Design Principles

**Implement a strong identity foundation:**

* Lease privileges
* Separation of duties
* No long term credentials

**Enable traceability:**

* Monitor, alert, and audit actions and changes to your environment in real time
* Integrate logs and metrics with systems to automatically respond and take action

**Apply security at all layers:**

* Apply to all layers (e.g., edge network, VPC, subnet, load balancer, every instance, operating system, and application).

**Automate security best practices**

* Software based security mechanism
* Controls defined/managed as code

**Protect data in transit and at rest**

**Prepare for security events:**

Incident managent process – automation to act - detection, investigation, and recovery.

**Definition**

Identity and access management

Detective controls

Infrastructure protection

Data protection

Incident response

Identity and access management

Protecting AWS Credentials

Root Account – Enable MFA and delete root keys

Use this account for only creating initial set of users and groups

Federation – integration with existing on-premises identity management (AD). No need to define IAM users, existing password can be used

IAM users – apply strong password policy + rotation policy + MFA

* + Keys for access to SDKS and command line
  + IAM roles to grant permissions
  + Access keys – don’t store inn  source code repo

IAM instance profiles – service to service authentication – EC2 + AWS STS (security token service) to generate and manage temporary credentials used in software that must authenticate to AWS APIs

Key service – IAM

Others important:

**Security token service -** lets you request temporary, limited-privilege credentials for authentication with other AWS APIs

**IAM instance profiles for EC2 instances** allow you to leverage the Amazon EC2 metadata service and managed, temporary credentials for accessing other AWS APIs.

Fine-Grained Authorization

least privilege ensures that authenticated identities are only permitted to perform the most minimal set of functions necessary to fulfill a specific task, while balancing usability and efficiency

implemented in AWS **using IAM roles and policies**.

Key service – IAM

Others important:

**AWS Organizations** lets you centrally manage accounts and enforce policies for multiple AWS accounts.

Detective Controls

to identify a potential security threat or incident

set the correct automated alerting notifications based on defined conditions

can help your organization identify and understand the scope of anomalous activity

Capture and Analyze Logs

collecting and aggregating logs is the ability to extract meaningful insight from the great volumes of log and event data generated by modern, complex architectures

Key service

**AWS Config**

**Amazon Elasticsearch Service -**

* manages and scales a cluster of the popular open-source search and analytics engine Elasticsearch
* use this solution to index, search, and render security data

**Amazon CloudWatch Logs**

**Amazon EMR**

* allows you to write applications to parse and analyze logs at scale,
* No need to manage Hadoop

**Amazon Simple Storage Service (S3) and Amazon Glacier**

* Store logs

**Amazon Athena**

* used to analyze logs (ex : CloudTrail logs)
* help you identify trends and further isolate activity by attribute, such as source IP address or user.

Integrate Auditing Controls with Notification and Workflow

Security events + Infra changes  ---

Infra changes  AWS config

Cloud watch events  route them to Lambda function or SNS

integrate the flow of security events and findings into a notification and workflow system such as a ticketing system, a bug/issue system, or other security information and event management (SIEM) system

CI/CD  test security issues before deploying to prod

Amazon inspector  perform configuration assessments for known common vulnerabilities and exposures (CVEs), assess your instances against security benchmarks, and fully automate the notification of defects

Key Service : CloudWatch Events

Others:

**AWS Config Rules**

**Amazon CloudWatch**

**Amazon CloudWatch API and AWS SDKs**

* Used to create custom events  inject into CloudWatch Events for rule-based processing and routing

**Amazon Inspector**

offers a programmatic way to find security defects or misconfigurations in your operating systems and applications.

Integration with CICD

Infrastructure Protection

**Protecting network and host-level boundaries**

Decide which components need to be public/customer facing LB

Key Service : VPC

Others:

* **SG**
* **Direct connect** – direct connectivity from ur DC to VPC

**System security configuration and maintenance**

Key Service : Security Gropus per-instance firewall

Others:

**Amazon Inspector –** Scans Vulnerabilities & Identify deviations from best practices in your guest operating systems and applications

**EC2 Systems Manager Run Command -**  provides a simple way of automating common administrative tasks such as remotely executing shell scripts or PowerShell commands with granular access control and visibility, installing software updates, or making changes to the configuration of the operating system

**EC2 Systems Manager State Manager –** Firewall setting and antivirus

**EC2 Systems Manager Inventory –** Configuration and software installed

**EC2 Systems Manager Parameter Store -**  provides a centralized store to manage your configuration data

**EC2 Systems Manager Patch –** deploy patches

**Enforcing service-level protection**

ensure that users and automated systems have exactly the level of access needed to perform their tasks (least privilege).

Key Service : IAM

Others:

**AWS KMS** - allows you to set policies on the individual key

**Amazon S3 -**  allows you to set bucket policies for each S3 bucket

Data Protection

**Data classification**

Key Service :Resource tagging

Others:

**Amazon Macie** uses machine learning to automatically discover, classify, and protect sensitive data in AWS. **(Discover + Classify + Protect)**

**AWS KMS** allows you to define encryption keys and access policies to them.

**Encryption/tokenization**

*Tokenization* is a process that allows you to define a token to represent an otherwise sensitive piece of information (Example credit card number)

*Encryption* is a way of transforming content in a manner that makes it unreadable without a secret key necessary to decrypt the content back into plain text.

Both tokenization and encryption can be used to secure and protect content as appropriate

Key Service :KMS

Others:

**AWS CloudHSM-** provides a hardware security module for managing your keys

**Amazon DynamoDB –** can be used to store encrypted content of token

**Protecting data at rest**

Key Service :KMS

Others:

**Amazon S3**

**Amazon EBS**

**Amazon Glacier**

**Protecting data in transit**

When protecting your data in transit, selecting protocols that implement the latest version of Transport Layer Security (TLS) is a common best practice

AWS Certificate Manager (ACM) - provides you the ability to manage and deploy certificates for your domains

Key Service : AWS Certificate Manager (ACM)

Others:

**ELB Classic Load Balancers and Application Load Balancers** help deploy and manage load balancers using secure endpoints.

**Amazon CloudFront** supports encrypted endpoints for your content distributions.

**AWS Shield** is a managed Distributed Denial of Service (DDoS) protection service that safeguards web applications running on AWS

**Data backup/replication/recovery**

Key Service : S3 - key AWS service that supports data backup, replication, and recovery

Others:

**Amazon S3 Cross-Region Replication** is an Amazon S3 bucket-level feature that enables automatic, asynchronous copying of objects across buckets in different AWS Regions.

**Amazon S3 lifecycle polices and versioning** allow you to implement a backup strategy and meet retention requirements.

**Amazon EBS snapshot operations** let you back up your volumes attached to EC2 instances.

Incidence Response

your organization should still implement a response plan and a plan to mitigate security incidents.

Clean Room

By using tags to properly describe your AWS resources, incident responders can quickly determine the potential impact of an incident. For example, tagging instances and other assets with an owner or work queue in a ticketing system allows the team to engage the right people more quickly. By tagging systems with a data classification or a criticality attribute, the impact of an incident can be estimated more accurately.

Key Service

**IAM** should be used to grant appropriate authorization to incident response teams.

**AWS CloudFormation** can be used to create a trusted environment for conducting deeper investigations.

**Amazon EC2 APIs** can be used to help isolate instances and mitigate the impact of a security incident.  Example Remove a VM from LB..

**AWS Step Functions** can be used to coordinate a sequence of steps to automate incident response.

Thanks & Regards,

Shantaram Vernekar