Overview of AWS Security - Analytics Services, Mobile and Applications Services

June 2016

(Please consult **http://aws.amazon.com/security/** for the latest version of this paper)

Notices

This document is provided for informational purposes only. It represents AWS' current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS' products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Analytics Services

Amazon Web Services 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 utilizes 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. Once 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. Since 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 console, CLI, API, and SDKs—and helps prevent IAM users from accessing and inadvertently changing clusters created by other IAM users. It is useful for clusters that are intended to be viewed by only a single IAM user and the main AWS account. You also have the option to make a cluster visible and accessible to all IAM users under a single AWS account.

For an additional layer of protection, you can launch the EC2 instances of your EMR cluster into an Amazon VPC, which is like launching it into a private subnet. This allows you to control access to the entire subnetwork. You can also launch the cluster into a 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 Elastic MapReduce 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 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 Kinesis streams to capture, store, and transport data. Amazon Kinesis automatically manages the infrastructure, storage, networking, and configuration needed to collect and process your data at the level of throughput your streaming applications need. You don't have to worry about provisioning, deployment, or ongoing-maintenance of hardware, software, or other services to enable real-time capture and storage of large-scale data. Amazon Kinesis also synchronously replicates data across three facilities in an AWS Region, providing high availability and data durability.

In Amazon Kinesis, data records contain a sequence number, a partition key, and a data blob, which is an un-interpreted, immutable sequence of bytes. The Amazon Kinesis service does not inspect, interpret, or change the data in the blob in any way. Data records are accessible for only 24 hours from the time they are added to an Amazon Kinesis stream, and then they are automatically discarded.

Your application is a consumer of an Amazon Kinesis stream, which typically runs on a fleet of Amazon EC2 instances. A Kinesis application uses the Amazon Kinesis Client Library to read from the Amazon Kinesis stream. The Kinesis Client Library takes care of a variety of details for you including failover, recovery, and load balancing, allowing your application to focus on processing the data as it becomes available. After processing the record your consumer code can pass it along to another Kinesis stream; write it to an Amazon S3 bucket, a Redshift data warehouse, or a DynamoDB table; or simply discard it. A connector library is available to help you integrate Kinesis with other AWS services (such as DynamoDB, Redshift, and Amazon S3) as well as third-party products like Apache Storm.

You can control logical access to Kinesis resources and management functions by creating users under your AWS Account using AWS IAM, and controlling which 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. See the Using IAM Guide for more information about IAM roles.

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 Kinesis, but you can then use the API to direct AWS Kinesis to create a stream in any AWS Region

AWS Data Pipeline Security

The AWS Data Pipeline service helps you process and move data between different data sources at specified intervals using data-driven workflows and built-in dependency checking.

When you create a pipeline, you define data sources, preconditions, destinations, processing steps, and an operational schedule. Once you define and activate a pipeline, it will run automatically according to the schedule you specified.

With AWS Data Pipeline, you don't have to worry about checking resource availability, managing inter-task dependencies, retrying transient failures/timeouts in individual tasks, or creating a failure notification system. AWS Data

Pipeline takes care of launching the AWS services and resources your pipeline needs to process your data (e.g., Amazon EC2 or EMR) and transferring the results to storage (e.g., Amazon S3, RDS, DynamoDB, or EMR).

When you use the console, AWS Data Pipeline creates the necessary IAM roles and policies, including a trusted entities list for you. IAM roles determine what your pipeline can access and the actions it can perform. Additionally, when your pipeline creates a resource, such as an EC2 instance, IAM roles determine the EC2 instance's permitted resources and actions. When you create a pipeline, you specify one IAM role that governs your pipeline and another IAM role to govern your pipeline's resources (referred to as a "resource role"), which can be the same role for both. As part of the security best practice of least privilege, we recommend that you consider the minimum permissions necessary for your pipeline to perform work and define the IAM roles accordingly.

Like most AWS services, AWS Data Pipeline also provides the option of secure (HTTPS) endpoints for access via SSL.

Deployment and Management Services

Amazon Web Services 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 hardware security modules (HSMs) and log AWS API activity for security and compliance purposes.

AWS Identity and Access Management (AWS 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 Services. AWS 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 services and resources required for the users to perform their jobs. AWS IAM is secure by default; new users have no access to AWS until permissions are explicitly granted.

AWS IAM is also integrated with the AWS Marketplace, so that you can control who in your organization can subscribe to the software and services offered in the Marketplace. Since subscribing to certain software in the Marketplace launches an EC2 instance to run the software, this is an important access control feature. Using AWS IAM to control access to the 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. Once you create AWS IAM user accounts, all interactions with AWS Services and resources should occur with AWS IAM user security credentials. More information about AWS IAM is available on the AWS website.

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 (e.g., mobile user, 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 life-span (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, 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.

- If your organization supports SAML 2.0 (Security Assertion Markup Language 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 who 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 who have personnel who only rarely need access to resources under another account, using roles helps ensures that credentials are provided temporarily, 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 you are granting temporary access to. 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. How the user passes in the token depends on the API and version of the AWS product the user is making calls to. More information about temporary security credentials is available on the AWS website.

The use of temporary credentials means 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.

Amazon CloudWatch Security

Amazon CloudWatch is a web service that provides monitoring for AWS cloud resources, starting with Amazon EC2. It provides customers with visibility into resource utilization, operational performance, and overall demand patterns including metrics such as CPU utilization, disk reads and writes, and network traffic. You can set up CloudWatch alarms to notify you if certain thresholds are crossed, or to take other automated actions such as adding or removing EC2 instances if Auto-Scaling is enabled.

CloudWatch captures and summarizes utilization metrics natively for AWS resources, but you can also have other logs sent to CloudWatch to monitor. You can route your guest OS, application, and custom log files for the software installed on your EC2 instances to CloudWatch, where they will be stored in durable fashion for as long as you'd like. You can configure CloudWatch to monitor the incoming log entries for any desired symbols or messages and to surface the results as CloudWatch metrics. You could, for example, monitor your web server's log files for 404 errors to detect bad inbound links or invalid user messages to detect unauthorized login attempts to your guest OS.

Like all AWS Services, Amazon CloudWatch requires that every request made to its control API be authenticated so only authenticated users can access and manage CloudWatch. Requests are signed with an HMAC-SHA1 signature calculated from the request and the user's private key. Additionally, the Amazon CloudWatch control API is only accessible via SSL- encrypted endpoints.

You can further control access to Amazon CloudWatch by creating users under your AWS Account using AWS IAM, and controlling what CloudWatch operations these users have permission to call.

AWS CloudHSM Security

The AWS CloudHSM service provides customers with dedicated access to a hardware security module (HSM) appliance designed to provide secure cryptographic key storage and operations within an intrusion-resistant, tamper-evident device. You can generate, store, and manage the cryptographic keys used for data encryption so that they are accessible only by you. AWS CloudHSM appliances are designed to securely store and process cryptographic key material for a wide variety of uses such as database encryption, Digital Rights Management (DRM), Public Key Infrastructure (PKI), authentication and authorization, document signing, and transaction processing. They support some of the strongest cryptographic algorithms available, including AES, RSA, and ECC, and many others.

The AWS CloudHSM service is designed to be used with Amazon EC2 and VPC, providing the appliance with its own private IP within a private subnet. You can connect to CloudHSM appliances from your EC2 servers through SSL/TLS, which uses two-way digital certificate authentication and 256-bit SSL encryption to provide a secure communication channel.

Selecting CloudHSM service in the same region as your EC2 instance decreases network latency, which can improve your application performance. You can configure a client on your EC2 instance that allows your applications to use the APIs provided by the HSM, including PKCS#11, MS CAPI and Java JCA/JCE (Java Cryptography Architecture/Java Cryptography Extensions).

Before you begin using an HSM, you must set up at least one partition on the appliance. A cryptographic partition is a logical and physical security boundary that restricts access to your keys, so only you control your keys and the operations performed by the HSM. AWS has administrative credentials to the appliance, but these credentials can only be used to manage the appliance, not the HSM partitions on the appliance. AWS uses these credentials to monitor and maintain the health and availability of the appliance. AWS cannot extract your keys nor can AWS cause the appliance to perform any cryptographic operation using your keys.

The HSM appliance has both physical and logical tamper detection and response mechanisms that erase the cryptographic key material and generate event logs if tampering is detected. The HSM is designed to detect tampering if the physical barrier of the HSM appliance is breached. In addition, after three unsuccessful attempts to access an HSM partition with HSM Admin credentials, the HSM appliance erases its HSM partitions.

When your CloudHSM subscription ends and you have confirmed that the contents of the HSM are no longer needed, you must delete each partition and its contents as well as any logs. As part of the decommissioning process, AWS zeroizes the appliance, permanently erasing all key material.

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

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 backend infrastructure.

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. Once the end user is authenticated with the provider, an OAuth or OpenID Connect token returned from the provider is passed by your application to Cognito, which returns a new 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 (e.g., mobile user, 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 life-span (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 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 Cognito Sync service and Amazon Mobile Analytics. If your application needs access to other AWS resources such as Amazon S3 or DynamoDB, you can modify your roles directly from the IAM management console.

With Amazon Cognito, there's no need to create individual AWS accounts or even IAM accounts for every one of your web/mobile app's 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. However, if they choose to do this later, Cognito will merge data and identification information.

Because Amazon Cognito stores data locally as well as 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 anything they put into the dataset, they can immediately retrieve whether they are online or not. 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. 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 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. Once Cognito receives the token, it returns a new Cognito ID for the user and a set of temporary, limited-privilege AWS credentials.

Each 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—on iOS for example, the 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.

Amazon Mobile Analytics

Amazon Mobile Analytics is a service for collecting, visualizing, and understanding mobile application usage data. It enables you to track customer behaviors, aggregate metrics, and identify meaningful patterns in your mobile applications. Amazon Mobile Analytics automatically calculates and updates usage metrics as the data is received from client devices running your app and displays the data in the console.

You can integrate Amazon Mobile Analytics with your application without requiring users of your app to be authenticated with an identity provider (like Google, Facebook, or Amazon). For these unauthenticated users, Mobile Analytics works with Amazon Cognito to provide temporary, limited-privilege credentials. To do this, you first create an identity pool in Cognito. The identity pool will use IAM roles, which is a set of permissions not tied to a specific IAM user or group but which allows an entity to access specific AWS resources. The entity assumes a role and receives temporary security credentials for authenticating to the AWS resources defined in the role. By default, Amazon Cognito creates a new role with limited permissions – end users only have access to the Cognito Sync service and Amazon Mobile Analytics. If your application needs access to other AWS resources such as Amazon S3 or DynamoDB, you can modify your roles directly from the IAM management console.

You can integrate the AWS Mobile SDK for Android or iOS into your application or use the Amazon Mobile Analytics REST API to send events from any connected device or service and visualize data in the reports. The Amazon Mobile Analytics API is only accessible via an SSL-encrypted endpoint.

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

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 would like to launch. Once the WorkSpaces are ready, users receive an email informing them where they can download the relevant client and log into 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 '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 multi-factor authentication upon sign in in the form of a hardware or software token. Amazon WorkSpaces supports MFA using an on-premise 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 EC2 instance within a VPC. You can create WorkSpaces in a VPC you already own or have the WorkSpaces service create one for you automatically using the WorkSpaces Quick Start option. When you use the Quick Start option, WorkSpaces not only creates the VPC, but it performs several other provisioning and configuration tasks for you, such as creating an Internet Gateway for the VPC, setting up a directory within the 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 WorkSpace instances. Or the 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 the WorkSpaces that belong to your Directory. This allows you to control network access from Amazon WorkSpaces in your VPC to other resources in your Amazon VPC and on-premises network.

Persistent storage for WorkSpaces is provided by Amazon EBS and is automatically backed up twice a day to Amazon S3. If 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 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's 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 WorkSpaces, and configured to install updates on a weekly basis. You can use an alternative patching approach or to 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 more easily apply Active Directory changes for all your WorkSpaces users.

Amazon WorkDocs

Amazon WorkDocs is a managed enterprise storage and sharing service with feedback capabilities for user collaboration. Users can store any type of file in a WorkDocs folder and allow others to view and download them. Commenting and annotation capabilities work on certain file types such as MS Word, and without requiring the application that was used to originally create the file. WorkDocs notifies contributors about review activities and deadlines via email and performs versioning of files that you have synced using the WorkDocs Sync application.

User information is stored in an Active Directory-compatible network directory. You can either create a new directory in the cloud, or connect Amazon WorkDocs to your on-premises directory. When you create a cloud directory using WorkDocs' quick start setup, it also creates a directory administrator account with the administrator email as the username. An email is sent to your administrator with instructions to complete registration. The administrator then uses this account to manage your directory.

When you create a cloud directory using WorkDocs' quick start setup, it also creates and configures a VPC for use with the directory. If you need more control over the directory configuration, you can choose the standard setup, which allows you to specify your own directory domain name, as well as one of your existing VPCs to use with the directory. If you want to use one of your existing VPCs, the VPC must have an Internet gateway and at least two subnets. Each of the subnets must be in a different Availability Zone.

Using the Amazon WorkDocs Management Console, administrators can view audit logs to track file and user activity by time, IP address, and device, and choose whether to allow users to share files with others outside their organization.
Users can then control who can access individual files and disable downloads of files they share.

All data in transit is encrypted using industry-standard SSL. The WorkDocs web and mobile applications and desktop sync clients transmit files directly to Amazon WorkDocs using SSL. WorkDocs users can also utilize Multi-Factor Authentication, or MFA, if their organization has deployed a Radius server. MFA uses the following factors: username, password, and methods supported by the Radius server. The protocols supported are PAP, CHAP, MS-CHAPv1, and MS-CHAPv2

You choose the AWS Region where each WorkDocs site's files are stored. Amazon WorkDocs is currently available in the US-East (Virginia), US-West (Oregon), and EU (Ireland) AWS Regions. All files, comments, and annotations stored in WorkDocs are automatically encrypted with AES-256 encryption.

Further Reading

https://aws.amazon.com/security/security-resources/

Introduction to AWS Security Processes

Overview of AWS Security - Storage Services
Overview of AWS Security - Database Services
Overview of AWS Security - Compute Services
Overview of AWS Security - Application Services
Overview of AWS Security - Analytics, Mobile and Application Services
Overview of AWS Security – Network Services