groups. After ingress rules are configured, the same rules apply to all Cache Clusters associated with that Cache Security Group.

To allow network access to your Cache Cluster, create a Cache Security Group and use the Authorize Cache Security Group Ingress API or CLI command to authorize the desired Amazon EC2 security group (which in turn specifies the Amazon EC2 instances allowed). IP-range based access control is currently not enabled for Cache Clusters. All clients to a Cache Cluster must be within the Amazon EC2 network, and authorized via Cache Security Groups.

Amazon ElastiCache for Redis provides backup and restore functionality, where you can create a snapshot of your entire Redis cluster as it exists at a specific point in time. You can schedule automatic, recurring daily snapshots, or you can create a manual snapshot at any time. For automatic snapshots, you specify a retention period; manual snapshots are retained until you delete them. The snapshots are stored in Amazon S3 with high durability, and can be used for warm starts, backups, and archiving.

Application Services

AWS offers a variety of managed services to use with your applications, including services that provide application streaming, queueing, push notification, email delivery, search, and transcoding.

Amazon Simple Queue Service (Amazon SQS) Security

Amazon SQS is a highly reliable, scalable message queuing service that enables asynchronous message-based communication between distributed components of an application. The components can be computers or Amazon EC2 instances or a combination of both. With Amazon SQS, you can send any number of messages to an Amazon SQS queue at any time from any component. The messages can be retrieved from the same component or a different one, right away or at a later time (within 14 days). Messages are highly durable; each message is persistently stored in highly available, highly reliable queues. Multiple processes can read/write from/to an Amazon SQS queue at the same time without interfering with each other.

Data Access Amazon SQS access is granted based on an AWS account or a user created with AWS IAM. After it is authenticated, the AWS account has full access to all user operations. An IAM user, however, only has access to the operations and queues for which they have been

granted access via policy. By default, access to each individual queue is restricted to the AWS account that created it. However, you can allow other access to a queue, using either an Amazon SQS-generated policy or a policy you write.

Encryption Amazon SQS is accessible via SSL-encrypted endpoints. The encrypted endpoints are accessible from both the Internet and from within Amazon EC2. Data stored within Amazon SQS is not encrypted by AWS; however, the user can encrypt data before it is uploaded to Amazon SQS, provided that the application using the queue has a means to decrypt the message when it's retrieved. Encrypting messages before sending them to Amazon SQS helps protect against access to sensitive customer data by unauthorized persons, including AWS.

Amazon Simple Notification Service (Amazon SNS) Security

Amazon SNS is a web service that makes it easy to set up, operate, and send notifications from the cloud. It provides developers with a highly scalable, flexible, and cost-effective capability to publish messages from an application and immediately deliver them to subscribers or other applications. Amazon SNS provides a simple web services interface that can be used to create topics that customers want to notify applications (or people) about, subscribe clients to these topics, publish messages, and have these messages delivered over clients' protocol of choice (for example, HTTP/HTTPS, email).

Amazon SNS delivers notifications to clients using a push mechanism that eliminates the need to check or poll for new information and updates periodically. Amazon SNS can be leveraged to build highly reliable, event-driven workflows and messaging applications without the need for complex middleware and application management. The potential uses for Amazon SNS include monitoring applications, workflow systems, time-sensitive information updates, mobile applications, and many others.

Data Access Amazon SNS provides access control mechanisms so that topics and messages are secured against unauthorized access. Topic owners can set policies for a topic that restricts who can publish or subscribe to a topic. Additionally, topic owners can encrypt transmission by specifying that the delivery mechanism must be HTTPS. Amazon SNS access is granted based on an AWS account or a user created with AWS IAM. After it is authenticated, the AWS account has full access to all user operations. An IAM user, however, only has access to the operations and topics for which they have been granted access via policy. By default, access to each individual topic is restricted to the AWS account that created it. However, you can allow other access to Amazon SNS, using either an Amazon SNS-generated policy or a policy you write.

Analytics Services

AWS provides cloud-based analytics services to help you process and analyze any volume of data, whether your need is for managed Hadoop clusters, real-time streaming data, petabyte scale data warehousing, or orchestration.

Amazon Elastic MapReduce (Amazon EMR) Security

Amazon Elastic MapReduce (Amazon EMR) is a managed web service you can use to run Hadoop clusters that process vast amounts of data by distributing the work and data among

several servers. It uses an enhanced version of the Apache Hadoop framework running on the web-scale infrastructure of Amazon EC2 and Amazon S3. You simply upload your input data and a data processing application into Amazon S3. Amazon EMR then launches the number of Amazon EC2 instances you specify. The service begins the job flow execution while pulling the input data from Amazon S3 into the launched Amazon EC2 instances. After the job flow is finished, Amazon EMR transfers the output data to Amazon S3, where you can then retrieve it or use it as input in another job flow.

When launching job flows on your behalf, Amazon EMR sets up two Amazon EC2 security groups: one for the master nodes and another for the slaves. The master security group has a port open for communication with the service. It also has the SSH port open to allow you to SSH into the instances using the key specified at startup. The slaves start in a separate security group, which only allows interaction with the master instance. By default, both security groups are set up to not allow access from external sources, including Amazon EC2 instances belonging to other customers. Because these are security groups within your account, you can reconfigure them using the standard EC2 tools or dashboard. To protect customer input and output datasets, Amazon EMR transfers data to and from Amazon S3 using SSL.

Amazon EMR provides several ways to control access to the resources of your cluster. You can use AWS IAM to create user accounts and roles and configure permissions that control which AWS features those users and roles can access. When you launch a cluster, you can associate an Amazon EC2 key pair with the cluster, which you can then use when you connect to the cluster using SSH. You can also set permissions that allow users other than the default Hadoop user to submit jobs to your cluster.

By default, if an IAM user launches a cluster, that cluster is hidden from other IAM users on the AWS account. This filtering occurs on all Amazon EMR interfaces (the AWS Management Console, CLI, API, and SDKs) and helps prevent IAM users from accessing and inadvertently changing clusters created by other IAM users.

For an additional layer of protection, you can launch the Amazon EC2 instances of your Amazon EMR cluster into an Amazon VPC, which is like launching it into a private subnet. This allows you to control access to the entire subnet. You can also launch the cluster into an Amazon VPC and enable the cluster to access resources on your internal network using a VPN connection. You can encrypt the input data before you upload it to Amazon S3 using any common data encryption tool. If you do encrypt the data before it is uploaded, you then need to add a decryption step to the beginning of your job flow when Amazon EMR fetches the data from Amazon S3.

Amazon Kinesis Security

Amazon Kinesis is a managed service designed to handle real-time streaming of big data. It can accept any amount of data, from any number of sources, scaling up and down as needed. You can use Amazon Kinesis in situations that call for large-scale, real-time data ingestion and processing, such as server logs, social media, or market data feeds, and web clickstream data. Applications read and write data records to Amazon Kinesis in streams. You can create any number of Amazon Kinesis streams to capture, store, and transport data.

You can control logical access to Amazon Kinesis resources and management functions by

creating users under your AWS account using AWS IAM, and controlling which Amazon Kinesis operations these users have permission to perform. To facilitate running your producer or consumer applications on an Amazon EC2 instance, you can configure that instance with an IAM role. That way, AWS credentials that reflect the permissions associated with the IAM role are made available to applications on the instance, which means you don't have to use your long-term AWS security credentials. Roles have the added benefit of providing temporary credentials that expire within a short timeframe, which adds an additional measure of protection.

The Amazon Kinesis API is only accessible via an SSL-encrypted endpoint (`kinesis.us-east-1.amazonaws.com`) to help ensure secure transmission of your data to AWS. You must connect to that endpoint to access Amazon Kinesis, but you can then use the API to direct Amazon Kinesis to create a stream in any AWS region.

Deployment and Management Services

AWS provides a variety of tools to help with the deployment and management of your applications. This includes services that allow you to create individual user accounts with credentials for access to AWS services. It also includes services for creating and updating stacks of AWS resources, deploying applications on those resources, and monitoring the health of those AWS resources. Other tools help you manage cryptographic keys using HSMs and log AWS API activity for security and compliance purposes.

AWS Identity and Access Management (IAM) Security

AWS IAM allows you to create multiple users and manage the permissions for each of these users within your AWS account. A user is an identity (within an AWS account) with unique security credentials that can be used to access AWS Cloud services. IAM eliminates the need to share passwords or keys and makes it easy to enable or disable a user's access as appropriate.

AWS IAM enables you to implement security best practices, such as least privilege, by granting unique credentials to every user within your AWS account and only granting permission to access the AWS Cloud services and resources required for the users to perform their jobs. IAM is secure by default; new users have no access to AWS until permissions are explicitly granted.

AWS IAM is also integrated with AWS Marketplace so that you can control who in your organization can subscribe to the software and services offered in AWS Marketplace. Because subscribing to certain software in AWS Marketplace launches an Amazon EC2 instance to run the software, this is an important access control feature. Using IAM to control access to AWS Marketplace also enables AWS account owners to have fine-grained control over usage and software costs.

AWS IAM enables you to minimize the use of your AWS account credentials. After you create IAM user accounts, all interactions with AWS Cloud services and resources should occur with IAM user security credentials.

Roles An *IAM role* uses temporary security credentials to allow you to delegate access to users or services that normally don't have access to your AWS resources. A role is a set of permissions to access specific AWS resources, but these permissions are not tied to a specific

IAM user or group. An authorized entity (for example, mobile user or Amazon EC2 instance) assumes a role and receives temporary security credentials for authenticating to the resources defined in the role. Temporary security credentials provide enhanced security due to their short lifespan (the default expiration is 12 hours) and the fact that they cannot be reused after they expire. This can be particularly useful in providing limited, controlled access in certain situations:

Federated (Non-AWS) User Access *Federated users* are users (or applications) who do not have AWS accounts. With roles, you can give them access to your AWS resources for a limited amount of time. This is useful if you have non-AWS users that you can authenticate with an external service, such as Microsoft Active Directory, Lightweight Directory Access Protocol (LDAP), or Kerberos. The temporary AWS credentials used with the roles provide identity federation between AWS and your non-AWS users in your corporate identity and authorization system.

Security Assertion Markup Language (SAML) 2.0 If your organization supports SAML 2.0, you can create trust between your organization as an Identity Provider (IdP) and other organizations as service providers. In AWS, you can configure AWS as the service provider and use SAML to provide your users with federated Single-Sign On (SSO) to the AWS Management Console or to get federated access to call AWS APIs.

Roles are also useful if you create a mobile or web-based application that accesses AWS resources. AWS resources require security credentials for programmatic requests; however, you shouldn't embed long-term security credentials in your application because they are accessible to the application's users and can be difficult to rotate. Instead, you can let users sign in to your application using Login with Amazon, Facebook, or Google and then use their authentication information to assume a role and get temporary security credentials.

Cross-Account Access For organizations that use multiple AWS accounts to manage their resources, you can set up roles to provide users who have permissions in one account to access resources under another account. For organizations that have personnel who only rarely need access to resources under another account, using roles helps to ensure that credentials are provided temporarily and only as needed.

Applications Running on EC2 Instances That Need to Access AWS Resources If an application runs on an Amazon EC2 instance and needs to make requests for AWS resources, such as Amazon S3 buckets or a DynamoDB table, it must have security credentials. Using roles instead of creating individual IAM accounts for each application on each instance can save significant time for customers who manage a large number of instances or an elastically scaling fleet using AWS Auto Scaling.

The temporary credentials include a security token, an Access Key ID, and a Secret Access Key. To give a user access to certain resources, you distribute the temporary security credentials to the user to whom you are granting temporary access. When the user makes calls to your resources, the user passes in the token and Access Key ID and signs the request with the Secret Access Key. The token will not work with different access keys.

The use of temporary credentials provides additional protection for you because you don't have to manage or distribute long-term credentials to temporary users. In addition, the temporary credentials get automatically loaded to the target instance so you don't have to embed them somewhere unsafe like your code. Temporary credentials are automatically rotated or changed multiple times a day without any action on your part and are stored securely by default.

Mobile Services

AWS mobile services make it easier for you to build, ship, run, monitor, optimize, and scale cloud-powered applications for mobile devices. These services also help you authenticate users to your mobile application, synchronize data, and collect and analyze application usage.

Amazon Cognito Security

Amazon Cognito provides identity and sync services for mobile and web-based applications. It simplifies the task of authenticating users and storing, managing, and syncing their data across multiple devices, platforms, and applications. It provides temporary, limited-privilege credentials for both authenticated and unauthenticated users without having to manage any back-end infrastructure.

Amazon Cognito works with well-known identity providers like Google, Facebook, and Amazon to authenticate end users of your mobile and web applications. You can take advantage of the identification and authorization features provided by these services instead of having to build and maintain your own. Your application authenticates with one of these identity providers using the provider's SDK. After the end user is authenticated with the provider, an OAuth or OpenID Connect token returned from the provider is passed by your application to Amazon Cognito, which returns a new Amazon Cognito ID for the user and a set of temporary, limited-privilege AWS credentials.

To begin using Amazon Cognito, you create an identity pool through the Amazon Cognito console. The *identity pool* is a store of user identity information that is specific to your AWS account. During the creation of the identity pool, you will be asked to create a new IAM role or pick an existing one for your end users. An *IAM role* is a set of permissions to access specific AWS resources, but these permissions are not tied to a specific IAM user or group. An authorized entity (for example, mobile user, Amazon EC2 instance) assumes a role and receives temporary security credentials for authenticating to the AWS resources defined in the role. Temporary security credentials provide enhanced security due to their short lifespan (the default expiration is 12 hours) and the fact that they cannot be reused after they expire.

The role you select has an impact on which AWS Cloud services your end users will be able to access with the temporary credentials. By default, Amazon Cognito creates a new role with limited permissions; end users only have access to the Amazon Cognito Sync service and Amazon Mobile Analytics. If your application needs access to other AWS resources, such as Amazon S3 or Amazon DynamoDB, you can modify your roles directly from the IAM console.

With Amazon Cognito, there is no need to create individual AWS accounts or even IAM accounts for every one of your web/mobile application end users who will need to access your AWS resources. In conjunction with IAM roles, mobile users can securely access AWS resources and application features and even save data to the AWS Cloud without having to create an account or log in. If they choose to create an account or log in later, Amazon Cognito will merge data and identification information.

Because Amazon Cognito stores data locally and also in the service, your end users can continue to interact with their data even when they are offline. Their offline data may be stale, but they can immediately retrieve anything they put into the dataset whether or not they are online. The client SDK manages a local SQLite store so that the application can work even when it is not connected. The SQLite store functions as a cache and is the target of all read and write operations. Amazon Cognito's sync facility compares the local version of the

data to the cloud version and pushes up or pulls down deltas as needed. Note that in order to sync data across devices, your identity pool must support authenticated identities. Unauthenticated identities are tied to the device, so unless an end user authenticates, no data can be synced across multiple devices.

With Amazon Cognito, your application communicates directly with a supported public identity provider (Amazon, Facebook, or Google) to authenticate users. Amazon Cognito does not receive or store user credentials, only the OAuth or OpenID Connect token received from the identity provider. After Amazon Cognito receives the token, it returns a new Amazon Cognito ID for the user and a set of temporary, limited-privilege AWS credentials. Each Amazon Cognito identity has access only to its own data in the sync store, and this data is encrypted when stored. In addition, all identity data is transmitted over HTTPS. The unique Amazon Cognito identifier on the device is stored in the appropriate secure location. For example on iOS, the Amazon Cognito identifier is stored in the iOS keychain. User data is cached in a local SQLite database within the application's sandbox; if you require additional security, you can encrypt this identity data in the local cache by implementing encryption in your application.

Applications

AWS applications are managed services that enable you to provide your users with secure, centralized storage and work areas in the cloud.

Amazon WorkSpaces Security

Amazon WorkSpaces is a managed desktop service that allows you to quickly provision cloud-based desktops for your users. Simply choose a Windows 7 bundle that best meets the needs of your users and the number of WorkSpaces that you want to launch. After the WorkSpaces are ready, users receive an email informing them where they can download the relevant client and log in to their WorkSpace. They can then access their cloud-based desktops from a variety of endpoint devices, including PCs, laptops, and mobile devices. However, your organization's data is never sent to or stored on the end-user device because Amazon WorkSpaces uses PC-over-IP (PCoIP), which provides an interactive video stream without transmitting actual data. The PCoIP protocol compresses, encrypts, and encodes the users' desktop computing experience and transmits as pixels only across any standard IP network to end-user devices.

In order to access their WorkSpace, users must sign in using a set of unique credentials or their regular Active Directory credentials. When you integrate Amazon WorkSpaces with your corporate Active Directory, each WorkSpace joins your Active Directory domain and can be managed just like any other desktop in your organization. This means that you can use Active Directory Group Policies to manage your users WorkSpaces to specify configuration options that control the desktop. If you choose not to use Active Directory or other type of on-premises directory to manage your user WorkSpaces, you can create a private cloud directory within Amazon WorkSpaces that you can use for administration.

To provide an additional layer of security, you can also require the use of MFA upon sign-in in the form of a hardware or software token. Amazon WorkSpaces supports MFA using an on-premises Remote Authentication Dial In User Service (RADIUS) server or any security provider that supports RADIUS authentication. It currently supports the PAP, CHAP, MS-

CHAP1, and MS-CHAP2 protocols, along with RADIUS proxies.

Each WorkSpace resides on its own Amazon EC2 instance within an Amazon VPC. You can create WorkSpaces in an Amazon VPC you already own or have the Amazon WorkSpaces service create one for you automatically using the Amazon WorkSpaces Quick Start option. When you use the Quick Start option, Amazon WorkSpaces not only creates the Amazon VPC, but it also performs several other provisioning and configuration tasks for you, such as creating an Internet Gateway for the Amazon VPC, setting up a directory within the Amazon VPC that is used to store user and WorkSpace information, creating a directory administrator account, creating the specified user accounts and adding them to the directory, and creating the Amazon WorkSpaces instances. Or the Amazon VPC can be connected to an on-premises network using a secure VPN connection to allow access to an existing on-premises Active Directory and other intranet resources. You can add a security group that you create in your Amazon VPC to all of the WorkSpaces that belong to your Active Directory. This allows you to control network access from Amazon WorkSpaces in your Amazon VPC to other resources in your Amazon VPC and on-premises network.

Persistent storage for Amazon WorkSpaces is provided by Amazon EBS and is automatically backed up twice a day to Amazon S3. If Amazon WorkSpaces Sync is enabled on a WorkSpace, the folder a user chooses to sync will be continuously backed up and stored in Amazon S3. You can also use Amazon WorkSpaces Sync on a Mac or PC to sync documents to or from your WorkSpace so that you can always have access to your data regardless of the desktop computer you are using.

Because it is a managed service, AWS takes care of several security and maintenance tasks like daily backups and patching. Updates are delivered automatically to your WorkSpaces during a weekly maintenance window. You can control how patching is configured for a user's WorkSpace. By default, Windows Update is turned on, but you have the ability to customize these settings or use an alternative patch management approach if you desire. For the underlying OS, Windows Update is enabled by default on Amazon WorkSpaces and configured to install updates on a weekly basis. You can use an alternative patching approach or configure Windows Update to perform updates at a time of your choosing. You can use IAM to control who on your team can perform administrative functions like creating or deleting WorkSpaces or setting up user directories. You can also set up a WorkSpace for directory administration, install your favorite Active Directory administration tools, and create organizational units and Group Policies in order to apply Active Directory changes more easily for all of your Amazon WorkSpaces users.

Summary

In this chapter, you learned that the first priority at AWS is Cloud security. Security within AWS is based on a “defense in depth” model where no one, single element is used to secure systems on AWS. Rather, AWS uses a multitude of elements—each acting at different layers of a system—in total to secure the system. AWS is responsible for some layers of this model, and customers are responsible for others. AWS also offers security tools and features of services for customers to use at their discretion. Several of these concepts, tools, and features were discussed in this chapter.

Security Model

The shared responsibility model is the security model where AWS is responsible for the security of the underlying cloud infrastructure, and the customer is responsible for securing workloads deployed in AWS. Customers benefit from a data center and network architecture built to satisfy the requirements of AWS most security-sensitive customers. This means that customers get a resilient infrastructure, designed for high security, without the capital outlay and operational overhead of a traditional data center.

Account Level Security

AWS credentials help ensure that only authorized users and processes access your AWS account and resources. AWS uses several types of credentials for authentication. These include passwords, cryptographic keys, digital signatures, and certificates. AWS also provides the option of requiring MFA to log in to your AWS account or IAM user accounts.

Passwords are required to access your AWS account, individual IAM user accounts, AWS Discussion Forums, and the AWS Support Center. You specify the password when you first create the account, and you can change it at any time by going to the Security Credentials page.

AWS MFA is an additional layer of security for accessing AWS Cloud services. When you enable this optional feature, you will need to provide a six-digit, single-use code in addition to your standard user name and password credentials before access is granted to your AWS account settings or AWS Cloud services and resources. You get this single-use code from an authentication device that you keep in your physical possession. This is multi-factor because more than one authentication factor is checked before access is granted: a password (something you know) and the precise code from your authentication device (something you have). An MFA device uses a software application that generates six-digit authentication codes that are compatible with the TOTP standard, as described in RFC 6238.

Access Keys are created by AWS IAM and delivered as a pair: the Access Key ID (AKI) and the Secret Access Key (SAK). AWS requires that all API requests be signed by the SAK; that is, they must include a digital signature that AWS can use to verify the identity of the requestor. You calculate the digital signature using a cryptographic hash function. If you use any of the AWS SDKs to generate requests, the digital signature calculation is done for you. The most recent version of the digital signature calculation process at the time of this writing is Signature Version 4, which calculates the signature using the HMAC-SHA-256 protocol.

AWS CloudTrail is a web service that records API calls made on your account and delivers log files to your Amazon S3 bucket. AWS CloudTrail's benefit is visibility into account activity by recording API calls made on your account.

Service-Specific Security

In addition to the Shared Responsibility Model and Account Level security, AWS offers security features for each of the services it provides. These security features are outlined below by technology domain.

Compute

Amazon Elastic Compute Cloud (Amazon EC2) Amazon EC2 supports RSA 2048 SSH-2 Key pairs for gaining first access to an Amazon EC2 instance. On a Linux instance, access is granted through showing possession of the SSH private key. On a Windows instance, access is granted by showing possession of the SSH private key in order to decrypt the administrator password.

Amazon Elastic Block Store (Amazon EBS) Data stored in Amazon EBS volumes is redundantly stored in multiple physical locations within the same Availability Zone as part of normal operation of that service and at no additional charge. AWS provides the ability to encrypt Amazon EBS volumes and their snapshots with AES-256. The encryption occurs on the servers that host the Amazon EC2 instances, providing encryption of data as it moves between Amazon EC2 instances and Amazon EBS storage.

Networking

Elastic Load Balancing Elastic Load Balancing configures your load balancer with a pre-defined cipher set that is used for TLS negotiation when a connection is established between a client and your load balancer. The pre-defined cipher set provides compatibility with a broad range of clients and uses strong cryptographic algorithms. Elastic Load Balancing allows you to identify the originating IP address of a client connecting to your servers, whether you're using HTTPS or TCP load balancing.

Amazon Virtual Private Cloud (Amazon VPC) Amazon VPC enables you to create an isolated portion of the AWS Cloud and launch Amazon EC2 instances that have private (RFC 1918) addresses in the range of your choice. Security features within Amazon VPC include security groups, network ACLs, routing tables, and external gateways. Each of these items is complementary to providing a secure, isolated network that can be extended through selective enabling of direct Internet access or private connectivity to another network.

Amazon CloudFront Amazon CloudFront gives customers an easy way to distribute content to end users with low latency and high data transfer speeds. It delivers dynamic, static, and streaming content using a global network of edge locations. To control access to the original copies of your objects in Amazon S3, Amazon CloudFront allows you to create one or more Origin Access Identities and associate these with your distributions. To control who can download objects from Amazon CloudFront edge locations, the service uses a signed-URL verification system.

Storage

Amazon Simple Storage Service (Amazon S3) Amazon S3 allows you to upload and retrieve data at any time, from anywhere on the web. Access to data stored in Amazon S3 is restricted by default; only bucket and object owners have access to the Amazon S3 resources they create. You can securely upload and download data to Amazon S3 via the SSL-encrypted endpoints. Amazon S3 supports several methods to encrypt data at rest.

Amazon Glacier Amazon Glacier service provides low-cost, secure, and durable storage. You can securely upload and download data to Amazon Glacier via the SSL-encrypted endpoints, and the service automatically encrypts the data using AES-256 and stores it durably in an immutable form.

AWS Storage Gateway AWS Storage Gateway service connects your on-premises software appliance with cloud-based storage to provide seamless and secure integration between your IT environment and AWS storage infrastructure. Data is asynchronously transferred from your on-premises storage hardware to AWS over SSL and stored encrypted in Amazon S3 using AES-256.

Database

Amazon DynamoDB Amazon DynamoDB is a managed NoSQL database service that provides fast and predictable performance with seamless scalability. You can control access at the database level by creating database-level permissions that allow or deny access to items (rows) and attributes (columns) based on the needs of your application.

Amazon Relational Database Service (RDS) Amazon RDS allows you to quickly create a relational DB Instance and flexibly scale the associated compute resources and storage capacity to meet application demand. You can control Amazon RDS DB Instance access via DB security groups, which act like a firewall controlling network access to your DB Instance. Database security groups default to deny all access mode, and customers must specifically authorize network ingress. Amazon RDS is supported within an Amazon VPC, and for Multi-AZ deployments, defining a subnet for all Availability Zones in a region will allow Amazon RDS to create a new standby in another Availability Zone should the need arise. You can encrypt connections between your application and your DB Instance using SSL, and you can encrypt data at rest within Amazon RDS instances for all database engines.

Amazon Redshift Amazon Redshift is a petabyte-scale SQL data warehouse service that runs on highly optimized and managed AWS compute and storage resources. The service enables you to configure firewall rules (security groups) to control network access to your data warehouse cluster. Database users are named user accounts that can connect to a database and are authenticated when they log in to Amazon Redshift. In Amazon Redshift, you grant database user permissions on a per-cluster basis instead of on a per-table basis. You may choose for Amazon Redshift to store all data in user-created tables in an encrypted format using hardware-accelerated AES-256 block encryption keys. This includes all data written to disk and also any backups. Amazon Redshift uses a four-tier, key-based architecture for encryption. These keys consist of data encryption keys, a database key, a cluster key, and a master key.

Amazon ElastiCache Amazon ElastiCache is a web service that makes it easy to set up, manage, and scale distributed in-memory cache environments in the cloud. Amazon ElastiCache allows you to control access to your Cache Clusters using Cache Security Groups.

A Cache Security Group acts like a firewall, controlling network access to your Cache Cluster.

Application Services

Amazon Simple Queue Service (SQS) Amazon SQS is a highly reliable, scalable message queuing service that enables asynchronous message-based communication between distributed components of an application. Amazon SQS access is granted based on an AWS account or a user created with AWS IAM. Data stored within Amazon SQS is not encrypted by AWS; however, the user can encrypt data before it is uploaded to Amazon SQS, provided that the application using the queue has a means to decrypt the message when it's retrieved.

Amazon Simple Notification Service (SNS) Amazon SNS is a web service that makes it easy to set up, operate, and send notifications from the cloud. It provides developers with a highly scalable, flexible, and cost-effective capability to publish messages from an application and immediately deliver them to subscribers or other applications. Amazon SNS allows topic owners to set policies for a topic that restrict who can publish or subscribe to a topic.

Analytics

Amazon Elastic MapReduce (Amazon EMR) Amazon EMR is a managed web service you can use to run Hadoop clusters that process vast amounts of data by distributing the work and data among several servers. When launching job flows on your behalf, Amazon EMR sets up two Amazon EC2 security groups: one for the master nodes and another for the slaves. You can launch the Amazon EC2 instances of your Amazon EMR cluster into an Amazon VPC, which is like launching it into a private subnet. You can encrypt the input data before you upload it to Amazon S3 using any common data encryption tool. If you do encrypt the data before it is uploaded, you then need to add a decryption step to the beginning of your job flow when Amazon EMR fetches the data from Amazon S3.

Amazon Kinesis Amazon Kinesis is a managed service designed to handle real-time streaming of big data. You can control logical access to Amazon Kinesis resources and management functions by creating users under your AWS account using AWS IAM and controlling which Amazon Kinesis operations these users have permission to perform. The Amazon Kinesis API is only accessible via an SSL-encrypted endpoint to help ensure secure transmission of your data to AWS.

Deployment and Management

AWS Identity and Access Management (IAM) AWS IAM allows you to create multiple users and manage the permissions for each of these users within your AWS account. A user is an identity (within an AWS account) with unique security credentials that can be used to access AWS Cloud services. IAM is secure by default; new users have no access to AWS until permissions are explicitly granted. A role is a set of permissions to access specific AWS resources, but these permissions are not tied to a specific IAM user or group.

Mobile Services

Amazon Cognito Amazon Cognito provides identity and sync services for mobile and web-based applications. Your application authenticates with one of the well-known identity providers such as Google, Facebook, and Amazon using the provider's SDK. After the end user is authenticated with the provider, an OAuth or OpenID Connect token returned from

the provider is passed by your application to Amazon Cognito, which returns a new Amazon Cognito ID for the user and a set of temporary, limited-privilege AWS credentials.

Applications

Amazon Workspaces Amazon WorkSpaces is a managed desktop service that allows you to quickly provision cloud-based desktops for your users. Amazon WorkSpaces uses PCoIP, which provides an interactive video stream without transmitting actual data. The PCoIP protocol compresses, encrypts, and encodes the user's desktop computing experience and transmits as pixels only across any standard IP network to end-user devices. In order to access their WorkSpace, users must sign in using a set of unique credentials or their regular Active Directory credentials. You can also require the use of MFA upon sign-in in the form of a hardware or software token. Amazon WorkSpaces supports MFA using an on-premises RADIUS server or any security provider that supports RADIUS authentication. It currently supports the PAP, CHAP, MS-CHAP1, and MS-CHAP2 protocols, along with RADIUS proxies.

Exam Essentials

Understand the shared responsibility model. AWS is responsible for securing the underlying infrastructure that supports the cloud, and you're responsible for anything you put on the cloud or connect to the cloud.

Understand regions and Availability Zones. Each region is completely independent. Each region is designed to be completely isolated from the other regions. This achieves the greatest possible fault tolerance and stability. Regions are a collection of Availability Zones. Each Availability Zone is isolated, but the Availability Zones in a region are connected through low-latency links.

Understand High-Availability System Design within AWS. You should architect your AWS usage to take advantage of multiple regions and Availability Zones. Distributing applications across multiple Availability Zones provides the ability to remain resilient in the face of most failure modes, including natural disasters or system failures.

Understand the network security of AWS. Network devices, including firewall and other boundary devices, are in place to monitor and control communications at the external boundary of the network and at key internal boundaries within the network. These boundary devices employ rule sets, ACLs, and configurations to enforce the flow of information to specific information system services.

AWS has strategically placed a limited number of access points to the cloud to allow for a more comprehensive monitoring of inbound and outbound communications and network traffic. These customer access points are called API endpoints, and they allow HTTPS access, which allows you to establish a secure communication session with your storage or compute instances within AWS.

Amazon EC2 instances cannot send spoofed network traffic. The AWS-controlled, host-based firewall infrastructure will not permit an instance to send traffic with a source IP or MAC address other than its own.

Unauthorized port scans by Amazon EC2 customers are a violation of the AWS Acceptable Use Policy. Violations of the AWS Acceptable Use Policy are taken seriously, and every reported violation is investigated.

It is not possible for an Amazon EC2 instance running in promiscuous mode to receive or "sniff" traffic that is intended for a different virtual instance.

Understand the use of credentials on AWS. AWS employs several credentials in order to positively identify a person or authorize an API call to the platform. Credentials include:

- Passwords
- AWS root account or IAM user account login to the AWS Management Console
- Multi-Factor Authentication (MFA)
- AWS root account or IAM user account login to the AWS Management Console
- Access Keys

- Digitally signed requests to AWS APIs (using the AWS SDK, CLI, or REST/Query APIs)

Understand the proper use of access keys. Because access keys can be misused if they fall into the wrong hands, AWS encourages you to save them in a safe place and not to embed them in your code. For customers with large fleets of elastically-scaling Amazon EC2 instances, the use of IAM roles can be a more secure and convenient way to manage the distribution of access keys.

Understand the value of AWS CloudTrail. AWS CloudTrail is a web service that records API calls made on your account and delivers log files to your Amazon S3 bucket. AWS CloudTrail's benefit is visibility into account activity by recording API calls made on your account.

Understand the security features of Amazon EC2. Amazon EC2 uses public-key cryptography to encrypt and decrypt login information. Public-key cryptography uses a public key to encrypt a piece of data, such as a password, and then the recipient uses the private key to decrypt the data. The public and private keys are known as a key pair.

To log in to your instance, you must create a key pair, specify the name of the key pair when you launch the instance, and provide the private key when you connect to the instance. Linux instances have no password, and you use a key pair to log in using SSH. With Windows instances, you use a key pair to obtain the administrator password and then log in using RDP.

A security group acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group.

Understand AWS use of encryption of data in transit. All service endpoints support encryption of data in transit via HTTPS.

Know which services offer encryption of data at rest as a feature. The following services offer a feature to encrypt data at rest:

- Amazon S3
- Amazon EBS
- Amazon Glacier
- AWS Storage Gateway
- Amazon RDS
- Amazon Redshift
- Amazon WorkSpaces

Exercises

The best way to become familiar with the security features of AWS is to do the exercises for each chapter and inspect the security features offered by the service. Take a look at this list of AWS Cloud services covered in different chapters and their security features:

Chapter 6, AWS IAM

- Exercise 6.1: Create an IAM Group
- Exercise 6.2: Create a Customized Sign-In Link and Password Policy
- Exercise 6.3: Create an IAM User
- Exercise 6.4: Create and Use an IAM Role
- Exercise 6.5: Rotate Keys
- Exercise 6.6: Set Up MFA
- Exercise 6.7: Resolve Conflicting Permissions

Chapter 3, Amazon EC2

- Exercise 3.1: Launch and Connect to a Linux Instance
- Exercise 3.2: Launch a Windows Instance with Bootstrapping

Chapter 3, Amazon EBS

- Exercise 3.8: Launch an Encrypted Volume

Chapter 2, Amazon S3

- Exercise 2.1: Create an Amazon Simple Storage Service (Amazon S3) Bucket
- Exercise 2.2: Upload, Make Public, Rename, and Delete Objects in Your Bucket

Chapter 4, Amazon VPC

- Exercise 4.1: Create a Custom Amazon VPC
- Exercise 4.2: Create Two Subnets for Your Custom Amazon VPC
- Exercise 4.3: Connect Your Amazon VPC to the Internet and Establish Routing
- Exercise 4.4: Launch an Amazon EC2 Instance and Test the Connection to the Internet.

Chapter 7, Amazon RDS

- Exercise 7.1: Create a MySQL Amazon RDS Instance
- Exercise 7.2: Simulate a Failover from One AZ to Another

Review Questions

1. Which is an operational process performed by AWS for data security?
   A. Advanced Encryption Standard (AES)-256 encryption of data stored on any shared storage device
   B. Decommissioning of storage devices using industry-standard practices
   C. Background virus scans of Amazon Elastic Block Store (Amazon EBS) volumes and Amazon EBS snapshots
   D. Replication of data across multiple AWS regions
   E. Secure wiping of Amazon EBS data when an Amazon EBS volume is unmounted
2. You have launched a Windows Amazon Elastic Compute Cloud (Amazon EC2) instance and specified an Amazon EC2 key pair for the instance at launch. Which of the following accurately describes how to log in to the instance?
   A. Use the Amazon EC2 key pair to securely connect to the instance via Secure Shell (SSH).
   B. Use your AWS Identity and Access Management (IAM) user X.509 certificate to log in to the instance.
   C. Use the Amazon EC2 key pair to decrypt the administrator password and then securely connect to the instance via Remote Desktop Protocol (RDP) as the administrator.
   D. A key pair is not needed. Securely connect to the instance via RDP.
3. A Database security group controls network access to a database instance that is inside a Virtual Private Cloud (VPC) and by default allows access from?
   A. Access from any IP address for the standard ports that the database uses is provided by default.
   B. Access from any IP address for any port is provided by default in the DB security group.
   C. No access is provided by default, and any access must be explicitly added with a rule to the DB security group.
   D. Access for the database connection string is provided by default in the DB security group.
4. Which encryption algorithm is used by Amazon Simple Storage Service (Amazon S3) to encrypt data at rest with Service-Side Encryption (SSE)?
   A. Advanced Encryption Standard (AES)-256
   B. RSA 1024
   C. RSA 2048
   D. AES-128

5. How many access keys may an AWS Identity and Access Management (IAM) user have active at one time?
   A. 0
   B. 1
   C. 2
   D. 3
6. Which of the following is the name of the security model employed by AWS with its customers?
   A. The shared secret model
   B. The shared responsibility model
   C. The shared secret key model
   D. The secret key responsibility model
7. Which of the following describes the scheme used by an Amazon Redshift cluster leveraging AWS Key Management Service (AWS KMS) to encrypt data-at-rest?
   A. Amazon Redshift uses a one-tier, key-based architecture for encryption.
   B. Amazon Redshift uses a two-tier, key-based architecture for encryption.
   C. Amazon Redshift uses a three-tier, key-based architecture for encryption.
   D. Amazon Redshift uses a four-tier, key-based architecture for encryption.
8. Which of the following Elastic Load Balancing options ensure that the load balancer determines which cipher is used for a Secure Sockets Layer (SSL) connection?
   A. Client Server Cipher Suite
   B. Server Cipher Only
   C. First Server Cipher
   D. Server Order Preference
9. Which technology does Amazon WorkSpaces use to provide data security?
   A. Secure Sockets Layer (SSL)/Transport Layer Security (TLS)
   B. Advanced Encryption Standard (AES)-256
   C. PC-over-IP (PCoIP)
   D. AES-128
10. As a Solutions Architect, how should you architect systems on AWS?
    A. You should architect for least cost.
    B. You should architect your AWS usage to take advantage of Amazon Simple Storage Service's (Amazon S3) durability.
    C. You should architect your AWS usage to take advantage of multiple regions and Availability Zones.

D. You should architect with Amazon Elastic Compute Cloud (Amazon EC2) Auto Scaling to ensure capacity is available when needed.

11. Which security scheme is used by the AWS Multi-Factor Authentication (AWS MFA) token?

    A. Time-Based One-Time Password (TOTP)

    B. Perfect Forward Secrecy (PFC)

    C. Ephemeral Diffie Hellman (EDH)

    D. Split-Key Encryption (SKE)

12. DynamoDB tables may contain sensitive data that needs to be protected. Which of the following is a way for you to protect DynamoDB table content? (Choose 2 answers)

    A. DynamoDB encrypts all data server-side by default so nothing is required.

    B. DynamoDB can store data encrypted with a client-side encryption library solution before storing the data in DynamoDB.

    C. DynamoDB obfuscates all data stored so encryption is not required.

    D. DynamoDB can be used with the AWS Key Management Service to encrypt the data before storing the data in DynamoDB.

    E. DynamoDB should not be used to store sensitive information requiring protection.

13. You have launched an Amazon Linux Elastic Compute Cloud (Amazon EC2) instance into EC2-Classic, and the instance has successfully passed the System Status Check and Instance Status Check. You attempt to securely connect to the instance via Secure Shell (SSH) and receive the response, "WARNING: UNPROTECTED PRIVATE KEY FILE," after which the login fails. Which of the following is the cause of the failed login?

    A. You are using the wrong private key.

    B. The permissions for the private key are too insecure for the key to be trusted.

    C. A security group rule is blocking the connection.

    D. A security group rule has not been associated with the private key.

14. Which of the following public identity providers are supported by Amazon Cognito Identity?

    A. Amazon

    B. Google

    C. Facebook

    D. All of the above

15. Which feature of AWS is designed to permit calls to the platform from an Amazon Elastic Compute Cloud (Amazon EC2) instance without needing access keys placed on the instance?

    A. AWS Identity and Access Management (IAM) instance profile

B. IAM groups

C. IAM roles

D. Amazon EC2 key pairs

16. Which of the following Amazon Virtual Private Cloud (Amazon VPC) elements acts as a stateless firewall?

A. Security group

B. Network Access Control List (ACL)

C. Network Address Translation (NAT) instance

D. An Amazon VPC endpoint

17. Which of the following is the most recent version of the AWS digital signature calculation process?

A. Signature Version 1

B. Signature Version 2

C. Signature Version 3

D. Signature Version 4

18. Which of the following is the name of the feature within Amazon Virtual Private Cloud (Amazon VPC) that allows you to launch Amazon Elastic Compute Cloud (Amazon EC2) instances on hardware dedicated to a single customer?

A. Amazon VPC-based tenancy

B. Dedicated tenancy

C. Default tenancy

D. Host-based tenancy

19. Which of the following describes how Amazon Elastic MapReduce (Amazon EMR) protects access to the cluster?

A. The master node and the slave nodes are launched into an Amazon Virtual Private Cloud (Amazon VPC).

B. The master node supports a Virtual Private Network (VPN) connection from the key specified at cluster launch.

C. The master node is launched into a security group that allows Secure Shell (SSH) and service access, while the slave nodes are launched into a separate security group that only permits communication with the master node.

D. The master node and slave nodes are launched into a security group that allows SSH and service access.

20. To help prevent data loss due to the failure of any single hardware component, Amazon Elastic Block Storage (Amazon EBS) automatically replicates EBS volume data to which of the following?

A. Amazon EBS replicates EBS volume data within the same Availability Zone in a region.

B. Amazon EBS replicates EBS volume data across other Availability Zones within the same region.

C. Amazon EBS replicates EBS volume data across Availability Zones in the same region and in Availability Zones in one other region.

D. Amazon EBS replicates EBS volume data across Availability Zones in the same region and in Availability Zones in every other region.