AWS Whitepapers

Classic Intrusion Analysis Frameworks for AWS Environments: Application and Enhancement

Classic Intrusion Analysis Frameworks for AWS Environments: Application and Enhancement: AWS Whitepapers

Table of Contents

Classic Intrusion Analysis Frameworks for AWS Environments: Application and Enhancement

Publication date: **March 31, 2021** (*Document history*)

Classic Intrusion Analysis Frameworks for AWS Environments: Application and Enhancement

Abstract

Today, many Chief Information Security Officers (CISOs) and cybersecurity practitioners are looking for an effective cybersecurity methodology that will help them achieve measurably better security for their organization. One approach that has helped some organizations is to use classic intrusion analysis frameworks to analyze cybersecurity risks and provide methodologies and technologies for responding to attacks.

This paper provides background context on classic intrusion analysis frameworks, and shows how the transition to the cloud undermines some of its key premises, naturally disrupting modern attacker intrusion methods, i.e. “breaking intrusion kill chains”. This paper outlines how to use both the classic intrusion analysis framework and the AWS Cloud to address external threats to your AWS environment’s security.

Introduction

Cybersecurity threats continue to challenge organizations around the world. Many of the cybersecurity strategies that organizations have employed over the past two decades have failed to stop network compromises and data breaches. This has led many Chief Information Security Officers (CISOs) and cybersecurity practitioners to look for more effective approaches to manage cybersecurity risk for their organizations.

One such approach, pioneered by Lockheed Martin Corporation, is described in Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains.

Intelligence-driven computer network defense is a necessity in light of advanced persistent threats. As conventional, vulnerability-focused processes are insufficient, understanding the threat itself, its intent, capability, doctrine, and patterns of operation is required to establish resilience. The intrusion kill chain provides a structure to analyze intrusions, extract indicators and drive defensive courses of actions. Furthermore, this model prioritizes investment for capability gaps, and serves as a framework to measure the effectiveness of the defenders' actions.

> **Note**
>
> Hutchins EM, Cloppert MJ, Amin RM. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. (Lockheed Martin, Bethesda, MD, 2011), 12.

Since Lockheed Martin's paper was published in 2011, many variations of this particular intrusion analysis approach have been developed in the cybersecurity industry, and many organizations have benefited from implementing this classic framework for intrusion analysis to guide mitigation and response strategy for their on-premises infrastructure.

This whitepaper offers an assessment of the classic intrusion analysis framework from an AWS Cloud perspective; pointing out where it applies, where it may not, and describing AWS mechanisms to support and enhance any customers' intrusion analysis approach.

What is an intrusion method?

Observation and analysis suggests that in traditional IT environments attackers often employ a repeatable process that helps them identify their target and potential weaknesses in their target's security posture, as well as ways to exploit these weaknesses. Once the victim's weaknesses are successfully exploited, attackers use illicit access to their victim's infrastructure for a range of nefarious purposes, including data theft, compromising data integrity, destroying data and/or infrastructure, disrupting operations, and perpetrating attacks on other victims. The process that attackers use to conduct an attack can be analyzed as an *intrusion kill chain*. The terminology comes from military usage, where the series of steps required to attack a target is called a "chain," and a successful attack is called a "kill." While the terminology is rather military oriented for cybersecurity purposes, the concepts are well-established and many remain reasonably useful. For the purposes of this paper, we use "intrusion methods" interchangeably with "kill chains" when not specifically discussing a particular classic intrusion analysis framework.

Attackers perform different tasks in each phase of their intrusion, as they plan and execute their intrusion attempts. In the mapping section, this paper modifies previous models slightly, so that organizations can leverage the inherent benefits of the cloud and the mitigations provided by AWS to stop intrusions.

Phases of Lockheed Martin Framework

The Lockheed Martin "kill chain" framework breaks down attacker behavior into the following phases:

Phase 1: Reconnaissance

This phase represents the work attackers do to research and select their targets, and understand their targets' digital footprints. These activities include port scans and vulnerability scans of publicly accessible systems of the targets and their supply chain partners.

Phase 2: Weaponization (Exploit Development)

In this phase, attackers plan and acquire the tools they'll use to try to exploit the weaknesses they believe the victim has. For example, they may build a malformed PDF file that is specially crafted to exploit a vulnerability in a PDF parser they know their intended victim uses. Attackers may

also develop malware to steal system login credentials from the victim. (Note that this document hereafter refers to this phase as *exploit development* rather than weaponization.)

Phase 3: Delivery

In this phase, the attackers transmit their *weapon* to the intended victim. Examples of delivery mechanisms include phishing emails, malicious email attachments, drive-by download sites, and so on.

Phase 4: Exploitation

After being delivered to the target, the *weapon* seeks to exploit the weakness it was designed for. This weakness may be a vulnerability or misconfiguration in an operating system, web browser, or other application. An exploit can also be designed to trick people into making poor trust decisions —also known as *social engineering*. Another weakness that attackers typically try to exploit is weak, leaked, or stolen passwords or other types of credentials.

Phase 5: Installation

After vulnerabilities have been successfully exploited, many attackers attempt to persist undetected in the environment as long as possible to accomplish their objectives. In this phase, attackers attempt to install tools that allow them to maintain running code and remote access to that code inside the victim's environment.

Phase 6: Command and Control

Attackers maintain illicit access to their victims' environments and potentially remotely control compromised infrastructure.

Phase 7: Actions on Objectives

At this point in the intrusion (or earlier, depending on the chosen methods), the attackers are in a position to achieve their objectives. Objectives can include data theft, compromising data integrity, destroying data and/or infrastructure, disrupting operations, and perpetrating attacks on other victims.

Weakness and limitations of classic frameworks

Some of the ideas behind the original approach, such as its focus on network perimeter concepts as well as *advanced persistent threats*, are not fully applicable in a modern, cloud-based environment. Also, the original approach tends to assume a relatively static, traditional IT environment, which is not what customers typically build in the cloud. Advanced persistent threats (APTs) have no place to live in DevSecOps environment, where entirely new, fully scanned, and tested copies of applications are deployed on new virtual infrastructure on a daily if not almost hourly basis; "network intrusion" has no definite meaning in a modern perimeter-less network of cooperating microservices, where all hosts are network-reachable without being vulnerable to persistent takeover. Even where network perimeters remain an important defense, the cloud makes it much easier to create and maintain segmentation or micro-perimeters such that lateral movement is far more difficult, and potential attackers far more constrained.

As a result, classic intrusion analysis frameworks suffer from a number of weaknesses and limitations when applied to modern cloud platforms. These frameworks are particularly problematic when analyzing systems developed with modern DevSecOps practices using cloud-native architectural patterns. But these framework weaknesses also apply to some extent even for more traditional applications that have been "lifted and shifted" to the cloud.

Perimeter-less or zero-trust network architectures

Perimeter-less or zero-trust network architectures render large parts of the classic intrusion analysis framework flawed or pointless. Zero-trust network architectures mitigate multi-step attacks designed to move through different phases or components. These architectures force every request or operation, regardless of their origin, to prove their trustworthiness, authenticity, and integrity. Privileges in these environments are none by default, and are never granted permanently. Subsequently, unauthorized actions are tightly controlled and the entire architecture is less vulnerable to abuse, from both insider and external threat actors. For more information, see How to think about Zero Trust architectures on AWS.

Stronger segmentation of networks in the cloud

Cloud offers stronger segmentation of networks than many on-premises IT environments. Every service or application can operate within its own virtual private cloud (VPC). Within a VPC, there are firewall and routing rules that can't be modified by an attacker, even if they gain access to a

service or instance, because the corresponding rules are managed and owned by different parts of the organization and control over those settings requires access to IAM credentials or roles that are not present in the environment to which the attacker has gained access.

Microservices architectures

In microservices architectures, data extraction and/or lateral movement is highly constrained. You can only see the data inside that microservice and, in terms of lateral movement, you can only call or respond to well-defined API calls or interfaces. Even if a microservice is indeed compromised, the attack surface and privileges obtained are limited to that microservice and the privileges it has to call other microservices.

Static environments versus dynamic environments

Cloud enables the use of dynamic environments, such as environments where deployments or installations can only occur through a well-defined continuous integration/continuous deployment (CI/CD) pipeline that enforces and verifies security controls. Dynamic environments are refreshed frequently, meaning that their configuration and services are redeployed based on an agreed secure state and can then be made immutable once deployed – such that no privileges exist to change them once deployed. Their immutability makes it extremely hard for an attacker to gain persistence, even between the time of attack and next deployment. Moreover, in such environments, changes or attempts to implement changes, are easily discoverable and, regardless, overwritten by the next healthy and secure state refresh. Newly deployed infrastructure and application code have been built and tested with the latest security patches as a normal part of the CI/CD process. All these qualities of dynamic, frequently changed and improved infrastructure and application code make the concepts of installation, persistence, and exploitation much less relevant, or even potentially meaningless, in the cloud. An environment which is frequently redeployed is an environment where even a novel Advanced Persistent Threat (APT) payload won't be able to persist in an active state beyond the next refresh, requiring an attacker to have to re-compromise the environment after the redeployment, with the flaw that led to the original compromise very likely to have been caught and patched.

Traditional IT environments versus the cloud

The value of a classical intrusion analysis framework does not directly apply to cloud environments when those environments are architected and operated according to current recommendations and

modern approaches. However, across an entire organization and the pool of applications in use, such classical frameworks remain viable. Among other reasons, it will take time for many systems or applications to be modernized to fit the more cloud-based patterns discussed in previous sections. These *lift and shift* kinds of systems, as well as systems on the journey to full immutable infrastructure modernization, still have characteristics of traditional systems, and so application of classic frameworks is still valuable. In addition, even a modern CI/CD system may be deployed with a relatively static virtual network environment, for example, and so some of these concepts can still be applied to "container" into which applications are deployed. So, let's turn to the traditional analysis and its application in light of cloud technologies. Cloud native services, alone or used in conjunction with third-party solutions, can help mitigate intrusion methods for more traditional systems and applications as well.

Stopping intrusion methods

Understanding the stages or phases that attackers use to execute attacks can help security teams formulate ways to prevent successful attacks. Lockheed Martin's intrusion kill chain framework describes a *courses of action matrix* (shown in the following table) that helps plan courses of action against each phase of any expected intrusion method. These actions include: detect, deny, disrupt, degrade, deceive, and destroy. (For definitions, see *Table 3* in Modifications for the cloud.)

Table 1: Courses of action matrix

Phase	Detect	Deny	Disrupt	Degrade	Deceive	Destroy
Reconnaissance						
Weaponization						
Delivery						
Exploitation						
Installation						
Command and Control						
Actions on Objectives						

The concept of the matrix is for defenders to build capabilities that detect, deny, disrupt, degrade, deceive, and destroy attackers' efforts in each phase of the intrusion. The goal is to stop the intrusion as early in the attack as possible because this reduces the recovery time, effort, cost, and damage associated with each attack. For example, detecting an attack in the *reconnaissance* phase is preferable to degrading the attack in the Command and Control (C2) phase.

Topics

Modifications for the cloud

The AWS approach extends the courses of action available to defenders so that organizations can leverage the inherent benefits of the cloud and the mitigations provided by AWS.

The courses of action have been modified in the following ways:

- Destroy is replaced with *respond*.

  Security teams need to plan for successful intrusions to be prepared to respond effectively and efficiently, should an intrusion ever occur. AWS provides customers with response capabilities that help reduce the time, effort, damage, and costs associated with intrusion attempts.

- *Contain* is added to the courses of action.

  Containing attackers that successfully penetrate an IT environment is a prudent course of action. Preventing lateral movement and the credential theft and reuse attacks typically associated with lateral movement, will reduce potential damage and speed recovery. AWS provides several controls that customers can use to help contain attackers. For more information, see *Appendix: Reference Material*.

- *Restore* is added to the matrix.

  In the scenario where every course of action that an organization has implemented fails across every phase of the intrusion method, restoring data and infrastructure as quickly as possible will be important to restoring business as usual. The AWS Cloud provides a number of powerful service features that allow the rapid restoration of data, such as point-in-time restore for databases, and object locking (immutability) and versioning for Amazon S3.

AWS also supports many conventional disaster recovery (DR) architectures from *pilot light* environments that may be suitable for small customer workload data center failures to *hot standby* environments that enable rapid failover at scale. With data centers in AAWS Regions all around the world, AWS provides a set of cloud-based disaster recovery services designed to provide rapid recovery of your IT infrastructure and data, such as CloudEndure Disaster Recovery.

Table 2: Modified courses of action matrix

Phase	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Reconnaissance: pre-intrusion								
Reconnaissance: post-intrusion								
Exploit development								
Delivery								
Exploitation								
Installation								
Command and Control								
Actions on Objectives								

The following table lists definitions of the courses of action included in the matrix. All of the following definitions are based on definitions published in Characterizing Effects on the Cyber Adversary, A Vocabulary for Analysis and Assessment.

Note

Defined in 2006 version of JP 3-13, as documented in Mitre, "Characterizing Effects on the Cyber Adversary, A Vocabulary for Analysis and Assessment", https://www.mitre.org/sites/default/files/publications/characterizing-effects-cyber-adversary-13-4173.pdf

Table 3: Courses of action definitions

Action	Definition
Detect	To discover or discern the existence, presence, or fact of an intrusion into information systems.
Deny	To prevent the adversary from accessing and using critical information, systems, and services.
Disrupt	To break or interrupt the flow of information.
Degrade	To reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means.
Deceive	To cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality.
Contain	The action of keeping something harmful under control or within limits.

Action	Definition
Respond	To react quickly to an adversary's or another's IO attack or intrusion.
Restore	To bring information and information systems back to their original state.

Sample mapping exercise

AWS services and AWS Partner Network (APN) services can be mapped to this courses of action *matrix*(Hutchins, Op.Cit., Page. 5). Performing this mapping exercise can help identify which services, features, and functionality can help organizations detect, deny, disrupt, deceive, contain attacks, as well as respond to them and help recover from them when necessary.

Using the *installation* phase as an example, you can use the following AWS services and third-party services to help you to detect, deny, disrupt, deceive, contain, respond, and recover during this phase of the intrusion.

Detect

To help detect attacker activity, use log data from Amazon CloudWatch, CloudWatch Logs, AWS CloudTrail, and VPC Flow Logs, and use reporting tools such as Amazon OpenSearch Service, Amazon QuickSight and third-party tools. Key examples are provided in the following table.

Table 4: Controls for Detect

Control Name	Description
Amazon GuardDuty	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.
Amazon Detective	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspiciou

Control Name	Description
	s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.
Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third-Party Tools	These controls monitor, detect, visualize, and receive notifications of attacks, and respond to changes in your AWS resources
AWS Security Hub	This control gives you a comprehensive view of your high priority security alerts and compliance status across AWS accounts.
AWS Security Hub Partners	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
AWS Systems Manager State Manager, AWS Systems Manager Inventory, AWS Config	When new AWS assets are created, or if malware is installed with a regular package, the AWS System Manager Inventory identifies it and sends it to AWS Config for evaluation.
Third-Party Security Tools for Containers	This control implements advanced security protection and behavioral security solutions for containers.
Third-Party Security Tools for AWS Lambda Functions	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Network Offerings – Anti-Malw are Protection	These controls help to detect and block malicious payloads.

Deny

There are numerous process and technology controls that can help prevent an adversary from accessing and using critical information, systems, and services. A DevSecOps culture and processes that enforce CI/CD static and dynamic security analysis, and immutable, short-lived infrastructures all help mitigate attackers' efforts in the installation phase of an attack. The following table contains numerous other control examples.

Table 5: Controls for Deny

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries	These controls can be configured to provide strong, least-privilege and need-to know security principles for both the users and services that can access your resources. IAM privileges are required to either grant or deny privileges for AWS administrators and engineers.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts	These controls can be utilized to provide strong, least-privilege and need-to know security principles for both users and services across a multi-account structure. SCPs can be used to deny (but not grant) privileges, overriding the potential privileges granted via IAM. You can control the privileges of all principals in child accounts and organizational units.
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies	These controls manage access to objects and can prevent upload of objects into the Amazon S3 bucket by malicious actors. Note that there are other AWS services, besides S3, that also have resource policies, such as Amazon Simple Queue Service and Amazon Simple Notification Service.

Control Names	Descriptions
Amazon Cognito	This control provides temporary, limited-p rivilege end-user credentials to allow access to appropriate AWS services.
Bottlerocket	This control provides a minimized OS environment capable of running and managing containers, which provides no extraneous listeners or services.
Amazon EC2 – Linux, Security-Enhanced Linux (SELinux) – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux, FreeBSD – Hardening and Minimization	These controls disable or remove unused services and packages.
Amazon EC2 – Windows – User Account Control (UAC)	UACs make it more difficult for malware to install and run.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC)	This control implements least-privilege account profiles.
Amazon EC2 – Windows –Device Guard	This control specifies which binaries are authorized to run on your server.
AWS Partner Network Offerings – Anti-Malw are Protection	These controls help to detect and block malicious payloads.

Disrupt

AWS Config and AWS Systems Manager State Manager can monitor the configuration of cloud services as well as Amazon EC2 instances, specify a configuration policy for the instances, and automatically apply updates or configuration changes. This approach can help disrupt post-exploitation changes to systems. The following table contains more examples.

Table 6: Controls for Disrupt

Control Names	Descriptions
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies	These controls manage access to objects and prevent upload of malicious objects into the Amazon S3 bucket.
AWS Systems Manager State Manager	This control helps you to define and maintain consistent OS configurations.
Amazon EC2 – Linux, SELinux – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Windows – User Account Control (UAC)	UACs make it more difficult for malware to install and run.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC)	This control implements least-privilege account profiles.
Amazon EC2 – Windows – Device Guard	This control specifies which binaries are authorized to run on your server.
AWS Partner Network Offerings – File Integrity Monitoring	This control helps to maintain the integrity of operating system and application files.

Control Names	Descriptions
AWS Partner Network Offerings – Anti-Malw are Protection	These controls help to detect and block malicious payloads.

Degrade

The objective of the Degrade control in the Installation phase is to negatively impact the attacker's control, communications, and data collection activities. Some examples of controls that help defenders do this are listed in the following table.

Table 7: Controls for Degrade

Control Names	Descriptions
AWS Systems Manager State Manager	This control helps you to define and maintain consistent OS configurations.
Immutable Infrastructure – Short-Lived Environments	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Deceive

Third-party deception technologies present themselves as part of legitimate infrastructure so that when an attacker attempts to compromise it, they help detect, contain, and recover faster. Integrating honeypots, honeynets, fake credentials, and fake documents into a legitimate IT environment enables organizations to detect attempted unauthorized access to resources and respond and recover rapidly. Examples of third-party companies that offer deception technologies include GuardiCore, PacketViper, and others.

Table 8: Controls for Deceive

Control Names	Descriptions
Honeypot and Honeynet Environments	These controls help to degrade, detect, and contain attacks.

Control Names	Descriptions
Honeywords and Honeykeys	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.

Contain

As previously noted, the segmented nature of typical cloud networks provides far more containment than traditional datacenter-wide layer 2 networks. Service-oriented architectures and microservices are also containment-oriented, as the compromise of one component can have a limited effect on a large application or system. The code running inside these services should be least-privileged in terms of the power to call cloud APIs, and then only with the temporary, fast-rotating STS credentials of an EC2 role, with VPC Endpoint Policies used to make it impossible to use even those short-lived credentials outside the cloud. All of these best practices make it difficult for a successful compromise of code inside a service to alter critical cloud configuration settings. Beyond that, with AWS Organizations, you can create service control policies (SCPs) that centrally control AWS service use across multiple AWS accounts. SCPs put bounds around the permissions that IAM policies can grant to entities in an account, such as users and roles. For example, SCP policies can be used to allow or deny certain activities so that no principals, not even the *super-user* in a child AWS account can bypass them, thus implementing a form of mandatory access control. This can help contain attackers during the Installation phase. The following table provides more examples of controls that can help contain those that seek unauthorized access.

Table 9: Controls for Contain

Control Names	Descriptions
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts	These controls strong privilege boundaries (AWS accounts) and, the ability to override privileges at the Organization or organizat ional unit (OU) level. They allow users to implement least-privilege and need-to know security principles for both users and services across a multi-account structure. You

Control Names	Descriptions
	can control all privileges in OUs and child accounts.
Amazon EC2 – Linux, SELinux – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC)	This control implements least-privilege account profiles.
Linux cgroups, namespaces, SELinux	These controls enforce capability profiles, which prevent running processes from accessing files, network sockets, and other processes.
Third-Party Security Tools for Containers	This control implements advanced security protection and behavioral security solutions for containers.
Third-Party Security Tools for AWS Lambda Functions	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Container and Abstract Services	These controls can help you prevent access to underlying infrastructure by your customers and threat actors, and segregate your service instances.
Hypervisor-Level Guest-to Guest and Guest-to-Host Segregation	This control leverages the string isolation capabilities of the AWS hypervisor.

Respond

AWS Config monitors the state of an entire swath of cloud services, and can be configured to alert on any unexpected changes. At the instance level, file integrity monitoring (FIM) and enforcement are controls that help maintain the integrity of operating system files and application files by verifying the current file state and a known good baseline of these files. Some of these solutions can help respond during the Installation phase. Additionally, you can use AWS Lambda to automatically execute a function in response to an event to mitigate the threat or stop lateral movement. AWS Systems Manager State Manager and some APN Technology Partners can monitor file integrity.

Table 10: Controls for Respond

Control Names	Descriptions
AWS Systems Manager State Manager	This control helps you to define and maintain consistent OS configurations.
AWS Systems Manager State Manager, or Third-Party or OSS File Integrity Monitoring Solutions on Amazon EC2	This control automates the process of keeping your Amazon EC2 and hybrid infrastructure in a state that you define.
AWS Systems Manager State Manager, AWS Systems Manager Inventory, AWS Config	When new AWS assets are created, or if malware is installed with a regular package, the AWS System Manager Inventory identifies it and sends it to AWS Config for evaluation.
AWS Partner Network Offerings – File Integrity Monitoring	This control helps to maintain the integrity of operating system and application files.
AWS Lambda	A serverless compute service that lets you run code without provisioning or managing servers, creating workload-aware cluster scaling logic, maintaining event integrations, or managing runtimes.

Restore

During the response to an incident, if a tool automatically shuts down a virtual instance of an operating environment, for example, autoscaling can create a new instance automatically from reference images to replace it, thus helping to restore infrastructure automatically and quickly. See the following table for details.

Table 11: Controls for Restore

Control Names	Descriptions
AWS Auto Scaling	This control adjusts capacity to maintain steady, predictable performance.
AWS Systems Manager State Manager	This control helps you to define and maintain consistent OS configurations.
AWS Partner Network Offerings – File Integrity Monitoring	This control helps to maintain the integrity of operating system and application files.
CloudFormation + Service Catalog	These controls help you to provision your infrastructure in an automated and secure manner. The CloudFormation template file serves as the single source of truth for your cloud environment.
Immutable Infrastructure – Short-Lived Environments	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

In addition to the services and functionality listed previously, many other AWS and third-party services and functionality are available courses of action in the Installation phase of the intrusion method. For more information, see the Installation phase section of Appendix: Reference Material.

Using multiple services and features to detect, deny, disrupt, degrade, deceive, contain, respond, and recover, in each phase of the intrusion method can make it increasingly difficult for attackers to be successful. Attackers are faced with the challenge of defeating layers of defensive cybersecurity capabilities in each phase of their intrusion methods, in order to be successful.

Conclusion

Classic intrusion analysis frameworks suffer from a number of weaknesses and limitations when applied to modern cloud platforms. These are particularly problematic when analyzing systems developed with modern DevSecOps practices using cloud-native architectural patterns. Combining a classic intrusion method analysis with the inherent benefits of the cloud, and the plethora of security controls that AWS provides can help many organizations achieve measurably better cybersecurity as compared to on-premises and managed service provider environments, likely at a lower cost.

Contributors

Contributors to this document include:

- Tim Rains, Regional Leader, Security & Compliance Business Acceleration
- Dave Walker, Principal Specialist Solutions Architect, Security and Compliance
- Mark Ryland, Director, Office of the CISO
- Orlando Scott-Cowley, EMEA Regional Leader, Security & Compliance Business Acceleration
- Mark Evans, Regional Leader, Security & Compliance Business Acceleration
- Rob Lyle, Specialist Solutions Architect Manager, Security and Networking
- Rob Samuel, Principal, Security Assurance
- Andrew Hodges, Senior Security Advisor
- Stephen Alexander, Senior Manager Enterprise Support
- Kriti Gera, Associate Solutions Architect
- Kush Vyas, Associate Solutions Architect

Further reading

For additional information, see:

- AWS Cloud Security
- AWS Security Services
- AWS Security Resources
- Cloud Endure Disaster Recovery
- Disaster Recovery of Workloads on AWS
- AWS Security Bulletins
- AWS Cloud Adoption Framework Security Perspective
- Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains
- Security Pillar AWS Well-Architected Framework

Document history

To be notified about updates to this whitepaper, subscribe to the RSS feed.

Change	Description	Date
Minor update	Fix non-inclusive language.	April 6, 2022
Initial release	Whitepaper first published.	March 31, 2021

Appendix: Reference material

This reference document contains an example of how AWS services and functionality can be mapped to a classic intrusion analysis framework. Although this example mapping is comprehensive, it's not exhaustive. There are additional controls that are not included in the example mapping that you should identify and leverage as appropriate.

Topics

Reconnaissance – Pre-Intrusion

This phase represents the work attackers do to research and select their targets, and understand their targets' digital footprints. This can include reconnaissance activities such as port scans and vulnerability scans of the targets' publicly accessible systems and of their supply chain partners.

Reconnaissance pre-intrusion activities occur prior to intrusion attempts. Examples include unusual API activity, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.

Control Objective – Detect

The objective of the *Detect* control in the *Reconnaissance Pre-Intrusion* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems."[**]

Control Names	Descriptions
Amazon GuardDuty (ID: Sec.Det.1)	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.
Amazon GuardDuty Partners (ID: Sec.Det.2)	These controls are a complement to Amazon GuardDuty.
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	Malicious sources scan and probe Internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes, and you can use this history to help identify and block malicious source IP addresses.
Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third Parties (ID: Sec.Det.6)	These controls help you to monitor, detect, visualize, receive notifications, and respond to changes in your AWS resources.
AWS Security Hub (ID: Sec.Det.3)	This control gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Amazon Detective (ID: Sec.Det.11)	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspiciou

Control Names	Descriptions
	s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.
AWS Network Firewall (ID: Sec.Inf.30)	This control detects reconnaissance activity using signature-based detection.

Control Objective – Deny

The objective of the *Deny* control in the *Reconnaissance Pre-Intrusion* phase is to "prevent the adversary from accessing and using critical information, systems, and services."**

Control Names	Descriptions
Amazon Virtual Private Cloud (Amazon VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnais sance. Amazon VPC Black Hole Routes (as an allow list or deny list of network reachable assets before Security Groups or NACLs).
AWS Identity and Access Management + AWS Organizations (ID: Sec.IAM.3)	In this context, attackers can't execute <service>:Describe* API calls without Allow permissions.
AWS Certificate Manager + Transport Layer Security (ID: Sec.DP.3)	Protecting data in transit denies attackers the ability to capture data in transit during the Reconnaissance phase, unless they are able to impersonate a legitimate endpoint.

Control Names	Descriptions
Network Infrastructure Solutions in the AWS Marketplace (ID: Sec.Inf.10)	Infrastructure solutions in the AWS Marketplace can help deny attackers access to data and infrastructure as they conduct reconnaissance.
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	This control is a solution that leverages automation to quickly and easily configure AWS WAF rules that help block Scanners and Probes, Known Attacker Origins, and Bots and Scrapers solutions.
AWS Direct Connect (ID: Sec.Inf.4)	This control establishes private connectivity to multiple Amazon VPCs.
AWS Network Firewall (ID: Sec.Inf.30)	The control blocks network scanning during the reconnaissance phase by blocking network scans and probes utilizing signature based intrusion prevention.

Control Objective – Disrupt

The objective of the *Disrupt* control in the *Reconnaissance Pre-Intrusion* phase is to “break or interrupt the flow of information.” **

Control Names	Descriptions
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to block traffic associated with an attack.
AWS Network Firewall (ID: Sec.Inf.30)	The control detects reconnaissance activity, blocking network scans and probes utilizing signature based intrusion prevention.

Control Objective – Degrade

The objective of the *Degrade* control in the *Reconnaissance Pre-Intrusion* phase is to "reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means."

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.

Control Objective – Deceive

The objective of the *Deceive* control in the *Reconnaissance Pre-Intrusion* phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality."**

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.
AWS WAF + AWS Lambda (ID: Sec.IR.2)	These controls trap the endpoint to detect content scrapers and bad bots. When the endpoint is accessed, a function adds the source IP address to a block list.

Control Objective – Contain

The objective of the *Contain* control in the *Reconnaissance Pre-Intrusion* phase is "keeping something harmful under control or within limits." **

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.

Control Objective – Respond

The objective of the Respond control in the *Reconnaissance Pre-Intrusion* phase is to provide "capabilities that help to react quickly to an adversary's or others' IO attack or intrusion." **

Control Names	Descriptions
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	Malicious sources scan and probe internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes. You can use this history to help identify and block malicious source IP addresses.
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activitie s and modify security configurations to block traffic associated with an attack.
Amazon GuardDuty Partners (ID: Sec.Det.2)	These controls are a complement to Amazon GuardDuty.

Control Names	Descriptions
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
Amazon CloudWatch Events & Alarms + Amazon SNS + SIEM Solutions (ID: Sec.Det.7)	These controls monitor, detect, visualize , receive notification about attacks, and respond to changes in your AWS resources.

Reconnaissance – Post-Intrusion

Activities in this phase occur after attacker's intrusion attempts have been successful. Attackers perform reconnaissance inside their victim's environment in an effort to build a map for themselves, which can then be referenced throughout the attack. These activities could include port scanning, ping sweeps, Windows Management Instrumentation (WMI) queries, and SNMP queries.

Control Objective – Detect

The objective of the *Detect* control in the *Reconnaissance Post-Intrusion* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems." [**]

Control Names	Descriptions
Amazon GuardDuty (ID: Sec.Det.1)	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.
Amazon GuardDuty Partners (ID: Sec.Det.2)	These controls are a complement to Amazon GuardDuty.

Control Names	Descriptions
Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third-Parties (ID: Sec.Det.6)	These controls monitor, detect, visualize, and receive notifications of attacks, and respond to changes in your AWS resources.
AWS Security Hub (ID: Sec.Det.3)	This control gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
Amazon Detective (ID: Sec.Det.11)	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspicious activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.
AWS Network Firewall (ID: Sec.Inf.30)	This control detects reconnaissance activity using signature-based detection.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.

Control Objective – Deny

The objective of the *Deny* control in the *Reconnaissance Post-Intrusion* phase is to “prevent the adversary from accessing and using critical information, systems, and services.” [**]

Control Names	Descriptions
Amazon Virtual Private Cloud (Amazon VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.
AWS Identity and Access Management + AWS Organizations (ID: Sec.IAM.3)	In this context, attackers can't execute <service>:Describe* API calls without Allow permissions.
AWS Certificate Manager + Transport Layer Security (ID: Sec.DP.3)	Protecting data in transit denies attackers the ability to capture data in transit during the Reconnaissance phase, unless they are able to impersonate a legitimate endpoint.
Network Infrastructure Solutions in the AWS Marketplace (ID: Sec.Inf.10)	Infrastructure solutions in the AWS Marketpla ce can help deny attackers access to data and infrastructure as they conduct reconnaissance.
Reverse Proxy Architecture (ID: Sec.Inf.11)	This control protects your servers from unwanted traffic.
Amazon Cognito (ID: Sec.IAM.5)	This control provides temporary, limited-p rivilege AWS credentials to allow access to other AWS services.
Bottlerocket (ID: Sec.Inf.32)	This control provides a minimized OS environment capable of running and managing containers, which provides no extraneous listeners or services.
AWS Network Firewall (ID: Sec.Inf.30)	The control blocks network scanning during the reconnaissance phase by blocking network

Control Names	Descriptions
	scans and probes utilizing signature based intrusion prevention.

Control Objective – Disrupt

The objective of the *Disrupt* control in the *Reconnaissance Post-Intrusion* phase is to "break or interrupt the flow of information." **

Control Names	Descriptions
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activitie s and modify security configurations to block traffic associated with an attack.
AWS Network Firewall (ID: Sec.Inf.30)	The control detects reconnaissance activity, blocking network scans and probes utilizing signature based intrusion prevention.

Control Objective – Degrade

The objective of the *Degrade* control in the *Reconnaissance Post-Intrusion* phase is to "reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means." **

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.

Control Objective – Deceive

The objective of the *Deceive* control in the *Reconnaissance Post-Intrusion* phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality." **

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.
Amazon Virtual Private Cloud + Automation (ID: Sec.IR.9)	These controls save the current security group of the host or instance, then isolate the host using restrictive ingress and egress security group rules.

Control Objective – Contain

The objective of the *Contain* control in the *Reconnaissance Post-Intrusion* phase is "keeping something harmful under control or within limits." **

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.
Amazon Virtual Private Cloud + Automation	These controls help contain compromis ed systems by using AWS Command Line

Control Names	Descriptions
(ID: Sec.IR.9)	Interface (CLI) or software development kits using predefined, restrictive security groups.

Control Objective – Respond

The objective of the *Respond* control in the *Reconnaissance Post-Intrusion* phase is to provide "capabilities that help to react quickly to an adversary's or others' IO attack or intrusion." **

Control Names	Descriptions
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	Malicious sources scan and probe internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes. You can use this history to help identify and block malicious source IP addresses.
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activitie s and modify security configurations to block traffic associated with an attack.
Amazon GuardDuty Partners (ID: Sec.Det.2)	These controls are a complement to Amazon GuardDuty.
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
Amazon CloudWatch Events & Alarms + Amazon SNS + SIEM Solutions (ID: Sec.Det.7)	These controls help you to monitor, detect, visualize, and receive notifications of attacks, so you can respond to changes in your AWS resources.

Exploit Development

There are no products or services included for this phase because *Exploit Development* typically occurs in secret, outside of the view of defenders. Subsequently, during this phase of the intrusion method, products and services cannot detect, deny, disrupt, degrade, or complete any other objective.

Delivery

During the *Delivery* phase in the intrusion method, attackers transmit their weapon to the intended victim. Some examples of delivery mechanisms include phishing emails, malicious email attachments, and drive-by download sites.

Control Objective – Detect

The objective of the *Detect* control in the *Delivery* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems." [**]

Control Names	Descriptions
Amazon GuardDuty (ID: Sec.Det.1)	Detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known bad IP address.
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	Malicious sources scan and probe internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes. You can use this history to help identify and block malicious source IP addresses.
AWS Shield (ID: Sec.Inf.13)	This control defends against most common, frequently occurring network and transport layer DDoS attacks that target your website or applications.

Control Names	Descriptions
Amazon VPC Flow Logs + Amazon CloudWatch Alarms (ID: Sec.Det.8)	These controls capture and monitor informati on about the IP traffic going to and from your Amazon VPC.
Amazon Detective (ID: Sec.Det.11)	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspiciou s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.
AWS IoT Device Defender + AWS IoT SiteWise (ID: Sec.Det.9)	Detects and provides analytics capabilities for anomalous behavior in IoT Things
Amazon CloudWatch Logs + Amazon Lookout for Metrics (ID: Sec.Det.10)	Detects and provides analytics capabilities for anomalous behavior in assets and services which send logs to CloudWatch Logs (subject to level of detail of logs being gathered)

Control Objective – Deny

The objective of the *Deny* control in the *Delivery* phase is to "prevent the adversary from accessing and using critical information, systems, and services." **

Control Names	Descriptions
Amazon Virtual Private Cloud (VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list

Control Names	Descriptions
	of network reachable assets, before Security Groups or NACLs.
Amazon Virtual Private Cloud VPN Gateway + AWS Direct Connect (ID: Sec.Inf.4)	These controls establish private connectivity to multiple Amazon VPCs.
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
AWS Shield (ID: Sec.Inf.13)	This control defends against most common, frequently occurring network and transport layer DDoS attacks that target your website or applications.
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls implement strong, least-pri vilege and need-to-know security principles for both users and services that access your resources.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies (ID: Sec.DP.6)	These controls specify access privileges to objects and prevent the upload of that malicious objects into the bucket.

Control Names	Descriptions
Amazon Cognito (ID: Sec.IAM.5)	This control provides temporary, limited-p rivilege AWS credentials to allow access to other AWS services.
Amazon EC2: Linux: SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux, FreeBSD – Hardening and Minimization (ID: Sec.Inf.19)	These controls disable or remove unused services and packages.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Microsoft Windows Security Baselines (ID: Sec.Inf.24)	This control allows you to harden system and user configurations.
AWS Physical & Operational Security Policies & Processes (ID: Platform.5)	AWS data centers are secure by design and our controls make that possible. We spend countless hours considering potential threats and designing, implementing, and testing controls to ensure the systems, technology, and people we deploy counteract risks.

Control Names	Descriptions
Bottlerocket (ID: Sec.Inf.32)	This control provides a minimized OS environment capable of running and managing containers, which provides no extraneous listeners or services.
AWS Network Firewall (ID: Sec.Inf.30)	Provides deep-packet inspection filtering of VPC network traffic using Suricata-syntax rules
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environme nt for signed code to handle sensitive data, accessible only by local virtual network socket interface
Amazon Simple Email Service (Amazon SES) (ID: Sec.Inf.31)	Supports content filtering on inbound and outbound email

Control Objective – Disrupt

The objective of the *Disrupt* control in the *Delivery* phase is to "break or interrupt the flow of information." [**]

Control Names	Descriptions
Amazon Virtual Private Cloud (Amazon VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.

Control Names	Descriptions
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
AWS Shield (ID: Sec.Inf.13)	This control defends against most common, frequently occurring network and transport layer DDoS attacks that target your website or applications.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	Rebuilt or refresh your environments periodica lly to make it more difficult for an attack payload to persist.
AWS Network Firewall (ID: Sec.Inf.30)	Provides deep-packet inspection filtering of VPC network traffic using Suricata-syntax rules
AWS IoT Device Defender + AWS IoT SiteWise (ID: Sec.Det.9)	Detects and provides analytics capabilities and customizable response automation for anomalous behavior in IoT Things
Amazon CloudWatch Logs + Amazon Lookout for Metrics (ID: Sec.Det.10)	Detects and provides analytics capabilities for anomalous behavior in assets and services which send logs to CloudWatch Logs (subject to level of detail of logs being gathered)

Control Objective – Degrade

The objective of the *Degrade* control in the *Delivery* phase is to "reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means." [**]

Control Names	Descriptions
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to degrade or block traffic associated with an attack.
AWS Shield (ID: Sec.Inf.13)	This control defends against most common, frequently occurring network and transport layer DDoS attacks that target your website or applications.
Load Balancing (ID: Sec.Inf.8)	With this control, before an attacker can consistently communicate with your resources , all the instances included in the load-bala nced service need to be compromised by the attack. If one or more instances has not been compromised, the load balancer switches to an unaffected instance, which degrades the attack.
Immutable Infrastructure - Short-Lived Environments (ID: Ops.2)	Rebuilt or refresh your environments periodica lly to make it more difficult for an attack payload to persist.

Control Objective – Deceive

The objective of the *Deceive* control in the *Delivery* phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality." [**]

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.

Control Names	Descriptions
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.
AWS WAF + AWS Lambda (ID: Sec.IR.2)	These controls trap endpoints to detect content scrapers and bad bots. When the endpoint is accessed a function adds the source IP address to a blocked list.

Control Objective – Contain

The objective of the *Contain* control in the *Delivery* phase is the "action of keeping something harmful under control or within limits." [**]

Control Names	Descriptions
AWS WAF (ID: Sec.Inf.1)	This control helps to protect your network from common web exploits that could affect application availability, compromise security, or consume excessive resources.
Amazon Virtual Private Cloud (Amazon VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.

Control Names	Descriptions
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
AWS Lambda, Amazon Simple Queue Service (Amazon SQS), AWS Step Functions (ID: Platform.2)	These services provide orchestration mechanisms for containment.
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environme nt for signed code to handle sensitive data, accessible only by local virtual network socket interface

Control Objective – Respond

The objective of the *Respond* control in the *Delivery* phase is to provide "capabilities that help to react quickly to an adversary's or others' IO attack or intrusion." [**]

Control Names	Descriptions
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	These controls help you to maintain the integrity of operating system and application files.
AWS WAF + AWS Lambda (ID: Sec.IR.2)	These controls trap endpoints to detect content scrapers and bad bots. When the endpoint is accessed, a function adds the source IP address to a blocked list.

Control Names	Descriptions
Third-Party WAF Integrations (ID: Sec.IR.3)	These controls are a complement to AWS WAF.
AWS Config Rules (ID: Sec.IR.5)	These rules are a configurable set of functions that trigger when an environment configura tion change is registered.
Amazon CloudWatch Events + Lambda (ID: Sec.IR.6)	These controls are a configurable set of functions that trigger when an environment configuration change is registered.
AWS Managed Services (ID: Ops.3)	AWS Managed Services monitors the overall health of your infrastructure resources, and handles the daily activities of investigating and resolving alarms or incidents.
AWS IoT Device Defender + AWS IoT SiteWise (ID: Sec.Det.9)	Detects and provides analytics capabilities and customizable response automation for anomalous behavior in IoT Things
Amazon CloudWatch Logs + Amazon Lookout for Metrics + Lambda (ID: Sec.Det.10)	Detects and provides analytics and response capabilities for anomalous behavior in assets and services which send logs to CloudWatch Logs (subject to level of detail of logs being gathered)

Control Objective – Restore

The objective of the *Restore* control in the *Delivery* phase is to “bring information and information systems back to their original state.” **

Control Names	Descriptions
AWS Systems Manager State Manager	This control helps you to define and maintain consistent OS configurations.

Control Names	Descriptions
(ID: Sec.Inf.14)	
CloudFormation + Service Catalog (ID: Ops.1)	These controls help you to provision your infrastructure in an automated and secure manner. The CloudFormation template file serves as the single source of truth for your cloud environment.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	Rebuilt or refresh your environments periodica lly to make it more difficult for an attack payload to persist.

Exploitation

In the *Exploitation* phase, after the weapon has been delivered to the target, the weapon tries to exploit the weakness it was designed for. This could be the exploitation of a vulnerability or misconfiguration in an operating system, web browser, or other application. An exploit can also be designed to trick people into making poor trust decisions, which is also known as *social engineering*. Another weakness that attackers typically try to exploit is weak, leaked, or stolen passwords.

Control Objective – Detect

The objective of the *Detect* control in the *Exploitation* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems." [**]

Control Names	Descriptions
Amazon GuardDuty (ID: Sec.Det.1)	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.
Amazon Detective	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root

Control Names	Descriptions
(ID: Sec.Det.11)	cause of potential security issues or suspiciou s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations."
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	Malicious sources scan and probe internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes. You can use this history to help identify and block malicious source IP addresses.
Amazon Virtual Private Cloud (Amazon VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.
AWS Config (ID: Sec.Det.5)	With this control, you can assess, audit, and evaluate the configurations of your AWS resources.
Third-party container security tools and services (ID: Sec.IR.14 and 35)	These controls complement to the security properties of containers solutions.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.

Control Names	Descriptions
AWS Partner Offerings – Anti-Malware Protection (ID: Sec.IR.12)	These controls help to detect and block malicious payloads.
AWS Lambda Partners (ID: Sec.Inf.27)	These controls are a complement to the security properties of Lambda functions.
AWS IoT Device Defender + AWS IoT SiteWise (ID: Sec.Det.9)	Detects and provides analytics capabilities for anomalous behavior in IoT Things
Amazon CloudWatch Logs + Amazon Lookout for Metrics (ID: Sec.Det.10)	Detects and provides analytics capabilities for anomalous behavior in assets and services which send logs to CloudWatch Logs (subject to level of detail of logs being gathered)

Control Objective – Deny

The objective of the *Deny* control in the *Exploitation* phase is to "prevent the adversary from accessing and using critical information, systems, and services." **

Control Names	Descriptions
Roles (ID: Sec.IAM.1)	These controls help deny or contain the blast radius of attacks.
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies (ID: Sec.DP.6)	These controls manage access to objects and prevent upload of malicious objects into the S3 bucket.
AWS Secrets Manager (ID: Sec.DP.7)	This control protects the secrets needed to access your applications, services, and IT resources.

Control Names	Descriptions
Amazon – EC2 Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux, FreeBSD – Hardening and Minimization (ID: Sec.Inf.19)	These controls disable or remove unused services and packages.
Amazon EC2 – Linux, Windows, FreeBSD – Address Space Layout Randomization (ASLR) (ID: Sec.Inf.20)	ASLR is a technology that helps prevent shellcode from being successful.
Amazon EC2 – Linux, Windows, FreeBSD – Data Execution Prevention (DEP) (ID: Sec.Inf.21)	DEP is a memory safety feature that makes it more difficult for malware to run.
Amazon EC2 – Windows – User Account Control (UAC) (ID: Sec.Inf.22)	UACs make it more difficult for malware to install and run.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Microsoft Windows Security Baselines (ID: Sec.Inf.24)	This control hardens system and user configurations.

Control Names	Descriptions
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Offerings – Anti-Malware Protection (ID: Sec.IR.12)	This control helps to detect and blocks malicious payloads.
AWS Lambda Partners (ID: Sec.Inf.27)	This control is a complement to the security properties of Lambda functions.
Container Partners – Security (ID: Sec.Inf.28)	This control is a complement to the security properties of containers solutions.
Amazon Simple Email Service Spam and Virus Protection (ID: Platform.3)	This control prevents mail from known spammers, or containing malware, from entering the system.
Bottlerocket (ID: Sec.Inf.32)	This control provides a minimised OS environment capable of running and managing containers, which provides no extraneous listeners or services.
AWS Key Management Service (AWS KMS) + AWS CloudHSM (ID: Sec.DP.1)	These services deny access to clear text of encryption keys.

Control Names	Descriptions
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environme nt for signed code to handle sensitive data, accessible only by local virtual network socket interface.

Control Objective – Disrupt

The objective of the *Disrupt* control in the *Exploitation* phase is to "break or interrupt the flow of information." **

Control Names	Descriptions
AWS WAF, WAF Managed Rules + Automation (ID: Sec.Inf.2)	Malicious sources scan and probe internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes. You can use this history to help identify and block malicious source IP addresses.
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies (ID: Sec.DP.6)	These controls manage access to objects and prevent uploads of malicious objects into the bucket.
AWS Secrets Manager (ID: Sec.DP.7)	This control protects the secrets needed to access your applications, services, and IT resources.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.

Control Names	Descriptions
(ID: Sec.Inf.18)	
Amazon EC2 – Linux, Windows, FreeBSD – Address Space Layout Randomization (ASLR) (ID: Sec.Inf.20)	ASLR is a technology that helps prevent shellcode from being successful.
Amazon EC2 – Linux, Windows, FreeBSD – Data Execution Prevention (DEP) (ID: Sec.Inf.21)	DEP is a memory safety feature that makes it more difficult for malware to run.
Amazon EC2 – Windows – User Account Control (UAC) (ID: Sec.Inf.22)	UACs make it more difficult for malware to install and run.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Offerings – Anti-Malware Protection (ID: Sec.IR.12)	This control helps to detect and block malicious payloads.

Control Names	Descriptions
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	This control rebuilds or refreshes environments periodically to make it more difficult for attack payloads to persist.
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environment for signed code to handle sensitive data, accessible only by local virtual network socket interface.
AWS Network Firewall (ID: Sec.Inf.30)	Provides deep-packet inspection filtering of VPC network traffic using Suricata-syntax rules.

Control Objective – Degrade

The objective of the *Degrade* control in the *Exploitation* phase is to "reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means." **

Control Names	Descriptions
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to degrade or block traffic associated with an attack.
AWS WAF (ID: Sec.Inf.1)	This control helps to protect you from common web exploits that could affect application availability, compromise security, or consume excessive resources.
Load Balancing (ID: Sec.Inf.8)	With this control, before an attacker can consistently communicate with your resources, all the instances included in the load-balanced service need to be compromised by the attack. If one or more instances has not been

Control Names	Descriptions
	compromised, the load balancer switches to an unaffected instance, which degrades the attack.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.
AWS Network Firewall (ID: Sec.Inf.30)	Provides deep-packet inspection filtering of VPC network traffic using Suricata-syntax rules

Control Objective – Deceive

The objective of the Deceive control in the Exploitation phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality."[**]

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.
AWS WAF + AWS Lambda (ID: Sec.IR.2)	These controls trap endpoints to detect content scrapers and bad bots. When the endpoint is accessed, a function adds the source IP address to a blocked list.

Control Objective – Contain

The objective of the *Contain* control in the *Exploitation* phase is the "action of keeping something harmful under control or within limits." **

Control Names	Descriptions
Roles (ID: Sec.IAM.1)	These controls help you to deny or contain the blast radius of attacks.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux, FreeBSD – Hardening and Minimization (ID: Sec.Inf.19)	These controls disable or remove unused services and packages.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Linux cgroups, namespaces, SELinux (ID: Sec.Inf.25)	These controls enforce capability profiles, which prevent running processes from

Control Names	Descriptions
	accessing files, network sockets, and other processes.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Container and Abstract Services (ID: Platform.1)	These controls can help you prevent access to underlying infrastructure by your customers and threat actors, and segregate your service instances.
Hypervisor-Level Guest-To-Guest and Guest-To-Host Separation (ID: Platform.4)	This control leverages the string isolation capabilities provided by the AWS hypervisor.
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environme nt for signed code to handle sensitive data, accessible only by local virtual network socket interface

Control Objective – Respond

The objective of the *Respond* control in the *Exploitation* phase is to provide "Capabilities that help to react quickly to an adversary's or others' IO attack or intrusion." [**]

Control Names	Descriptions
Amazon GuardDuty Partners (ID: Sec.Det.2)	These controls are a complement to Amazon GuardDuty.

Control Names	Descriptions
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Offerings – Behavioral Monitoring, Response Tools and Services (ID: Sec.Inf.29)	These controls provide insights into the threats in your environment.
AWS Managed Services (ID: Ops.3)	AWS Managed Services monitors the overall health of your infrastructure resources, and handles the daily activities of investigating and resolving alarms or incidents.
Amazon CloudWatch Logs + Amazon Lookout for Metrics + Lambda (ID: Sec.Det.10)	Detects and provides analytics capabilities and response for anomalous behavior in assets and services which send logs to CloudWatch Logs (subject to level of detail of logs being gathered)

Control Objective – Restore

The objective of the Restore control in the Exploitation phase is to “bring information and information systems back to their original state.” **

Control Names	Descriptions
AWS Auto Scaling (ID: Sec.Inf.9)	This control adjusts capacity to maintain steady, predictable performance.

Control Names	Descriptions
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	These controls help you to maintain the integrity of operating system and application files.
CloudFormation + Service Catalog (ID: Ops.1)	These controls help you to provision your infrastructure in an automated and secure manner. The CloudFormation template file serves as the single source of truth for your cloud environment.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Installation

In the *Installation* phase, after vulnerabilities have been successfully exploited, many attackers will attempt to persist undetected in the environment as long as possible, in order to accomplish their objectives. In this phase, attackers will attempt to install tools that allow them to maintain remote access to the victim's environment.

Control Objective – Detect

The objective of the *Detect* control in the *Installation* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems." [**]

Control Names	Descriptions
Amazon GuardDuty	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port

Control Names	Descriptions
(ID: Sec.Det.1)	scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.
Amazon Detective (ID: Sec.Det.11)	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspiciou s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.
Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third-Par ties (ID: Sec.Det.6)	These controls monitor, detect, visualize, and receive notifications of attacks, and respond to changes in your AWS resources
AWS Security Hub (ID: Sec.Det.3)	This control gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
AWS Systems Manager State Manager, AWS Systems Manager Inventory, AWS Config (ID: Sec.Inf.16)	When new AWS assets are created, or if malware is installed with a regular package, the AWS Systems Manager Inventory identifies it and sends it to AWS Config for evaluation.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for containers.

Control Names	Descriptions
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Offerings – Anti-Malware Protection (ID: Sec.IR.12)	These controls help to detect and block malicious payloads.
AWS IoT Device Defender + AWS IoT SiteWise (ID: Sec.Det.9)	Detects and provides analytics capabilities and customizable response automation for anomalous behavior in IoT Things
Amazon CloudWatch Logs + Amazon Lookout for Metrics (ID: Sec.Det.10)	Detects and provides analytics capabilities for anomalous behavior in assets and services which send logs to CloudWatch Logs (subject to level of detail of logs being gathered)

Control Objective – Deny

The objective of the *Deny* control in the *Installation* phase is to "prevent the adversary from accessing and using critical information, systems, and services." **

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.

Control Names	Descriptions
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies (ID: Sec.DP.6)	These controls manage access to objects and prevent upload of malicious objects into the S3 bucket.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux, FreeBSD – Hardening and Minimization (ID: Sec.Inf.19)	These controls disable or remove unused services and packages.
Amazon EC2 – Windows – User Account Control (UAC) (ID: Sec.Inf.22)	UACs make it more difficult for malware to install and run.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Amazon EC2 – Windows – Device Guard (ID: Sec.Inf.26)	This control specifies which binaries are authorized to run on your server.
AWS Partner Offerings – Anti-Malware Protection (ID: Sec.IR.12)	These controls help to detect and block malicious payloads.

Control Names	Descriptions
Bottlerocket (ID: Sec.Inf.32)	This control provides a minimized OS environment capable of running and managing containers, which provides no extraneous listeners or services.
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environme nt for signed code to handle sensitive data, accessible only by local virtual network socket interface

Control Objective – Disrupt

The objective of the *Disrupt* control in the *Installation* phase is to "break or interrupt the flow of information." [**]

Control Names	Descriptions
Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies (ID: Sec.DP.6)	These controls manage access to objects and prevent upload of malicious objects into the S3 bucket.
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.

Control Names	Descriptions
Amazon EC2 – Windows – User Account Control (UAC) (ID: Sec.Inf.22)	UACs make it more difficult for malware to install and run.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Amazon EC2 – Windows – Device Guard (ID: Sec.Inf.26)	This control specifies which binaries are authorized to run on your server.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	This control helps to maintain the integrity of operating system and application files.
AWS Partner Offerings – Anti-Malware Protection (ID: Sec.IR.12)	These controls help to detect and block malicious payloads.

Control Objective – Degrade

The objective of the *Degrade* control in the *Installation* phase is to "reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means." **

Control Names	Descriptions
Load Balancing (ID: Sec.Inf.8)	With this control, before an attacker can consistently communicate with your resources , all the instances included in the load-bala nced service need to be compromised by the

Control Names	Descriptions
	attack. If one or more instances has not been compromised, the load balancer switches to an unaffected instance, which degrades the attack.
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.
Amazon EC2 – Linux, FreeBSD – Hardening and Minimization (ID: Sec.Inf.19)	These controls disable or remove unused services and packages.
Amazon EC2 – Windows – Device Guard (ID: Sec.Inf.26)	This control specifies which binaries are authorized to run on your server.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	This control helps to maintain the integrity of operating system and application files.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Control Objective – Deceive

The objective of the *Deceive* control in the *Installation* phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality."**

Control Names	Descriptions
Honeypot and Honeynet Environments	These controls help to degrade, detect, and contain attacks.

Control Names	Descriptions
(ID: Sec.IR.10)	
Honeywords and Honeykeys (ID: Sec.IR.11)	When an attacker attempts to use stolen, false credentials, these controls help to detect and contain the attack, so you can recover faster.

Control Objective– Contain

The objective of the *Contain* control in the *Installation* phase is the "action of keeping something harmful under control or within limits." **

Control Names	Descriptions
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Linux cgroups, namespaces, SELinux	These controls enforce capability profiles, which prevent running processes from

Control Names	Descriptions
(ID: Sec.Inf.25)	accessing files, network sockets, and other processes.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Container and Abstract Services (ID: Platform.1)	These controls can help you prevent access to underlying infrastructure by your customers and threat actors, and segregate your service instances.
Hypervisor-Level Guest-to-Guest and Guest-to-Host Separation (ID: Platform.4)	This control leverages the string isolation capabilities of the AWS hypervisor.
AWS Nitro Enclaves (ID: Sec.DP.5)	Provides an isolated execution environme nt for signed code to handle sensitive data, accessible only by local virtual network socket interface.

Control Objective – Respond

The objective of the *Respond* control in the *Installation* phase is to provide “Capabilities that help to react quickly to an adversary’s or others’ IO attack or intrusion.” **

Control Names	Descriptions
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.

Control Names	Descriptions
AWS Systems Manager State Manager, or Third-Party or OSS File Integrity Monitoring Solutions on Amazon EC2 (ID: Sec.Inf.15)	This control automates the process of keeping your Amazon EC2 and hybrid infrastructure in a state that you define.
AWS Systems Manager State Manager, AWS Systems Manager Inventory, AWS Config (ID: Sec.Inf.16)	When new AWS assets are created, or if malware is installed with a regular package, the AWS Systems Manager Inventory identifies it and sends it to AWS Config for evaluation.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	This control helps to maintain the integrity of operating system and application files.
AWS Security Hub Automated Response and Remediation (ID: Sec IR.7)	AWS Security Hub Automated Response and Remediation is an add-on solution that works with AWS Security Hub to provide a ready-to-deploy architecture and a library of automated playbooks Refer AWS Security Blog : How to deploy the AWS Solution for Security Hub Automated Response and Remediation

Control Objective – Restore

The objective of the *Restore* control in the *Installation* phase is to “bring information and information systems back to their original state.” **

Control Names	Descriptions
AWS Auto Scaling (ID: Sec.Inf.9)	This control adjusts capacity to maintain steady, predictable performance.

Control Names	Descriptions
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	This control helps to maintain the integrity of operating system and application files.
CloudFormation + Service Catalog (ID: Ops.1)	These controls help you to provision your infrastructure in an automated and secure manner. The CloudFormation template file serves as the single source of truth for your cloud environment.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Command and Control

In the *Command and Control* (C2) phase, attackers maintain illicit access to their victims' environments and can remotely control compromised infrastructure.

Control Objective – Detect

The objective of the *Detect* control in the *C2* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems." [**]

Control Names	Descriptions
Amazon GuardDuty (ID: Sec.Det.1)	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login

Control Names	Descriptions
	requests, or unblocked port probing from a known, bad IP address.
Amazon Detective (ID: Sec.Det.11)	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspiciou s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.
Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third Parties (ID: Sec.Det.6)	These controls help you to monitor, detect, visualize, receive notifications, and respond to changes in your AWS resources.
AWS Security Hub (ID: Sec.Det.3)	This control gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
Outbound Proxy Partners (ID: Sec.Inf.8)	Because it is an intermediary for requests, this control can detect malicious traffic before it reaches your network.
AWS EC2 Forward Proxy Servers (ID: Sec.Inf.12)	Because it is in intermediary for requests, this control can detect malicious traffic before it reaches your network.

Control Names	Descriptions
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.

Control Objective – Deny

The objective of the *Deny* control in the *C2* phase is to "prevent the adversary from accessing and using critical information, systems, and services." **

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon Cognito (ID: Sec.IAM.5)	This control provides temporary, limited-p rivilege AWS credentials to allow access to other AWS services.
Amazon Virtual Private Cloud (VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list

Control Names	Descriptions
	of network reachable assets, before Security Groups or NACLs.
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.

Control Objective – Disrupt

The objective of the *Disrupt* control in the *C2* phase is to "break or interrupt the flow of information." [**]

Control Names	Descriptions
Amazon Virtual Private Cloud (VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.

Control Names	Descriptions
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to degrade or block traffic associated with an attack.
Amazon GuardDuty + AWS Lambda + AWS WAF, Security Groups, NACLs (ID: Sec.IR.4)	These controls detect attacks and modify security configurations to block traffic associated with an attack.
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Control Objective – Degrade

The objective of the *Degrade* control in the *C2* phase is to “reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means.” **

Control Names	Descriptions
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to degrade or block traffic associated with an attack.
Amazon GuardDuty + AWS Lambda + AWS WAF, Security Groups, NACLs (ID: Sec.IR.4)	These controls detect attacks and modify security configurations to block traffic associated with an attack.
–Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Control Objective – Deceive

The objective of the *Deceive* control in the *C2* phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality." [**]

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.

Control Objective – Contain

The objective of the *Contain* control in the *C2* phase is "keeping something harmful under control or within limits." [**]

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries	These controls provide strong, least-privilege and need-to-know security principles for both

Control Names	Descriptions
(ID: Sec.IAM.2)	the users and services that can access your resources.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon Virtual Private Cloud (VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
AWS Network Firewall (ID: Sec.Inf.30)	AWS Network Firewall's flexible rules engine lets you define firewall rules that give you fine-grained control over network traffic, such as blocking outbound Server Message Block (SMB) requests to prevent the spread of malicious activity.
Linux cgroups, namespaces, SELinux (ID: Sec.Inf.25)	These controls enforce capability profiles, which prevent running processes from accessing files, network sockets, and other processes.

Control Names	Descriptions
AWS Container and Abstract Services (ID: Platform.1)	These controls can help you prevent access to underlying infrastructure by your customers and threat actors, and segregate your service instances.
Hypervisor-Level Guest-to-Guest and Guest-to-Host Separation (ID: Platform.4)	This control leverages the string isolation capabilities of the AWS hypervisor.
AWS Lambda, Amazon Simple Queue Service (Amazon SQS), AWS Step Functions (ID: Platform.2)	These services provide orchestration mechanisms for containment.

Control Objective – Respond

The objective of the *Respond* control in the *C2* phase is to provide "Capabilities that help to react quickly to an adversary's or others' IO attack or intrusion." [**]

Control Names	Descriptions
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Offerings – Behavioral Monitoring, Response Tools and Services (ID: Sec.Inf.29)	These controls provide insight into the threats in your environment.

Control Names	Descriptions
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to block traffic associated with an attack.
AWS Managed Services (ID: Ops.3)	AWS Managed Services monitors the overall health of your infrastructure resources, and handles the daily activities of investigating and resolving alarms or incidents.

Control Objective – Restore

The objective of the *Restore* control in the *C2* phase is to "bring information and information systems back to their original state." **

Control Names	Descriptions
AWS Auto Scaling (ID: Sec.Inf.9)	This control adjusts capacity to maintain steady, predictable performance.
AWS Systems Manager State Manager (ID: Sec.Inf.14)	This control helps you to define and maintain consistent OS configurations.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	This control helps to maintain the integrity of operating system and application files.
CloudFormation + Service Catalog (ID: Ops.1)	These controls help you to provision your infrastructure in an automated and secure manner. The CloudFormation template file serves as the single source of truth for your cloud environment.

Control Names	Descriptions
Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.
AWS DR Options (ID: Ops.4)	These controls can help you rapidly recover your IT infrastructure and data.

Actions on Objectives

At in the *Actions* phase of the intrusion, the attackers are now in a position to achieve their objectives. Objectives can include data theft, compromising data integrity, destroying data and infrastructure, disrupting operations, and perpetrating attacks on other victims.

Control Objective – Detect

The objective of the *Detec*t control in the *Actions* phase is to "discover or discern the existence, presence, or fact of an intrusion into information systems." **

Control Names	Descriptions
Amazon GuardDuty (ID: Sec.Det.1)	This control detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.
Amazon Detective (ID: Sec.Det.11)	Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspiciou s activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that

Control Names	Descriptions
	enables you to easily conduct faster and more efficient security investigations.
Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third Parties (ID: Sec.Det.6)	These controls help you to monitor, detect, visualize, receive notifications, and respond to changes in your AWS resources.
AWS Security Hub (ID: Sec.Det.3)	This control gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts.
AWS Security Hub Partners (ID: Sec.Det.4)	AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.
Outbound Proxy Partners (ID: Sec.Inf.8)	Because it is an intermediary for requests, this control can detect malicious traffic before it reaches your network.
AWS EC2 Forward Proxy Servers (ID: Sec.Inf.12)	Because it is in intermediary for requests, this control can detect malicious traffic before it reaches your network.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partners Offerings – SQL Behavioral Analytics Proxies (ID: Sec.DP.4)	These controls detect unauthorized actions on SQL applications.

Control Objective – Deny

The objective of the *Deny* control in the *Actions* phase is to "prevent the adversary from accessing and using critical information, systems, and services." **

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon Cognito (ID: Sec.IAM.5)	This control provides temporary, limited-p rivilege AWS credentials to allow access to other AWS services.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.

Control Names	Descriptions
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Key Management Service (AWS KMS) + AWS CloudHSM (ID: Sec.DP.1)	These controls prevent attackers from exfiltrating clear text data that has been encrypted.
AWS KMS Key Policies (ID: Sec.DP.2)	This control implements strong access control policies for encryption keys.

Control Objective – Disrupt

The objective of the *Disrupt* control in the *Actions* phase is to "break or interrupt the flow of information." [**]

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.

Control Names	Descriptions
–Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Config Rules (ID: Sec.IR.5)	These rules are a configurable set of functions that trigger when an environment configura tion change is registered.
–Immutable Infrastructure – Short-Lived Environments (ID: Ops.2)	These controls rebuild or refresh your environments periodically to make it more difficult for an attack payload to persist.

Control Objective – Degrade

The objective of the *Degrade* control in the *Actions* phase is to "reduce the effectiveness or efficiency of adversary command and control (C2) or communications systems, and information collection efforts or means." **

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.
Load Balancing (ID: Sec.Inf.8)	With this control, before an attacker can consistently communicate with your resources , all the instances included in the load-bala nced service need to be compromised by the attack. If one or more instances has not been compromised, the load balancer switches to an unaffected instance, which degrades the attack.
Amazon EC2 – Linux, SELinux – Mandatory Access Control (ID: Sec.Inf.17)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control (ID: Sec.Inf.18)	This control is a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls.
Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC) (ID: Sec.Inf.23)	This control implements least-privilege account profiles.
AWS Partners Offerings – SQL Behavioral Analytics Proxies (ID: Sec.DP.4)	These controls detect unauthorized actions on SQL applications.

Control Objective – Deceive

The objective of the *Deceive* control in the *Actions* phase is to "cause a person to believe what is not true. MILDEC [military deception] seeks to mislead adversary decision makers by manipulating their perception of reality." **

Control Names	Descriptions
Honeypot and Honeynet Environments (ID: Sec.IR.10)	These controls help to degrade, detect, and contain attacks.

Control Objective – Contain

The objective of the *Contain* control in the *Actions* phase is "keeping something harmful under control or within limits." **

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.
AWS Organizations + Service Control Policies (SCPs) + AWS Accounts (ID: Sec.IAM.4)	These controls provide strong, least-privilege and need-to-know security principles for both users and services across a multi-acc ount structure. You can control administrators privileges in child accounts.
Amazon Virtual Private Cloud (VPC) (ID: Sec.Inf.3)	Amazon VPC can help prevent attackers from scanning network resources during reconnaissance. Amazon VPC Black Hole Routes operate as an allow list or deny list of network reachable assets, before Security Groups or NACLs.

Control Names	Descriptions
Amazon EC2 Security Groups (ID: Sec.Inf.5)	This control is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Network Access Control Lists (ID: Sec.Inf.6)	This control is a virtual Access Control List that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.
Linux cgroups, namespaces, SELinux (ID: Sec.Inf.25)	These controls enforce capability profiles, which prevent running processes from accessing files, network sockets, and other processes.
AWS Container and Abstract Services (ID: Platform.1)	These controls can help you prevent access to underlying infrastructure by your customers and threat actors, and segregate your service instances.
Hypervisor-Level Guest-to-Guest and Guest-to-Host Separation (ID: Platform.4)	This control leverages the string isolation capabilities of the AWS hypervisor.

Control Objective – Respond

The objective of the *Respond* control in the *Actions* phase is to provide "Capabilities that help to react quickly to an adversary's or others' IO attack or intrusion." [**]

Control Names	Descriptions
Third-Party Security Tools for Containers (ID: Sec.IR.14)	This control implements advanced security protection and behavioral security solutions for Containers.

Control Names	Descriptions
Third-Party Security Tools for AWS Lambda Functions (ID: Sec.IR.15)	This control implements advanced security protection and behavioral security solutions for Lambda functions.
AWS Partner Offerings – Behavioral Monitoring, Response Tools and Services (ID: Sec.Inf.29)	These controls provide insight into the threats in your environment.
Amazon GuardDuty + AWS Lambda (ID: Sec.IR.1)	These controls detect reconnaissance activities and modify security configurations to block traffic associated with an attack.
AWS Managed Services (ID: Ops.3)	AWS Managed Services monitors the overall health of your infrastructure resources, and handles the daily activities of investigating and resolving alarms or incidents.
AWS Security Hub Automated Response and Remediation (ID: Sec IR.7)	AWS Security Hub Automated Response and Remediation is an add-on solution that works with AWS Security Hub to provide a ready-to-deploy architecture and a library of automated playbooks Refer to the AWS Security Blog : How to deploy the AWS Solution for Security Hub Automated Response and Remediation

Control Objective – Restore

The objective of the *Restore* control in the *Actions* phase is to “bring information and information systems back to their original state.” **

Control Names	Descriptions
AWS Auto Scaling (ID: Sec.Inf.9)	This control adjusts capacity to maintain steady, predictable performance.
AWS Partner Offerings – File Integrity Monitoring (ID: Sec.IR.13)	This control helps to maintain the integrity of operating system and application files.
CloudFormation + Service Catalog (ID: Ops.1)	These controls help you to provision your infrastructure in an automated and secure manner. The CloudFormation template file serves as the single source of truth for your cloud environment.
AWS DR Options (ID: Ops.4)	These controls can help you rapidly recover your IT infrastructure and data.

Control Name Descriptions

This section provides detailed descriptions of the controls used in the intrusion method, in the order they are presented in the phases. Intrusion method implementations are likely to follow a category-based implementation.

For a list of these controls, ordered by category, and for recommendations on how to prioritize implementations, see the Prioritizing Control Implementations section.

Amazon GuardDuty

Amazon GuardDuty provides intelligent threat detection by collecting, analyzing, and correlating billions of events from AWS CloudTrail, Amazon VPC Flow Logs, and DNS Logs across all of your associated AWS accounts. Amazon GuardDuty cross-references those events with threat intelligence feeds from the AWS Threat Intelligence team and third parties.

Amazon GuardDuty detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known, bad IP address.

Amazon GuardDuty Partners

Amazon GuardDuty APN Partner products are a complement to Amazon GuardDuty.

Amazon Detective

Amazon Detective makes it easy to analyze, investigate, and quickly identify the root cause of potential security issues or suspicious activities. Amazon Detective automatically collects log data from your AWS resources and uses machine learning, statistical analysis, and graph theory to build a linked set of data that enables you to easily conduct faster and more efficient security investigations.

AWS security services like Amazon GuardDuty, Amazon Macie, and AWS Security Hub as well as partner security products can be used to identify potential security issues, or findings. These services are really helpful in alerting you when something is wrong and pointing out where to go to fix it. But sometimes there might be a security finding where you need to dig a lot deeper and analyze more information to isolate the root cause and take action. Determining the root cause of security findings can be a complex process that often involves collecting and combining logs from many separate data sources, using extract, transform, and load (ETL) tools or custom scripting to organize the data, and then security analysts having to analyze the data and conduct lengthy investigations.

Amazon Detective simplifies this process by enabling your security teams to easily investigate and quickly get to the root cause of a finding.

Bottlerocket

Bottlerocket is a Linux-based open-source operating system that is purpose-built by Amazon Web Services for running containers on virtual machines or bare metal hosts. Most customers today run containerized applications on general-purpose operating systems that are updated package-by-package, which makes OS updates difficult to automate. Updates to Bottlerocket are applied in a single step rather than package-by-package. This single-step update process helps reduce management overhead by making OS updates easy to automate using container orchestration services such as Amazon EKS and Amazon ECS. The single-step updates also improve uptime for container applications by minimizing update failures and enabling easy update rollbacks.

Additionally, Bottlerocket includes only the essential software to run containers, which improves resource usage and reduces the attack surface.

AWS WAF, WAF Managed Rules + Automation

AWS WAF is a web application firewall that helps protect your web applications from common web exploits that could affect application availability, compromise security, or consume excessive resources. AWS WAF gives you control over which traffic to allow or block to your web applications by defining customizable web security rules. You can use AWS WAF to create custom rules that block common attack patterns, such as SQL injection or cross-site scripting, and rules that are designed for your specific application.

For more information, see AWS WAF Security Automations Protection Capabilities.

Scanners and Probes

Malicious sources scan and probe Internet-facing web applications for vulnerabilities. They send a series of requests that generate HTTP 4xx error codes, and you can use this history to help identify and block malicious source IP addresses. This solution creates an AWS Lambda function that automatically parses Amazon CloudFront or Application Load Balancer access logs, counts the number of bad requests from unique source IP addresses, and updates AWS WAF to block further scans from those addresses.

Known Attacker Origins (IP Reputation Lists)

A number of organizations maintain *IP address reputation lists*, which are lists of IP addresses operated by known attackers, such as spammers, malware distributors, and botnets. These services leverage the information in these reputation lists to help you block requests from malicious IP addresses.

Bots and Scrapers

Operators of publicly accessible web applications have to trust that the clients accessing their content identify themselves accurately, and that they will use services as intended. However, some automated clients, such as content scrapers or bad bots, misrepresent themselves to bypass restrictions. These services help you identify and block bad bots and scrapers.

AWS WAF Security Automations

Once deployed, AWS WAF protects your Amazon CloudFront distributions or Application Load Balancers by inspecting web requests. You can use AWS WAF to create custom, application-specific

rules that block attack patterns to ensure application availability, secure resources, and prevent excessive resource consumption.

For more information, see AWS WAF Security Automations.

Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third Parties

In addition to Amazon CloudWatch, CloudWatch Logs, AWS CloudTrail, and reporting tools such as Amazon OpenSearch Service, and Amazon QuickSight, you can integrate with third-party tools such as Splunk, Trend Micro, and Alertlogic.

You can use Amazon CloudWatch Logs to monitor, store, and access your log files from Amazon Elastic Compute Cloud (Amazon EC2) instances, AWS CloudTrail, Amazon Route 53, and other sources. You can then retrieve the associated log data from CloudWatch Logs.

Amazon CloudWatch Logs Insights enable you to interactively search and analyze your log data in Amazon CloudWatch Logs. You can run queries to help you quickly and effectively respond to operational issues. If an issue occurs, you can use Amazon CloudWatch Logs Insights to identify potential causes and validate deployed fixes.

When you create your AWS account, AWS CloudTrail is automatically enabled. When activity occurs in your AWS account, that activity is recorded in a CloudTrail event. You can easily view events in the CloudTrail console in the event history. From the event history, you can view, search, and download the past 90 days of activity in your AWS account. You can also create a CloudTrail trail to archive, analyze, and respond to changes in your AWS resources.

Amazon CloudWatch Logs + Amazon Lookout for Metrics

Amazon Lookout for Metrics uses machine learning (ML) to automatically detect and diagnose anomalies (i.e. outliers from the norm) in business and operational time series data, such as a sudden dip in sales revenue or customer acquisition rates. In a couple of clicks, you can connect Amazon Lookout for Metrics to popular data stores like Amazon S3, Amazon Redshift, and Amazon Relational Database Service (RDS), as well as third-party SaaS applications, such as Salesforce, Servicenow, Zendesk, and Marketo, and start monitoring metrics that are important to your business.

You can use Amazon CloudWatch metrics as a data source for an Amazon Lookout for Metrics detector. For details, see the Amazon Lookout for Metrics Developer Guide.

Amazon CloudWatch Logs + Amazon Lookout for Metrics + Lambda

After Amazon Lookout for Metrics creates an anomaly detection model, or detector, you can attach alerts to it using supported output connectors such as Amazon Simple Notification Service (Amazon SNS), AWS Lambda functions, Datadog, PagerDuty, Webhooks, and Slack. You can create custom alerts to notify you when Amazon Lookout for Metrics detects an anomaly of a specified severity level. For more details, see Preview: Amazon Lookout for Metrics, an Anomaly Detection Service for Monitoring the Health of Your Business.

AWS Security Hub

AWS Security Hub gives you a comprehensive view of your high-priority security alerts and compliance status across AWS accounts. With Security Hub, you now have a single place that aggregates, organizes, and prioritizes your security alerts, or findings, from multiple AWS services, such as Amazon GuardDuty, Amazon Inspector, and Amazon Macie, as well as from AWS Partner offerings.

A Security Hub *insight* is a collection of related findings defined by an aggregation statement and optional filters. An insight identifies a security area that requires attention and intervention. Security Hub offers several managed (default) insights that you cannot modify or delete. You can also create custom insights to track security issues that are unique to your AWS environment and usage.

AWS Security Hub Automated Response and Remediation

The AWS Solutions Implementation AWS Security Hub Automated Response and Remediation addresses this challenge by providing predefined response and remediation actions based on industry compliance standards and best practices.

AWS Security Hub Automated Response and Remediation is an add-on solution that works with AWS Security Hub to provide a ready-to-deploy architecture and a library of automated playbooks. The solution makes it easier for AWS Security Hub customers to resolve common security findings and to improve their security posture in AWS.

AWS Security Hub Partners

AWS Security Hub APN Partner products are a complement to Amazon GuardDuty.

Amazon Virtual Private Cloud (Amazon VPC)

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

Amazon VPC provides advanced security features, such as security groups and network access control lists, to enable inbound and outbound filtering at the instance level and subnet level. In addition, you can store data in Amazon S3 and restrict access so that it's only accessible from instances in your Amazon VPC. Optionally, you can choose to launch Dedicated Instances, which run on hardware dedicated to a single customer for additional isolation.

In this context, Amazon VPC can help prevent attackers from scanning network resources during reconnaissance.

NAT Gateways

You can use a network address translation (NAT) gateway to enable instances in a private subnet to connect to the internet or other AWS services, but prevent the internet from initiating a connection with those instances.

Amazon VPC Black Hole Routes

You can use Amazon VPC black hole routes as an allow list or deny list of network reachable assets before Security Groups or NACLs.

The state of a route appears in the route table (active | black hole). When the state is *black hole*, the route's target isn't available. For example, the specified gateway isn't attached to the VPC, or the specified NAT instance has been terminated.

For more information, see describe-route-tables in the *AWS CLI Command Reference*.

VPC Endpoints

A VPC endpoint enables you to privately connect your VPC to supported AWS services and VPC endpoint services powered by PrivateLink, without requiring an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC do not require public IP addresses to communicate with resources in the service. Traffic between your VPC and other services does not leave the Amazon network.

AWS PrivateLink

AWS PrivateLink is a purpose-built technology designed for customers to access AWS services in a highly available and scalable manner, while keeping all the network traffic within the AWS network. When you create endpoints for AWS services powered by PrivateLink, these service endpoints appear as Elastic Network Interface (ENI) with private IP addresses in your VPCs. PrivateLink makes it unnecessary to allow list public IP addresses, or manage Internet connectivity using an Internet Gateway, Network Address Translation (NAT) devices, or firewall proxies to connect to AWS services. AWS services available on PrivateLink also support private connectivity over AWS Direct Connect, so applications in your own data centers can connect to AWS services through the Amazon private network using the service endpoints.

Amazon EC2 Security Groups

A security group is a virtual firewall that controls inbound and outbound traffic to your network resources and Amazon EC2 instance.

Network Access Control Lists

Similar to a firewall, Network Access Control Lists (NACLs) control traffic in and out of one or more subnets. To add an additional layer of security to your Amazon VPC, you can set up NACLs with rules similar to your security groups.

AWS Identity and Access Management + AWS Organizations

AWS Identity and Access Management (IAM) enables you to securely manage access to AWS services and resources. Using IAM, you can create and manage AWS users and groups, and specify permissions to allow and deny their access to AWS resources.

AWS Certificate Manager + Transport Layer Security

Protecting data in transit denies attackers the ability to capture data in transit during Reconnaissance, unless they are able to impersonate a legitimate endpoint.

AWS Certificate Manager is a service that enables you to easily provision, manage, and deploy public and private Transport Layer Security (TLS) certificates for use with AWS services and your internal connected resources. TLS certificates are used to secure network communications and establish the identity of websites over the internet and resources on private networks.

For more information, see:

- Introducing s2n, a New Open Source TLS Implementation
- Easier Certificate Validation Using DNS with AWS Certificate Manager
- Maintaining Transport Layer Security All the Way to Your Container: Using the Network Load Balancer with Amazon ECS
- Encrypting Data in Transit

Network Infrastructure Solutions in the AWS Marketplace

The infrastructure solutions in the AWS Marketplace can help deny attackers access to your data and infrastructure as they conduct reconnaissance.

For more information, see Network Infrastructure Software on the AWS Marketplace.

Amazon Virtual Private Cloud VPN Gateway + AWS Direct Connect

An Amazon Virtual Private Cloud (Amazon VPC) VPN gateway connection creates a link between your data center (or network) and your Amazon VPC . A customer gateway is the anchor on your side of that connection, and can be a physical or software appliance. The anchor on the AWS side of the VPN connection is known as a virtual private gateway.

For more information, see https://docs.aws.amazon.com/vpn/index.html.

You can use AWS Direct Connect to establish a private virtual interface from your on-premises network directly to your Amazon VPC. This interface provides you with a private, high-bandwidth network connection between your network and your Amazon VPC. With multiple virtual interfaces, you can establish private connectivity to multiple VPCs, while maintaining network isolation.

Amazon GuardDuty + AWS Lambda

Amazon GuardDuty gives you intelligent threat detection by collecting, analyzing, and correlating billions of events from AWS CloudTrail, Amazon VPC Flow Logs, and DNS Logs across all of your associated AWS accounts, and then cross-references them with threat intelligence feeds from the AWS Threat Intelligence team and third-party information.

Amazon GuardDuty detects reconnaissance activity, such as unusual API activity, intra-VPC port scanning, unusual patterns of failed login requests, or unblocked port probing from a known bad IP address.

To learn more about the types of findings that Amazon GuardDuty can identify, see Finding types.

AWS Lambda is a compute service that lets you run code without provisioning or managing servers. You can use AWS Lambda to run your code in response to events, such as when Amazon GuardDuty finds that a CloudWatch Event was triggered.

For example, the following blog post describes how to use AWS GuardDuty and AWS Web Application Firewall to automatically block suspicious hosts by triggering AWS Lambda responders: How to use Amazon GuardDuty and AWS Web Application Firewall to automatically block suspicious hosts.

When an attack is detected by Amazon GuardDuty, AWS Lambda responders can be used to modify security configurations to block traffic associated with an attack in progress as well as isolate the potentially compromised environment.

This same approach, AWS GuardDuty > Amazon CloudWatch Events > AWS Lambda responders, can be used to disrupt other flows.

Honeypot and Honeynet Environments

Deception technology products present themselves as part of a legitimate infrastructure. When an attacker attempts to compromise the infrastructure, the deception technology helps to degrade, detect, and contain the attack, so your system recovers from attacks faster.

Deception technology products for honeypot or honeynet environments include third-party deception technology solutions, both commercial and open-source. Solutions include Guardicore, Attivo, Illusive, TopSpin, or TrapX. There are also numerous open-source solution honeypot and honeynet projects on GitHub and elsewhere.

Honeywords and Honeykeys

Planting false credentials makes attackers think they have something of value when they do not. When an attacker attempts to use stolen, false credentials, it helps your system to detect and contain the attacker, so your system recovers faster. The detected use of honeykeys and honeywords is always a true positive for intrusion attempts.

AWS WAF + AWS Lambda

AWS WAF is a web application firewall that helps protect your web applications from common web exploits that could affect application availability, compromise security, or consume excessive

resources. AWS WAF gives you control over which traffic to allow or block to your web applications by defining customizable web security rules. You can use AWS WAF to create custom rules that block common attack patterns, such as SQL injection or cross-site scripting, and rules that are designed for your specific application.

For more information, see AWS WAF Security Automations Architecture Overview.

Honeypot (A) for Bad Bots and Scrapers

This component automatically sets up a honeypot, which is a security mechanism intended to lure and deflect an attempted attack. The honeypot is a trap endpoint that you can insert in your website to detect inbound requests from content scrapers and bad bots. If a source gets access to the honeypot, the Access Handler AWS Lambda function intercepts and inspects the request to extract its IP address, and then add it to an AWS WAF block list.

Amazon CloudWatch Events & Alarms + Amazon SNS + SIEM Solutions

You can combine Amazon CloudWatch Events & Alarms with Amazon Simple Notification Service (SNS) and integrate them with security information and event management (SIEM) solutions like Splunk or AlertLogic.

Amazon CloudWatch Events delivers a near real-time stream of system events that describe changes in Amazon Web Services (AWS) resources. Using simple rules that you can quickly set up, you can match events and route them to one or more target functions or streams.

You can create a CloudWatch alarm that watches a single CloudWatch metric or the result of a math expression based on CloudWatch metrics. The alarm performs one or more actions based on the value of the metric or expression relative to a threshold over a number of time periods. The action can be a notification sent to an Amazon SNS topic.

You can also add alarms to CloudWatch dashboards and monitor them visually or integrate with other SIEM solutions.

For more information, see:

- Creating custom responses to GuardDuty findings with Amazon CloudWatch Events
- Using Amazon CloudWatch Alarms

Network Infrastructure Solutions in AWS Marketplace

The network infrastructure solutions that are available in the AWS Marketplace can help you deny access to the attackers attempting get your data and infiltrate your infrastructure as they conduct reconnaissance.

For more information, see Network Infrastructure Software on the AWS Marketplace.

Amazon Cognito

Amazon Cognito provides solutions to control access to backend resources from your application. You can define roles and assign users to different roles so your application can access only the resources that are authorized for each user.

Amazon Cognito Identity Pools (Federated Identities)

Amazon Cognito identity pools (federated identities) enable you to create unique identities for your users and federate them with identity providers. With an identity pool, you can obtain temporary, limited-privilege AWS credentials to access other AWS services.

Reverse Proxy Architecture

Reverse proxies are a powerful software architecture primitive for fetching resources from a server on behalf of a client. They serve a number of purposes, from protecting servers from unwanted traffic to offloading some of the heavy lifting of HTTP traffic processing.

For more information, see NGINX reverse proxy sidecar for a web container hosted with Amazon ECS and AWS Fargate.

Amazon Virtual Private Cloud + Automation

With Amazon Virtual Private Cloud (Amazon VPC) Subnet Isolation, you can contain compromised systems by using AWS Command Line Interface (AWS CLI), or software development kits using predefined, restrictive security groups. You can save the current security group of the host or instance, and then isolate the host using restrictive ingress and egress security group rules.

For more information, see AWS Security Incident Response Guide.

AWS Shield

AWS Shield is a managed Distributed Denial of Service (DDoS) protection service that safeguards applications running on AWS. AWS Shield provides always-on detection and automatic, inline mitigations that minimize application downtime and latency, so you don't have to engage AWS Support to benefit from DDoS protection. There are two tiers of AWS Shield: Standard and Advanced.

All AWS customers benefit from the automatic protections of AWS Shield Standard, at no additional charge. AWS Shield Standard defends against most common, frequently occurring network and transport layer DDoS attacks that target your website or applications. When you use AWS Shield Standard with Amazon CloudFront and Amazon Route 53, you receive comprehensive availability protection against all known infrastructure (Layer 3 and 4) attacks.

Amazon VPC Flow Logs + Amazon CloudWatch Alarms

Amazon VPC Flow Logs enables you to capture information about the IP traffic going to and from network interfaces in your Amazon VPC. Flow log data is stored using Amazon CloudWatch Logs. After you've created a flow log, you can view and retrieve its data in Amazon CloudWatch Logs and Amazon S3, or another analytics tool. You can also use flow logs as a security tool to monitor the traffic that is reaching your instance.

For more information, see Publishing flow logs to CloudWatch Logs.

AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries

AWS Identity and Access Management (IAM) enables you to manage access to AWS services and resources securely. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

You manage access in AWS by creating policies and attaching them to IAM identities (users, groups of users, or roles) or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions.

A permissions boundary is a managed policy in which you set the maximum permissions that an identity-based policy can grant to an IAM entity. When you set a permissions boundary for an entity, the entity can perform only the actions that are allowed by both its identity-based policies and its permissions boundaries.

AWS Identity and Access Management (IAM) Roles

AWS Identity and Access Management (IAM) Roles help deny or contain the blast radius of attacks. For example, you can use an IAM role to grant permissions to applications running on Amazon EC2 instances.

For more information, see Using an IAM role to grant permissions to applications running on Amazon EC2 instances.

Role-based access control (RBAC) and attribute-based access control (ABAC) are authorization strategies that provide flexibility in granting permissions to resources.

AWS Organizations + Service Control Policies (SCPs) + AWS Accounts

With AWS Organizations, you can create Service Control Policies (SCPs) that centrally control AWS service use across multiple AWS accounts. SCPs put bounds around the permissions that AWS Identity and Access Management (IAM) policies can grant to entities in an account, such as IAM users and roles. For example, IAM policies for an account in your organization cannot grant access to AWS Direct Connect if access is not also allowed by the SCP for the account. Entities can only use the services allowed by both the SCP and the IAM policy for the account.

Service control policies (SCPs) are one type of policy that you can use to manage your organization. SCPs enable you to restrict, at the account level of granularity, what services and actions are available to the users, groups, and roles in those accounts.

Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies

A bucket policy is a resource-based AWS Identity and Access Management (IAM) policy. You add a bucket policy to a bucket to grant other AWS accounts or IAM users access permissions for the bucket and the objects in it.

With an Amazon S3 bucket policy, malicious objects that contain attacker payload can be prevented from directly being uploaded into the bucket. All objects must be uploaded through the application and subject to malware checks before they can be stored in an Amazon S3 bucket.

For more information, see Managing access to resources.

Amazon EC2 – Linux, SELinux – Mandatory Access Control

Mandatory Access Control can be configured as a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls for users (including root) and processes.

Amazon EC2 – FreeBSD, Trusted BSD – Mandatory Access Control

Mandatory Access Control for FreeBSD and Trusted BSD can be configured as a system policy that cannot be overridden, which mediates access to files, devices, sockets, other processes, and API calls for users (including root) and processes.

Amazon EC2 – Linux, FreeBSD – Hardening and Minimization

Hardening and minimization make it difficult to exploit a vulnerability in a service by reducing the services that are running and removing or uninstalling unnecessary services.

For example, applying carefully considered sets of recommendations from CIS Amazon Linux 2 Benchmark - Level 2, and testing possible impacts on applications can reduce the attack surface of these instances running in Amazon EC2.

Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC)

RBAC and DAC provide least-privilege account profiles mediated by root that control which users can get access to your resources.

Amazon EC2 – Windows – Device Guard

With Windows Device Guard for your Amazon EC2 instance, you can specify which binaries are authorized to run on your server, including user mode and kernel mode binaries, which enhances AppLocker functionality.

Microsoft Windows Security Baselines

A security baseline is a group of Microsoft-recommended configuration settings. You can use security baselines to:

- Set configuration settings. For example, you can use Group Policy to configure a device with the setting values specified in the baseline.

- Make sure that user and device configuration settings are compliant with the baseline.

For more information, see Windows security baselines on the Microsoft Documentation site.

AWS Physical & Operational Security Policies & Processes

Amazon Web Services is responsible for protecting the global infrastructure that runs all of the services offered in the AWS Cloud. This infrastructure comprises the hardware, software, networking, and facilities that run AWS services. Protecting this infrastructure is AWS's number one priority, and while you can't visit our data centers or offices to see this protection firsthand, we provide several reports from third-party auditors who have verified our compliance with a variety of computer security standards and regulations.

For more information, see AWS Compliance, AWS Overview of Security Processes, and Data Center Controls.

Immutable Infrastructure – Short-Lived Environments

In the case of Amazon EC2, Containers, and especially AWS Lambda, short environment lifetimes (when compared to traditional datacenters) mean that an environment being targeted at time t="now" may not be the same as an environment being targeted in time t="now+5 minutes". When environments are being rebuilt or refreshed every few minutes, it is a much more difficult task to make an attack payload persist.

When your production environment does not have privileges configured, including for the network infrastructure, there is nothing for attackers to modify and your environment is more resilient to attacks.

Load Balancing

With load balancing, before an attacker can consistently get access to your resources, all the instances included in the load-balanced service need to be compromised by the attack. If one or more instances has not been compromised, the load balancer switches to an unaffected instance, which degrades the attack.

For more information, see Elastic Load Balancing.

AWS Lambda, Amazon Simple Queue Service (Amazon SQS), AWS Step Functions

These services are orchestration mechanisms for containment.

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume—there is no charge when your code is not running. With AWS Lambda, you can run code for virtually any type of application or backend service, all with zero administration. Just upload your code and AWS Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app.

Amazon Simple Queue Service (Amazon SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. Amazon SQS eliminates the complexity and overhead associated with managing and operating message-oriented middleware, and empowers developers to focus on differentiating work. Using Amazon SQS, you can send, store, and receive messages between software components at any volume, without losing messages or requiring other services to be available. Get started with Amazon SQS in minutes using the AWS console, Command Line Interface or SDK of your choice, and three simple commands.

AWS Step Functions lets you coordinate multiple AWS services into serverless workflows so you can build and update apps quickly. Using AWS Step Functions, you can design and run workflows that stitch together services such as AWS Lambda and Amazon ECS into feature-rich applications. Workflows are made up of a series of steps: the output of one step is the input for the next step.

AWS WAF

AWS WAF is a web application firewall that helps protect your web applications from common web exploits that could affect application availability, compromise security, or consume excessive resources.

AWS container and abstract services

AWS container and abstract services prevent access to underlying infrastructure by both customers and threat actors, and segregate service instances while enabling customers to apply selective permissions to allow third parties to access them. This limits the scope and effect of any attack a threat actor could perpetrate using these services.

Linux cgroups, namespaces, SELinux

SELinux and the technologies that support it (including cgroups and namespaces) can enforce capability profiles that prevent running processes from accessing files, network sockets, and other processes, using Mandatory Access Control. This means that, even if a running process is compromised, it can be prevented from accessing other resources on the OS instance, or even doing things expected of regular Unix processes (such as forking copies of itself or reading world-read files).

Hypervisor-Level Guest-to-Guest and Guest-to-Host Separation

The hypervisor for Amazon EC2 instances and other services is in-scope for PCI-DSS certification for separating different operating system instances from each other, and guest operating systems from itself. Hypervisor separation, and the additional technologies incorporated as modules into it (such as Security Groups), provide mechanisms to mitigate some risks of one compromised Amazon EC2 instance being used as a platform to compromise other instances.

AWS Systems Manager State Manager

AWS Systems Manager State Manager helps you define and maintain consistent OS configurations such as firewall settings and anti-malware definitions to comply with your policies. You can monitor the configuration of a large set of instances, specify a configuration policy for the instances, and automatically apply updates or configuration changes.

AWS Partner Network (APN) Offerings – File Integrity Monitoring

File integrity monitoring (FIM) and enforcement are controls that help maintain the integrity of operating system files and application files by verifying the current file state and a known good baseline of these files. Check features of products carefully for enforcement or reversion capabilities. For information about AWS Marketplace solutions, see File Integrity Monitoring on the AWS Marketplace.

Third-Party WAF Integrations

Some examples of third-party tools that integrate with AWS WAF include Trend Micro, Imperva, and Alert Logic.

For more information about third-party integrations with AWS WAF, see AWS WAF Partners.

AWS Config

AWS Config enables you to assess, audit, and evaluate the configurations of your AWS resources. AWS Config continuously monitors and records your AWS resource configurations and allows you to automate the evaluation of recorded configurations against desired configurations. With AWS Config, you can review changes in configurations and relationships between AWS resources, dive into detailed resource configuration histories, and determine your overall compliance against the configurations specified in your internal guidelines. This enables you to simplify compliance auditing, security analysis, change management, and operational troubleshooting.

AWS Config rules

AWS Config rules are a configurable and extensible set of Lambda functions (for which source code is available) that trigger when an environment configuration change is registered by the AWS Config service. If AWS Config rules deem a configuration change to be undesirable, they can act to remediate it. In common with all other Lambda functions, AWS Config rules can be assigned IAM Roles with permissions that enable them to make appropriate remedial API calls.

Amazon CloudWatch Events + Lambda

Amazon CloudWatch Events occur when various states are detected (such as GuardDuty findings), and can be used to trigger Lambda functions in the same manner as AWS Config rules. If the Lambda functions deem an event to be undesirable, they can act to remediate it, using IAM Roles with the correct permissions to allow them to make appropriate remedial API calls.

AWS Managed Services

AWS Managed Services monitors the overall health of your infrastructure resources, and handles the daily activities of investigating and resolving alarms or incidents. AWS Managed Services protects your information assets and helps keep your AWS infrastructure secure. With anti-malware protection, intrusion detection, and intrusion prevention systems, AWS Managed Services manages security policies per stack, and is able to quickly recognize and respond to any intrusion.

CloudFormation + Service Catalog

AWS CloudFormation provides a common language for you to describe and provision all the infrastructure resources in your cloud environment. CloudFormation enables you to use a simple text file to model and provision, in an automated and secure manner, all the resources needed for

your applications across all regions and accounts. This file serves as the single source of truth for your cloud environment.

Service Catalogallows organizations to create and manage catalogs of IT services that are approved for use on AWS. These IT services can include everything from virtual machine images, servers, software, and databases to complete multi-tier application architectures. Service Catalog allows you to centrally manage commonly deployed IT services, and helps you achieve consistent governance and meet your compliance requirements, while enabling users to quickly deploy only the approved IT services they need.

AWS Systems Manager State Manager, or Third-Party or OSS File Integrity Monitoring Solutions on Amazon EC2

AWS Systems Manager State Manager is a secure and scalable configuration management service that automates the process of keeping your Amazon EC2 and hybrid infrastructure in a state that you define.

File integrity monitoring (FIM) and enforcement are controls that help maintain the integrity of operating system files and application files by verifying the current file state and a known good baseline of these files. Check features of products carefully for enforcement or reversion capabilities. For information about AWS Marketplace solutions, see File Integrity Monitoring on the AWS Marketplace.

Third-Party Security Tools for Containers

There are several third-party security tools available for containers offered by AWS Partners on the AWS Marketplace. AWS Container Competency Partners help AWS customers better run their container workloads on AWS. These solutions extend our AWS container services by providing additional security, monitoring, and management capabilities.

Third-Party Security Tools for AWS Lambda Functions

Two of the third-party security tools available for Lambda functions are offered by Check Point Software Technologies Ltd. and Palo Alto Networks.

Please check the AWS Marketplace for a full, up-to-date list of AWS Partner offerings.

AWS Partner Offerings – Behavioral Monitoring, Response Tools and Services

AWS Partner offerings, such as Alert Logic and Trend Micro, provide insight into the real threats in your environments.

AWS Partner Offerings – Anti-Malware Protection

AWS Technology Partner anti-malware protection offerings help to detect and block malicious payloads.

AWS Lambda Partners

AWS Lambda Partners provide services and tools that help customers build or migrate their solutions to a microservices based serverless architecture, without having to worry about provisioning or managing servers.

AWS Container Partners – Security

AWS Container Competency Partners have a technology product or solution on AWS that offers support to run workloads on containers. The product or solution integrates with AWS services in a way that improves the AWS customer's ability to run workloads using containers on AWS.

AWS IoT Device Defender + AWS IoT SiteWise

AWS IoT Device Defender is a fully managed service that helps you secure your fleet of IoT devices. AWS IoT Device Defender continuously audits your IoT configurations to make sure that they aren't deviating from security best practices.

AWS IoT SiteWise is a managed service that makes it easy to collect, store, organize, and monitor data from industrial equipment at scale to help you make better, data-driven decisions.

For more information, see Configuration and vulnerability analysis in AWS IoT SiteWise.

AWS Secrets Manager

AWS Secrets Manager helps you protect secrets needed to access your applications, services, and IT resources. The service enables you to easily rotate, manage, and retrieve database credentials, API

keys, and other secrets throughout their lifecycle. Users and applications retrieve secrets with a call to Secrets Manager APIs, eliminating the need to hardcode sensitive information in plain text.

Amazon EC2 – Linux, Windows, FreeBSD – Address Space Layout Randomization (ASLR)

Address space layout randomization (ASLR) is a technology that prevents shellcode from being successful. It does this by randomly offsetting the location of modules and certain in-memory structures. Although this can help prevent the exploitation of vulnerabilities, there are limits to how effective it is. Processors, and operating systems need to provide ASLR support, and on some operating systems, applications must opt in.

Amazon EC2 – Linux, Windows, FreeBSD – Data Execution Prevention (DEP)

Data execution prevention (DEP) is a memory safety feature that makes it more difficult for malware to run. It prevents certain memory blocks, such as the stack, from being executed.

For more information about DEP, see Data Execution Prevention on Microsoft's Documentation site.

Amazon EC2 – Windows – User Account Control (UAC)

UACs, also known as least-privilege user account, make it more difficult for malware to install and run.

For more information about UACs, see How User Account Control works on the Microsoft Documentation website.

Amazon Simple Email Service Spam and Virus Protection

Amazon Simple Email Service (Amazon SES) uses a number of spam and virus protection measures. It uses block lists to prevent mail from known spammers from entering the system. It also performs virus scans on every incoming email message that contains an attachment. Amazon SES makes its spam detection verdicts available to you, so you can decide if you trust each message. In addition to the spam and virus verdicts, Amazon SES provides the DKIM and SPF check results.

Amazon SES uses in-house content filtering technologies to scan email content for spam and malware. In exceptional cases, accounts identified as sending spam or other low-quality email might be suspended, or Amazon SES may take such other action as it deems appropriate. When malware is detected, Amazon SES prevents these emails from being sent.

For more information, see Amazon SES FAQs.

AWS Auto Scaling

AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost.

If a *Respond* control automatically kills an instance of an operating environment (such as an Amazon EC2 instance, container, or Lambda function), AWS Auto Scaling creates new instances from reference images to replace it in line with load requirements.

AWS Systems Manager State Manager, AWS Systems Manager Inventory, AWS Config

AWS Config continuously monitors and records your AWS resource configurations and allows you to automate the evaluation of recorded configurations against your specified configurations.

When attackers create new AWS assets, or if malware is installed with a regular package, the AWS Systems Manager Inventory identifies it and sends it to AWS Config for evaluation.

Amazon EC2 Forward Proxy Servers

A forward proxy server is an intermediary for requests from internal users and servers, often caching content to speed up subsequent requests. Companies usually implement proxy solutions to provide URL and web content filtering, IDS/IPS, data loss prevention, monitoring, and advanced threat protection.

Outbound Proxy Partners

Outbound proxy partner products such as Sophos UTM provide multiple security functions, including firewall, intrusion prevention, VPN, and web filtering. Sophos Outbound Gateway provides a distributed, fault-tolerant architecture to provide visibility, policy enforcement, and elastic scalability to outbound web traffic.

Amazon GuardDuty + AWS Lambda + AWS WAF, Security Groups, NACLs

Amazon GuardDuty gives you intelligent threat detection by collecting, analyzing, and correlating billions of events from AWS CloudTrail, Amazon VPC Flow Logs, and DNS Logs across all of your

associated AWS accounts. It then cross-references them with threat intelligence feeds from AWS's Threat Intelligence team and third-party feeds.

When an attack is detected by Amazon GuardDuty, AWS Lambda responders can be used to modify security configurations to block traffic associated with an attack in progress as well as isolate the potentially-compromised environment.

For more information, see How to use Amazon GuardDuty and AWS Web Application Firewall to automatically block suspicious hosts

AWS DR Options

The AWS Cloud supports many popular disaster recovery (DR) architectures, from *pilot light* environments that might be suitable for small customer workload data center failures, to *hot standby* environments that enable rapid failover at scale. With data centers in regions all around the world, AWS provides a set of cloud-based disaster recovery services designed to provide rapid recovery of your IT infrastructure and data.

For more information about available disaster recover technology, see CloudEndure Disaster Recovery and Disaster Recovery of Workloads on AWS.

AWS Partners Offerings – SQL Behavioral Analytics Proxies

Third-party behavioral analytics proxies for SQL, such as SecuPi, can detect unauthorized actions on SQL applications and act to constrain access when there is unexpected behavior.

AWS Nitro Enclaves

AWS Nitro Enclaves enables customers to create isolated compute environments to further protect and securely process highly sensitive data such as personally identifiable information (PII), healthcare, financial, and intellectual property data within their Amazon EC2 instances. Nitro Enclaves uses the same Nitro Hypervisor technology that provides CPU and memory isolation for EC2 instances.

AWS Key Management Service (AWS KMS) + AWS CloudHSM

AWS Key Management Service (AWS KMS)and AWS CloudHSM can prevent attackers from exfiltrating clear text data that has been encrypted, as well as crypto key material used to encrypt data.

AWS Key Management Service (AWS KMS) makes it easy to manage encryption keys used to encrypt data stored by your applications regardless of where you store it.

AWS CloudHSM is a cloud-based hardware security module (HSM) that enables you to easily generate and use your own encryption keys on the AWS Cloud.

For more information, see AWS CloudHSM.

AWS KMS Key Policies

The primary method to manage access to your AWS KMS keys (formerly CMKs) is with policies. Policies are documents that describe who has access to what. Policies attached to an IAM identity are identity-based policies (or IAM polices), and policies attached to other kinds of resources are resource-based policies. In AWS KMS, you must attach resource-based policies to your AWS KMS keys. These are key policies. All KMS CMKs have a key policy.

AWS Network Firewall

AWS Network Firewall is a stateful, managed, network firewall and intrusion detection and prevention service for your virtual private cloud (VPC). Its flexible rules engine lets you define firewall rules that give you fine-grained control over network traffic, such as blocking outbound Server Message Block (SMB) requests to prevent the spread of malicious activity.

AWS Network Firewall includes features that provide protections from common network threats. Its intrusion prevention system (IPS) provides active traffic flow inspection so you can identify and block vulnerability exploits using signature-based detection. AWS Network Firewall also offers web filtering that can stop traffic to known bad URLs and monitor fully qualified domain names.

Prioritizing Control Implementations

Organizations often want to know where to start if they want to implement a classic intrusion analysis framework. This section describes two ways that you can use the control number associated with each control listed in the Appendix: Reference Material section to prioritize control implementations. Control numbers can be aligned with the AWS Cloud Adoption Framework (AWS CAF) or can be used to prioritize implementations based on control coverage. Each of these approaches are discussed.

Each unique control included in the *Appendix: Reference Material* section has a unique control number assigned to it. For example, the following example table, the control number is *Sec.IAM.2*.

Control number example

Control Names	Descriptions
AWS Identity and Access Management (IAM) + IAM Policies and Policies Boundaries (ID: Sec.IAM.2)	These controls provide strong, least-privilege and need-to-know security principles for both the users and services that can access your resources.

The same control number appears in each place in the intrusion method analysis framework that the associated control is used. For example, each appearance of the *Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies* control in the method analysis framework includes the Sec.DP.6 control number. The control numbers are based on the AWS CAF. The guidance and current recommendations provided by the AWS CAF help you build a comprehensive approach to cloud computing across your organization, and throughout your IT lifecycle. Using the AWS CAF helps you realize measurable business benefits from cloud adoption faster and with less risk.

The AWS CAF organizes guidance into six areas of focus, known as *perspectives*. Each perspective covers distinct responsibilities owned or managed by functionally related stakeholders. In general, the Business, People, and Governance Perspectives focus on business capabilities, while the Platform, Security, and Operations Perspectives focus on technical capabilities.

AWS CAF perspectives

For a full explanation of the AWS CAF, see AWS Cloud Adoption Framework.

Numerous controls listed in *this paper* are from the AWS CAF Security perspective. To help you with your implementation, you can use the AWS CAF Security Epics. The Security Epics consist of groups of user stories (use cases and abuse cases) that you can work on during sprints. Each of these epics has multiple iterations that address increasingly complex requirements and layering in robustness.

Although we advise the use of Agile methodologies, the epics can also be treated as general work streams or topics that help in prioritizing and structuring delivery using any other framework. Some CAF perspectives, such as the Operations and Platform perspectives, do not have epics.

AWS CAF Security Epics

Control Number Format

The format of the control numbers is:

<CAF perspective>.<CAF perspective epic>.<sequential_number>

The *CAF perspective epic* only applies to AWS CAF perspectives that have epics, such as the Security perspective.

Some examples of control numbers:

- Sec.IAM.1 – CAF Security Perspective, Identity & Access Management Epic, control 1
- Sec.Det.1 – CAF Security Perspective, Detective Security Epic, control 1
- Sec.DP.3 – CAF Security Perspective, Data Protection Epic, control 3
- Sec.Inf.11 – CAF Security Perspective, Infrastructure Security Epic, control 11
- Sec.IR.5 – CAF Security Perspective, Incident Response Epic, control 5
- Platform.1 – CAF Platform Perspective, control 1
- Ops.2 – CAF Operations Perspective, control 2

Prioritize Controls with the Control Number and AWS CAF

Organizations that use AWS CAF to build a comprehensive approach to cloud computing across their organization and have also decided to implement some or all of the controls described in this paper, can use the tables in this section to cross-reference their efforts. This table makes it easy to identify which intrusion method controls can be implemented as organizations perform sprints associated with AWS CAF perspectives and epics.

For example, when an organization plans to work on the *Detective Controls Epic*, the table shows them that when they implement the controls listed under that epic, they will also be enabling other capabilities as part of their intrusion analysis strategy.

This approach can help organizations prioritize which intrusion method controls to implement as part of a broader AWS CAF strategy.

Table 12 – Controls Mapped to AWS Cloud Adoption Framework (AWS CAF)

Control ID	Control Name
Security Perspective – Identity and Access Management (IAM) Epic	
Sec.IAM.1	AWS Identity and Access Management (IAM) Roles
Sec.IAM.2	AWS Identity and Access Management (IAM) + IAM Policies and Policy Boundaries
Sec.IAM.3	AWS Identity and Access Management (IAM) + AWS Organizations
Sec.IAM.4	AWS Organizations + Service Control Policies (SCPs) + AWS Accounts
Sec.IAM.5	Amazon Cognito
Security Perspective – Detective Controls Epic	
Sec.Det.1	Amazon GuardDuty
Sec.Det.2	Amazon GuardDuty Partners

Control ID	Control Name
Sec.Det.3	AWS Security Hub
Sec.Det.4	AWS Security Hub Partners
Sec.Det.5	AWS Config
Sec.Det.6	Amazon CloudWatch, CloudWatch Logs, CloudTrail + Insights, Reporting & Third Parties
Sec.Det.7	Amazon CloudWatch Events & Alarms + Amazon SNS + SIEM Solutions
Sec.Det.8	Amazon VPC Flow Logs + CloudWatch Alarms or other analytics tools
Sec.Det.9	AWS IoT Device Defender + AWS IoT SiteWise
Sec.Det.10	Amazon CloudWatch Logs + Amazon Lookout for Metrics
Sec.Det.11	Amazon Detective
Security Perspective – Infrastructure Security Epic	
Sec.Inf.1	AWS WAF
Sec.Inf.2	AWS WAF, WAF Managed Rules + Automation
Sec.Inf.3	Amazon Virtual Private Cloud (Amazon VPC)
Sec.Inf.4	AWS Direct Connect
Sec.Inf.5	Amazon EC2 Security Groups
Sec.Inf.6	Network Access Control Lists (NACLs)
Sec.Inf.7	Outbound Proxy Partners

Control ID	Control Name
Sec.Inf.8	Load Balancing
Sec.Inf.9	AWS Auto Scaling
Sec.Inf.10	Network infrastructure solutions in the AWS Marketplace
Sec.Inf.11	Reverse Proxy architecture
Sec.Inf.12	Amazon EC2 Forward Proxy Servers
Sec.Inf.13	AWS Shield
Sec.Inf.14	AWS Systems Manager State Manager
Sec.Inf.15	AWS Systems Manager State Manager, or Third-Party or OSS File Integrity Monitoring Solutions on Amazon EC2
Sec.Inf.16	AWS Systems Manager State Manager, AWS Systems Manager Inventory, AWS Config
Sec.Inf.17	Amazon EC2 – Linux, SELinux – Mandatory Access Control
Sec.Inf.18	Amazon EC2 – FreeBSD Trusted BSD – Mandatory Access Control
Sec.Inf.19	Amazon EC2 – Linux, FreeBSD – Hardening and Minimization
Sec.Inf.20	Amazon EC2 – Linux, Windows, FreeBSD – Address Space Layout Randomization (ASLR)
Sec.Inf.21	Amazon EC2 – Linux, Windows, FreeBSD – Data Execution Prevention (DEP)

Control ID	Control Name
Sec.Inf.22	Amazon EC2 – Windows – User Account Control (UAC)
Sec.Inf.23	Amazon EC2 – Linux – Role-Based Access Control (RBAC) and Discretionary Access Control (DAC)
Sec.Inf.24	Microsoft Windows Security Baselines
Sec.Inf.25	Linux cgroups, namespaces, SELinux
Sec.Inf.26	Amazon EC2 – Windows – Device Guard
Sec.Inf.27	AWS Lambda Partners
Sec.Inf.28	Container Partners – Security
Sec.Inf.29	AWS Partner Offerings – Behavioral Monitoring, Response Tools and Services
Sec.Inf.30	AWS Network Firewall
Sec.Inf.31	Amazon Simple Email Service (Amazon SES)
Sec.Inf.32	Bottlerocket
Security Perspective - Data Protection Epic	
Sec.DP.1	AWS Key Management Service (KMS) + AWS CloudHSM
Sec.DP.2	AWS KMS Key Policies
Sec.DP.3	AWS Certificate Manager + Transport Layer Security (TLS)
Sec.DP.4	AWS Partner Offerings – SQL Behavioral Analytics Proxies

Control ID	Control Name
Sec.DP.5	AWS Nitro Enclaves
Sec.DP.6	Amazon Simple Storage Service (Amazon S3) Bucket Policies, Object Policies
Sec.DP.7	AWS Secrets Manager
Security Perspective - Incident Response Epic	
Sec.IR.1	Amazon GuardDuty + AWS Lambda
Sec.IR.2	AWS WAF + AWS Lambda
Sec.IR.3	Third-Party WAF Integrations
Sec.IR.4	Amazon GuardDuty + AWS Lambda + AWS WAF, Security Groups, NACLs
Sec.IR.5	AWS Config Rules
Sec.IR.6	Amazon CloudWatch Events + Lambda
Sec.IR.7	AWS Security Hub Automated Response and Remediation
Sec.IR.8	Amazon CloudWatch Logs + Amazon Lookout for Metrics + Lambda
Sec.IR.9	Amazon Virtual Private Cloud (Amazon VPC) + automation
Sec.IR.10	Honeypot and Honeynet Environments
Sec.IR.11	Honeywords and Honeykeys
Sec.IR.12	AWS Partner Offerings – Anti-Malware Protection

Control ID	Control Name
Sec.IR.13	AWS Partner Offerings – File Integrity Monitoring
Sec.IR.14	Third-Party Security Tools for Containers
Sec.IR.15	Third-Party Security Tools for AWS Lambda Functions
Platform Perspective	
Platform.1	AWS Container and Abstract Services
Platform.2	AWS Lambda, Amazon Simple Queue Service (Amazon SQS), AWS Step Functions
Platform.3	Amazon Simple Email Service
Platform.4	Hypervisor-Level Guest-to-Guest and Guest-to-Host Separation
Platform.5	AWS physical and operational security policies and processes
Operations Perspective	
Ops.1	CloudFormation + Service Catalog
Ops.2	Immutable Infrastructure – Short-Lived Environments
Ops.3	AWS Managed Services
Ops.4	AWS DR Solutions

Prioritize Controls Based on Control Coverage

Another way to leverage the unique control numbers, is to identify which controls provide the greatest level of coverage, and potentially provided the biggest ROI.

For example, the following table shows that by implementing control *Sec.IR.15*, (Third-Party Security Tools for AWS Lambda Functions), it can potentially help detect, deny, disrupt, contain, and respond in the Exploitation phase of an attack. This mapping helps identify the benefits of enabling that one control, which provides significant Infrastructure Security capability coverage in multiple places in the intrusion method analysis framework.

Table 13 – Example of an AWS Cloud Adoption Framework (AWS CAF) security control appearing multiple times in a Courses of Action Matrix

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Exploitation	Sec.Det.1	Sec.IAM.1	Sec.Inf.2	Sec.IR.1	Sec.IR.10	Sec.IAM.1	Sec.Det.2	Sec.Inf.9
	Sec.Det.9			Sec.Inf.1	Sec.IR.11	Sec.IAM.4	Sec.Det.11	Sec.Inf.14
	Sec.Det.10	Sec.DP.7	Sec.DP.7	Sec.Inf.9	Sec.IR.2	Sec.Inf.17	Sec.IR.14	Sec.IR.13
	Sec.Det.11	Sec.Inf.17	Sec.Inf.17	Sec.Inf.30		Sec.Inf.18	Sec.IR.15	Ops.1
	Sec.Inf.2	Sec.Inf.18	Sec.Inf.18	Ops.2		Sec.Inf.19	Sec.Inf.29	Ops.2
	Sec.Inf.3	Sec.Inf.19	Sec.Inf.20			Sec.Inf.23	Ops.3	
	Sec.Det.5	Sec.Inf.20	Sec.Inf.21			Sec.Inf.25	Sec.IR.7	
	Sec.IR.14	Sec.Inf.21	Sec.Inf.22			Sec.IR.14		
	Sec.IR.15	Sec.Inf.22	Sec.Inf.23			Sec.IR.15		
	Sec.IR.12	Sec.Inf.23	Sec.IR.14			Platform.1		
	Sec.Inf.27	Sec.Inf.24	Sec.IR.15			Platform.4		
	Sec.Inf.28	Sec.IR.14	Sec.IR.12			Sec.DP.5		
		Sec.IR.15	Sec.Inf.30					
		Sec.IR.12	Ops.2					
		Sec.Inf.27	Sec.DP.5					
		Sec.Inf.28	Sec.DP.6					
		Sec.Inf.32						
		Platform.3						
		Sec.DP.1						
		Sec.DP.5						
		Sec.DP.6						

The following table shows each place in the *courses of action matrix* that each control number appears. You can use the control number for each control to help you prioritize your control implementations. For example, notice that control *Sec.Det.1* (Amazon GuardDuty) can provide Detection capabilities in all phases of the intrusion method analysis framework (except Exploit Development).

Table 14 – Controls Mapped to the Intrusion Method

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Recon – Pre-Intrusion	Sec.Det.1 Sec.Det.2 Sec.Inf.2 Sec.Det.6 Sec.Det.3 Sec.Det.4 Sec.Inf.30 Sec.Det.11 Sec.IR.10	Sec.Inf.3 Sec.IAM.3 Sec.DP.3 Sec.Inf.10 Sec.Inf.2 Sec.Inf.4 Sec.Inf.30	Sec.IR.1 Sec.Inf.30	Sec.IR.10 Sec.IR.11	Sec.IR.10 Sec.IR.11 Sec.IR.2	Sec.IR.10 Sec.IR.11	Sec.Inf.2 Sec.IR.1 Sec.Det.2 Sec.Det.4 Sec.Det.7	—
Recon – Post-Intrusion	Sec.Det.1 Sec.Det.2 Sec.Det.6 Sec.Det.3 Sec.Det.4 Sec.Inf.30 Sec.Det.11 Sec.IR.10 Sec.IR.11	Sec.Inf.3 Sec.IAM.3 Sec.DP.3 Sec.Inf.11 Sec.Inf.11 Sec.IAM.5 Sec.Inf.30 Sec.Inf.32	Sec.IR.1 Sec.Inf.30	Sec.IR.10 Sec.IR.11	Sec.IR.10 Sec.IR.11 Sec.IR.9	Sec.IR.10 Sec.IR.11 Sec.IR.9	Sec.Inf.2 Sec.IR.1 Sec.Det.2 Sec.Det.4 Sec.Det.7	—

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Exploit Developm nt	—	—	—	—	—	—	—	—

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Delivery	Sec.Det.1 Sec.Inf.2 Sec.Inf.13 Sec.Det.8 Sec.Det.9 Sec.Det.10 Sec.Det.11	Sec.Inf.3 Sec.Inf.4 Sec.Inf.5 Sec.Inf.6 Sec.Inf.13 Sec.IAM.2 Sec.IAM.4 Sec.IAM.5 Sec.Inf.17 Sec.Inf.18 Sec.Inf.19 Sec.Inf.23 Sec.Inf.24 Platform.5 Sec.Inf.30	Sec.Inf.3 Sec.Inf.5 Sec.Inf.6 Sec.Inf.13 Ops.2 Sec.Inf.30 Sec.Det.9 Sec.Det.10	Sec.IR.1 Sec.Inf.13 Sec.Inf.8 Ops.2	Sec.IR.10 Sec.IR.11 Sec.IR.2	Sec.Inf.1 Sec.Inf.3 Sec.Inf.5 Sec.Inf.6 Sec.IAM.4 Sec.Inf.25 Platform.1 Platform.2 Platform.4 Sec.DP.5	Sec.Inf.14 Sec.IR.13 Sec.IR.2 Sec.IR.3 Sec.IR.5 Sec.IR.6 Sec.IR.7 Ops.3 Sec.Det.9 Sec.Det.10	Sec.Inf.14 Ops.1 Ops.2

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
		Sec.Inf.31 Sec.Inf.32 Sec.DP.5 Sec.DP.6						

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Exploitation	Sec.Det.1 Sec.Det.9 Sec.Det.10 Sec.Det.11 Sec.Inf.2 Sec.Inf.3 Sec.Det.5 Sec.IR.14 Sec.IR.15 Sec.IR.12 Sec.Inf.27 Sec.Inf.28	Sec.IAM.1 Sec.DP.7 Sec.Inf.17 Sec.Inf.18 Sec.Inf.19 Sec.Inf.20 Sec.Inf.21 Sec.Inf.22 Sec.Inf.23 Sec.Inf.24 Sec.IR.14 Sec.IR.15 Sec.IR.12 Sec.Inf.27 Sec.Inf.28	Sec.Inf.2 Sec.DP.7 Sec.Inf.17 Sec.Inf.18 Sec.Inf.20 Sec.Inf.21 Sec.Inf.22 Sec.Inf.23 Sec.IR.14 Sec.IR.15 Sec.IR.12 Sec.Inf.30 Ops.2 Sec.DP.5 Sec.DP.6	Sec.IR.1 Sec.Inf.1 Sec.Inf.9 Sec.Inf.30 Ops.2	Sec.IR.10 Sec.IR.11 Sec.IR.2	Sec.IAM.1 Sec.IAM.4 Sec.Inf.17 Sec.Inf.18 Sec.Inf.19 Sec.Inf.23 Sec.Inf.25 Sec.IR.14 Sec.IR.15 Platform.1 Platform.4 Sec.DP.5	Sec.Det.2 Sec.Det.11 Sec.IR.14 Sec.IR.15 Sec.Inf.29 Ops.3 Sec.IR.7	Sec.Inf.9 Sec.Inf.14 Sec.IR.13 Ops.1 Ops.2

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
		Sec.Inf.32 Platform.3 Sec.DP.1 Sec.DP.5 Sec.DP.6						

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Installation	Sec.Det.1 Sec.Det.6 Sec.Det.3 Sec.Det.4 Sec.Det.9 Sec.Det.10 Sec.Det.11 Sec.Inf.16 Sec.IR.14 Sec.IR.15 Sec.IR.12	Sec.IAM.2 Sec.IAM.4 Sec.IAM.5 Sec.Inf.17 Sec.Inf.18 Sec.Inf.22 Sec.Inf.23 Sec.Inf.26 Sec.Inf.32 Sec.IR.12 Sec.DP.5 Sec.DP.6	Sec.Inf.14 Sec.Inf.17 Sec.Inf.18 Sec.Inf.22 Sec.Inf.23 Sec.Inf.26 Sec.IR.13 Sec.IR.12 Sec.DP.6	Sec.Inf.8 Sec.Inf.14 Sec.Inf.19 Sec.Inf.26 Sec.IR.13 Ops.2	Sec.IR.10 Sec.IR.11	Sec.IAM.4 Sec.Inf.17 Sec.Inf.18 Sec.Inf.23 Sec.Inf.25 Sec.IR.14 Sec.IR.15 Platform.1 Platform.4 Sec.DP.5	Sec.Inf.14 Sec.Inf.15 Sec.Inf.16 Sec.IR.13 Sec IR.7	Sec.Inf.10 Sec.Inf.14 Sec.IR.13 Ops.1 Ops.2

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Command and Control	Sec.Det.1 Sec.Det.6 Sec.Det.3 Sec.Det.4 Sec.Det.11 Sec.Inf.8 Sec.Inf.12 Sec.IR.14 Sec.IR.15	Sec.IAM.2 Sec.IAM.4 Sec.IAM.5 Sec.Inf.3 Sec.Inf.5 Sec.Inf.6 Sec.IR.14 Sec.IR.15	Sec.Inf.3 Sec.Inf.5 Sec.Inf.6 Sec.IR.14 Sec.IR.15 Sec.IR.1 Sec.IR.4 Ops.2	Sec.IR.1 Sec.IR.4 Ops.2	Sec.IR.10	Sec.IAM.2 Sec.IAM.4 Sec.Inf.3 Sec.Inf.5 Sec.Inf.6 Sec.Inf.25 Sec.Inf.30 Platform.1 Platform.2 Platform.4	Sec.IR.14 Sec.IR.15 Sec.Inf.29 Sec.IR.1 Ops.3	Sec.Inf.9 Sec.Inf.14 Sec.IR.13 Ops.1 Ops.2 Ops.4

	Detect	Deny	Disrupt	Degrade	Deceive	Contain	Respond	Restore
Actions on Objectives	Sec.Det.1 Sec.Det.6 Sec.Det.3 Sec.Det.4 Sec.Det.11 Sec.Inf.8 Sec.Inf.13 Sec.IR.14 Sec.IR.15 Sec.DP.4	Sec.IAM.2 Sec.IAM.4 Sec.IAM.5 Sec.Inf.17 Sec.Inf.18 Sec.Inf.23 Sec.IR.14 Sec.IR.15 Sec.DP.1 Sec.DP.2	Sec.IAM.2 Sec.Inf.17 Sec.Inf.18 Sec.Inf.23 Sec.IR.14 Sec.IR.15 Sec.IR.5 Ops.2	Sec.IAM.2 Sec.Inf.9 Sec.Inf.17 Sec.Inf.18 Sec.Inf.23 Sec.DP.4	Sec.IR.10	Sec.IAM.2 Sec.IAM.4 Sec.Inf.3 Sec.Inf.5 Sec.Inf.6 Sec.Inf.25 Platform.1 Platform.4	Sec.IR.14 Sec.IR.15 Sec.Inf.29 Sec.IR.1 Ops.3 Sec IR.7	Sec.Inf.9 Sec.IR.13 Ops.1 Ops.4

Note

**Defined in the 2006 version of JP 3-13, as documented in Mitre, "Characterizing Effects on the Cyber Adversary, A Vocabulary for Analysis and Assessment", https://www.mitre.org/sites/default/files/publications/characterizing-effects-cyber-adversary-13-4173.pdf

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.