# AWS Security Checklist

*This checklist provides customer recommendations that align with the Well-Architected Framework Security Pillar.*

## Identity & Access Management

1. **Secure your AWS account.**
   Use AWS Organizations to manage your accounts, use the root user by exception with multi-factor authentication (MFA) enabled, and configure account contacts.
2. **Rely on centralized identity provider.**
   Centralize identities using either AWS Single Sign-On or a third-party provider to avoid routinely creating IAM users or using long-term access keys—this approach makes it easier to manage multiple AWS accounts and federated applications.
3. **Use multiple AWS accounts to separate workloads and workload stages such as production and non-production.**
   Multiple AWS accounts allow you to separate data and resources, and enable the use of Service Control Policies to implement guardrails. AWS Control Tower can help you easily set up and govern a multi-account AWS environment.
4. **Store and use secrets securely.**
   Where you cannot use temporary credentials, like tokens from AWS Security Token Service, store your secrets like database passwords using AWS Secrets Manager which handles encryption, rotation, and access control..

## Detection

1. **Enable foundational services: AWS CloudTrail, Amazon GuardDuty, and AWS Security Hub.**
   For all your AWS accounts configure CloudTrail to log API activity, use GuardDuty for continuous monitoring, and use AWS Security Hub for a comprehensive view of your security posture..
2. **Configure service and application level logging.**
   In addition to your application logs, enable logging at the service level, such as Amazon VPC Flow Logs and Amazon S3, CloudTrail, and Elastic Load Balancer access logging, to gain visibility into events. Configure logs to flow to a central account, and protect them from manipulation or deletion.
3. **Configure monitoring and alerts, and investigate events.**
   Enable AWS Config to track the history of resources, and Config Managed Rules to automatically alert or remediate on undesired changes. For all your sources of logs and events, from AWS CloudTrail, to Amazon GuardDuty and your application logs, configure alerts for high priority events and investigate.

## Infrastructure Protection

1. **Patch your operating system, applications, and code.**
   Use AWS Systems Manager Patch Manager to automate the patching process of all systems and code for which you are responsible, including your OS, applications, and code dependencies.

2. **Implement distributed denial-of-service (DDoS) protection for your internet facing resources.**
   Use Amazon Cloudfront, AWS WAF and AWS Shield to provide layer 7 and layer 3/layer 4 DDoS protection.
3. **Control access using VPC Security Groups and subnet layers.**
   Use security groups for controlling inbound and outbound traffic, and automatically apply rules for both security groups and WAFs using AWS Firewall Manager. Group different resources into different subnets to create routing layers, for example database resources do not need a route to the internet.

## Data Protection

1. **Protect data at rest.**
   Use AWS Key Management Service (KMS) to protect data at rest across a wide range of AWS services and your applications. Enable default encryption for Amazon EBS volumes, and Amazon S3 buckets.
2. **Encrypt data in transit.**
   Enable encryption for all network traffic, including Transport Layer Security (TLS) for web based network infrastructure you control using AWS Certificate Manager to manage and provision certificates.
3. **Use mechanisms to keep people away from data.**
   Keep all users away from directly accessing sensitive data and systems. For example, provide an Amazon QuickSight dashboard to business users instead of direct access to a database, and perform actions at a distance using AWS Systems Manager automation documents and Run Command.

## Incident Response

1. **Ensure you have an incident response (IR) plan.**
   Begin your IR plan by building runbooks to respond to unexpected events in your workload. For details, see the AWS Security Incident Response Guide.
2. **Make sure that someone is notified to take action on critical findings.**
   Begin with GuardDuty findings. Turn on GuardDuty and ensure that someone with the ability to take action receives the notifications. Automatically creating trouble tickets is the best way to ensure that GuardDuty findings are integrated with your operational processes.
3. **Practice responding to events.**
   Simulate and practice incident response by running regular game days, incorporating the lessons learned into your incident management plans, and continuously improving them.

*For more best practices, see the Security Pillar of the Well-Architected Framework and Security Documentation.*

**Notices**

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. 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.

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.