

SRINIVAS UNIVERSITY
INSTITUTE OF ENGINEERING AND TECHNOLOGY

MUKKA, SURATHKAL, MANGALORE-574146



MAJOR PROJECT REPORT

ON

**“Detecting & Mitigating Ransomware Threats in AWS
Architecture Using AWS Services”**

*Submitted in the partial fulfillment of the requirements for the award of
the degree of*

**BACHELOR OF TECHNOLOGY
IN
CLOUD TECHNOLOGY & INFORMATION SECURITY**

Submitted By,

Akshay Shetty	1SU19CI005
Delvy Austin Fernandes	1SU19CI010

Under the Guidance of

Dr. A Sasi Kumar

Professor of CTDS Department

2022-2023

SRINIVAS UNIVERSITY
INSTITUTE OF ENGINEERING & TECHNOLOGY
Mukka, Mangalore-574146



CERTIFICATE

This is to certify that the project entitled “**Detecting & Mitigating Ransomware Threats in AWS Architecture Using AWS Services**” is a Bonafide work carried out by **Akshay Shetty, Delvy Austin Fernandes** bearing the **1SU19CI005, 1SU19CI0010** in the partial fulfillment of **Bachelor of Technology in Cloud Technology & Information Security** of the **Srinivas University Institute of Engineering and Technology** during the year **2022-2023**. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the department library. The internship report has been approved as it satisfies the academic requirements regarding the prescribed internship work for the said degree.

Name & Signature of the Guide

Dr. A. Sasi Kumar

Name & Signature of the H.O.D

Mr. Daniel Francis Selvaraj

Signature of the Dean

Dr. Thomas Pinto

Dean, SUIET, Mukka

External examiners (Viva)

Name of the Examiners

Signature with date

1. -----

2. -----

SRINIVAS UNIVERSITY
INSTITUTE OF ENGINEERING & TECHNOLOGY
Mukka, Mangalore-574146



DECLARATION

We, **Akshay Shetty & Delvy Austin Fernandes** the students in the eighth semester, **B.Tech** in Cloud Technology & Information Security, Srinivas University, Mukka, hereby declare that the project entitled “**Detecting & Mitigating Ransomware Threats in AWS Architecture Using AWS Services**” has been completed by me in partial fulfillment of the requirements for the award of degree in **Bachelor of Technology in Cloud Technology & Information Security, of Srinivas University Institute of Engineering and Technology** and no part of it has been submitted for the award of degree or diploma in any university or institution previously.

Date:_____

Place: Mukka

ACKNOWLEDGEMENT

We would like to take this opportunity to express our profound gratitude to our respected project guide, **Dr. A. Sasi Kumar, Professor of Data Science in the CTDS department**, for his ever-inspiring guidance, constant encouragement, and support.

We would like to express our deep gratitude and indebtedness to our project coordinator, **Mrs. Renisha S., Professor of the CTDS Department**, for the encouragement and guidance that she has extended while carrying out our project.

We sincerely thank **Mr. Daniel Francis Selvaraj, Head of the Department of CTDS**, for being an inspiration and support throughout this project.

We are extremely grateful to our respected Dean, **Dr. Thomas Pinto**, for providing the facilities to carry out the project.

We would also like to thank our management, the **A. Shama Rao Foundation**, Mangalore, and **iNurture Education Private Limited**, Bangalore for providing the means and support for the completion of the project.

We would like to thank all the teaching and non-teaching staff of the **CTDS Department** and **SUIET** for their support and help.

Finally, we express our profound gratitude to our parents and friends, who have helped us in every conceivable manner with their valuable suggestions, encouragement, and moral support.

Akshay Shetty
Delvy Austin Fernandes

TABLE OF CONTENTS

➤ List of Figures	1
➤ Abbreviations & Nomenclature	2
➤ Agenda	3
➤ Abstract	4
1. Introduction	
1.1 The Domain	7
1.2 Problem Statement	14
1.3 The Technology	16
a) Hardware Requirements	19
b) Software Requirements	19
c) Operating System Requirements	29
1.4 Objectives of Project	29
2. Literature Survey / Related Works	
2.1 Literature Review	30
2.2 Existing Systems	31
2.3 Proposed System-Planning	32
2.4 Motivation to implement	34
3. System Design Methodology	
3.1 System Architecture Diagram	35
3.2 Description of work	35
3.3 Some Usable Algorithms	37
4. Implementation / Results	
4.1 Experimental Setup	39
4.2 Comparative Study	60
4.3 Proposed Results	60
5. Conclusion	61
6. References	62

List of Figures

Figures	Page No.
1. Evolution of Ransomware	5
2. Table representing the shared responsibility model in AWS	11
3. History of AWS	10
4. AWS Global Infrastructure	10
5. Worldwide AWS Regions	10
6. A representation of AWS Well-Architected Framework	12
7. Table representing shared responsibility model in AWS	13
8. Some of the Ransomware Statistics	15
9. Various losses due to Ransomware	15
10. Comparisons between On-premises versus Cloud Environment	16
11. Simple Architecture in AWS	17
12. Representative image of a café business that is planning to go online operationally	32
13. Representational image of the café migrating its online system into the cloud	33
14. Representational image of a business going global using AWS cloud services	33
15. A representational large-scale architecture	35
16. Pictorial representation of Cross Region & Cross account DR	37
17. Benefits of backup & restore in AWS	37

Abbreviations and Nomenclature

ACL: Access Control List

AMI: Amazon Machine Image

API Gateway: Amazon API Gateway

Application Auto Scaling: AWS Application Auto Scaling is a web service that you can use to configure automatic scaling for AWS resources beyond Amazon EC2, such as Amazon ECS services, Amazon EMR clusters, and DynamoDB tables.

ASG (Auto Scaling group): A representation of multiple EC2 instances that share similar characteristics and are treated as a logical grouping for instance scaling and management.

AZ (Availability Zone): A distinct location within a Region that's insulated from failures in other Availability Zones, and provides inexpensive, low-latency network connectivity to other Availability Zones in the same Region.

AWS: Amazon Web Services.

AWS Billing: Billing and Cost Management.

AWS CDK: AWS Cloud Development Kit

AWS CLI: AWS Command Line Interface.

CloudFront: Amazon CloudFront (content delivery network)

CloudFormation: AWS CloudFormation (infrastructure as code)

CloudWatch: Amazon CloudWatch (monitoring and observability service)

Cognito: Amazon Cognito (user authentication and authorization service)

EC2: Amazon Elastic Compute Cloud

ELB: Elastic Load Balancer

Elastic Beanstalk: AWS Elastic Beanstalk is a web service for deploying and managing applications in the AWS Cloud without worrying about the infrastructure that runs those applications.

EBS: Elastic Block Store

EMR: Amazon Elastic MapReduce (big data processing service)

IAM: Identity and Access Management

IoT: Internet of Things

JSON: JavaScript Object Notation.

Lambda: AWS Lambda (serverless compute service)

Macie: Amazon Macie is a security service that uses machine learning to automatically discover, classify, and protect sensitive data in AWS.

Route 53: AWS Domain Name System (DNS) service

RDS: Amazon Relational Database Service

S3: Amazon Simple Storage Service

SQS: Amazon Simple Queue Service

SNS: Amazon Simple Notification Service

SSO: Single Sign-On

Stack in CloudFormation: A collection of AWS resources that you create and delete as a single unit.

YAML: Yet another markup language.

Agenda

In this project we will do three things:

➤ ***Protect:***

Use AWS Systems Manager to update and configure our EC2 instances – to mitigate system vulnerabilities.

➤ ***Detect:***

Use Guard Duty, Security Hub, and Systems Manager to detect malicious behaviour by an EC2 instance and investigate it further for the presence of ransomware.

➤ ***Respond:***

Use Systems Manager automation to gather data from an affected system, snapshot it, and isolate it. This enables the incident response team to do further investigation and forensic work.

Abstract

Ransomware attacks have emerged as a significant threat to organizations' data and infrastructure, causing extensive financial and operational damage. It becomes crucial to develop robust security measures to detect and mitigate the risks associated with ransomware attacks, as businesses increasingly rely on cloud computing platforms like Amazon Web Services (AWS).

Identify & Protect, Detect & Respond threat detection can continuously monitor our AWS accounts and workloads for malicious activity and deliver detailed security findings for visibility and remediation. Early detection of abnormal network activity is key to mitigating ransomware threats and their impact.

Identify and protect:

Identifying your systems, critical data, and applications will help you baseline normal user activity as well as the integrity of systems and potential vulnerabilities. By rapidly identifying and patching vulnerabilities, organizations can reduce their exposure to ransomware events by limiting the ways it can get in.

Detect and respond:

Threat detection can continuously monitor your AWS accounts and workloads for malicious activity and deliver detailed security findings for visibility and remediation. Early detection of anomalous network activity is key to mitigating ransomware threats and their impact.


Recover:

Organizations that identify critical data upfront can back up that data to create an immutable recovery copy. Data can be recovered to a specific point in time and rapidly restored reducing an incident's impact. With AWS services, you can centralize and automate data backups, simplify backup management, and protect your application data across AWS and on-premises environments.

1. Introduction

Ransomware is a type of malicious software that encrypts a victim's files or locks their computer, rendering the data inaccessible until a ransom is paid to the attacker. It is a form of cyber extortion, where hackers demand payment in exchange for restoring the victim's files or granting access to their systems.

The evolution of ransomware



- ✓ Ransomware = **malicious code** designed to gain unauthorized data access and **encrypt data** to block access by legitimate users
- ✓ First instance of ransomware (1989) had simple encryption tools and user interaction was necessary. Now more complicated, using complex algorithms, no interaction necessary.
- ✓ Ransomware is a **top concern for organizations** with CryptoLocker (2014), Petya (2016), WannaCry (2017), notPetya (2017), Ryuk (2019)
- ✓ Current trends include RaaS, [cryptocurrency mining](#)

Evolution of Ransomware

Ransomware attacks typically involve the following steps:

- 1) **Delivery:** Ransomware can be delivered through various means, including malicious email attachments, infected websites, or compromised software.
- 2) **Encryption:** Once the ransomware infects a system, it starts encrypting files on the victim's computer or network. This process converts the data into a format that can only be decrypted with a unique decryption key held by the attacker.
- 3) **Ransom Note:** After encrypting the files, the ransomware displays a message to the victim, often in the form of a text file or a pop-up window. This note contains instructions on how to pay the ransom and obtain the decryption key.
- 4) **Ransom Payment:** The attacker demands payment, typically in a cryptocurrency like Bitcoin, as it provides a certain level of anonymity. The ransom note provides details on the payment process, which may involve accessing a specific website on the dark web or communicating through anonymous channels.
- 5) **Decryption (in some cases):** If the victim decides to pay the ransom, they may receive a decryption key or tool from the attacker to unlock their files. However, there's no guarantee that the attacker will honour the payment or provide a functional decryption key.

It's important to note that paying the ransom does not guarantee the recovery of the files or protection against future attacks. Paying the ransom encourages and funds the activities of cybercriminals, perpetuating the ransomware ecosystem.

To protect against ransomware attacks, it is crucial to maintain regular backups of important data, keep software and systems up to date with security patches, use robust antivirus and anti-malware software, exercise caution when opening email attachments or clicking on suspicious links, and educate employees or individuals about the best practices for cybersecurity.

Criminal organizations use sophisticated methods to compromise their victims' systems and encrypt the data, usually to extort money from the victim. An AWS customer benefits from AWS data centres and network infrastructure, which are architected to protect information, identities, and applications.

Ransomware attacks targeting cloud computing environments have become a significant concern in recent years. Cloud computing offers numerous benefits, such as scalability, cost-efficiency, and ease of management, but it also introduces new security challenges. Ransomware is a type of malicious software that encrypts the victim's data and demands a ransom payment in exchange for the decryption key.

Here are some key points to consider regarding ransomware in cloud computing:

- **Shared Responsibility Model:** In cloud computing, the responsibility for security is shared between the cloud service provider (CSP) and the customer. The CSP is responsible for securing the underlying infrastructure, while the customer is responsible for securing their data and applications. Ransomware attacks can occur due to vulnerabilities on either side.
- **Misconfigurations:** Misconfigured cloud storage buckets, weak access controls, or improperly configured security groups can leave cloud environments vulnerable to ransomware attacks. Attackers can exploit these misconfigurations to gain unauthorized access and deploy ransomware.
- **Phishing and Social Engineering:** Ransomware attacks often begin with phishing emails or social engineering techniques targeting cloud users. Attackers trick users into clicking malicious links or downloading infected attachments, which can lead to the deployment of ransomware within the cloud environment.
- **Multi-Tenant Environments:** Cloud computing typically involves multi-tenant environments where multiple customers share the same infrastructure. If one customer becomes infected with ransomware, there is a risk that it can spread to other customers' data within the same environment if adequate isolation measures are not in place.
- **Data Backup and Recovery:** Regularly backing up data and having an effective disaster recovery plan is crucial to mitigating the impact of ransomware attacks. If data is encrypted by ransomware, organizations can restore their systems using clean backups instead of paying the ransom.
- **Encryption and Access Controls:** Employing strong encryption mechanisms for data at rest and in transit within the cloud environment can help protect against unauthorized access and mitigate the impact of ransomware attacks.

Additionally, implementing robust access controls, including multi-factor authentication, can reduce the likelihood of unauthorized access.

- **Security Monitoring and Incident Response:** Implementing robust security monitoring tools and practices can help detect ransomware attacks early. Having an incident response plan in place allows organizations to respond swiftly to contain and mitigate the effects of a ransomware attack.

Organizations and cloud users need to stay updated on the latest security best practices, regularly patch and update their systems, train employees on security awareness, and work closely with their cloud service providers to ensure a secure cloud environment.

1.1 The Domain

Cloud computing is a technology that enables the delivery of various computing services over the Internet. It encompasses a wide range of services and resources that can be accessed and utilized remotely. The domain of cloud computing covers the following areas:

- **Infrastructure as a Service (IaaS):**
This includes the provision of virtualized computing resources such as virtual machines, storage, and networks. Users can deploy and manage their software applications and configurations on these resources.
- **Platform as a Service (PaaS):**
PaaS provides a platform for developers to build, test, and deploy applications. It includes the underlying infrastructure, operating system, and development tools required to support the application lifecycle.
- **Software as a Service (SaaS):**
SaaS delivers software applications over the internet on a subscription basis. Users can access and use these applications through a web browser or specialized client applications without having to worry about installation, maintenance, or infrastructure management.
- **Serverless Computing:**
Serverless computing abstracts the underlying infrastructure from developers, allowing them to focus on writing and executing code without managing servers or scaling resources. It automatically scales applications based on demand and charges users based on actual usage.
- **Storage as a Service:**
This includes cloud-based storage solutions that provide scalable and durable storage for data. Users can store and retrieve their data from anywhere with an internet connection without having to manage the physical infrastructure.

- **Database as a Service (DBaaS):**
DBaaS offers managed database solutions in the cloud, eliminating the need for users to set up and maintain their database infrastructure. It provides features such as automatic backups, scaling, and high availability.
- **Big Data and Analytics:**
Cloud computing offers services and tools for processing and analysing large volumes of data. This includes services like data warehousing, data lakes, and big data analytics frameworks.
- **Internet of Things (IoT):**
Cloud platforms provide capabilities to connect, manage, and process data from IoT devices. They offer services for device management, data ingestion, real-time analytics, and integration with other cloud services.
- **Artificial Intelligence (AI) and Machine Learning (ML):**
Cloud computing provides resources and services for training and deploying AI and ML models. It includes pre-built models, development frameworks, and scalable computing resources for running complex AI and ML workloads.

These are some of the main domains within cloud computing, and the technology continues to evolve with new services and capabilities being introduced regularly.

Amazon Web Services

AWS stands for Amazon Web Services, which is a comprehensive cloud computing platform offered by Amazon. It provides a wide range of cloud services and solutions that enable organizations to build and deploy applications, store and analyse data, and scale their infrastructure on-demand. AWS offers a vast array of services across various domains, including:

- **Computing Services:**
AWS provides virtual servers known as Amazon EC2 (Elastic Compute Cloud) that allow users to run applications in the cloud. EC2 offers flexible compute capacity, scalability, and a variety of instance types to suit different workload requirements.
- **Storage and Content Delivery:**
AWS offers different storage options, including Amazon S3 (Simple Storage Service) for object storage, Amazon EBS (Elastic Block Store) for block-level storage volumes and Amazon Glacier for long-term archival storage. It also provides Amazon CloudFront, a content delivery network (CDN) for fast and secure content delivery.
- **Database Services:**
AWS offers a range of managed database services, including Amazon RDS (Relational Database Service) for relational databases like MySQL, PostgreSQL, and Oracle, Amazon DynamoDB for NoSQL databases, Amazon Redshift for data warehousing, and Amazon Neptune for graph databases, among others.

- **Networking and Content Delivery:**
AWS provides services for virtual private clouds (VPCs), which allow users to create isolated network environments in the cloud. It also offers services like Amazon Route 53 for domain name system (DNS) management, AWS Direct Connect for dedicated network connections, and Elastic Load Balancing for distributing incoming application traffic.
- **Analytics and Big Data:**
AWS offers services for processing and analysing large volumes of data, including Amazon EMR (Elastic MapReduce) for big data processing using frameworks like Hadoop and Spark, Amazon Athena for querying data stored in S3 using SQL, and Amazon Kinesis for real-time streaming data analytics.
- **Machine Learning and Artificial Intelligence:**
AWS provides a comprehensive set of services for AI and ML, such as Amazon SageMaker for building, training, and deploying ML models, Amazon Rekognition for image and video analysis, and Amazon Comprehend for natural language processing and text analysis.
- **Security and Identity Services:**
AWS offers various security services and tools, including AWS Identity and Access Management (IAM) for managing user access and permissions, AWS Shield for DDoS protection, AWS WAF (Web Application Firewall) for protecting web applications, and AWS Key Management Service (KMS) for managing encryption keys.
- **Management and Monitoring:**
AWS provides services for managing and monitoring cloud resources, such as AWS CloudFormation for infrastructure as code, AWS CloudTrail for logging and auditing API calls, and Amazon CloudWatch for monitoring and collecting metrics from AWS services.

These are just a few examples of the extensive range of services and solutions provided by AWS. The platform continues to evolve and expand with new offerings being introduced regularly to address various business needs and technological advancements.

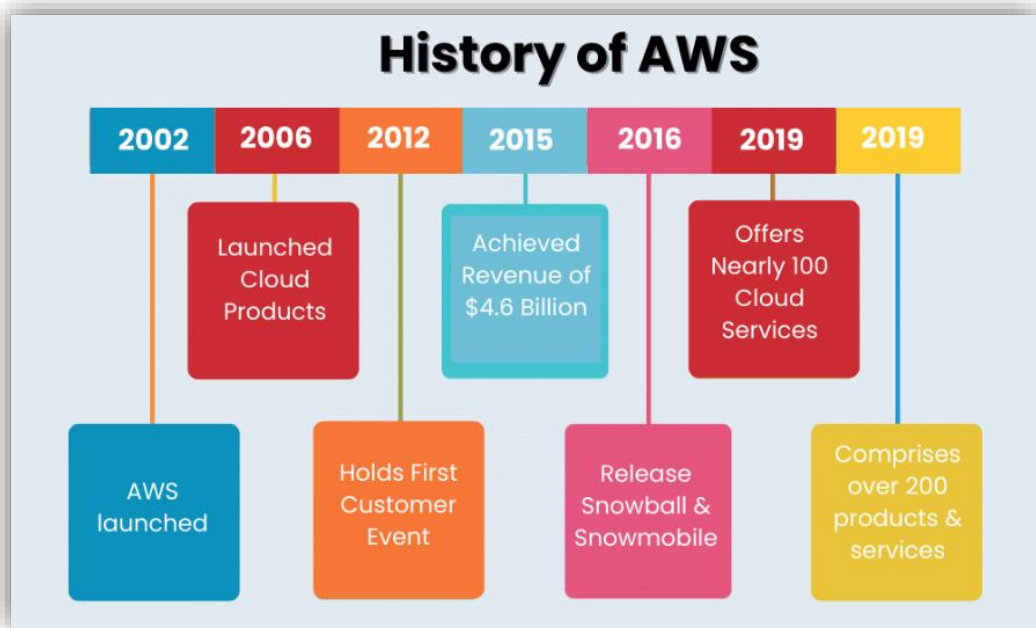


Fig: History of AWS.

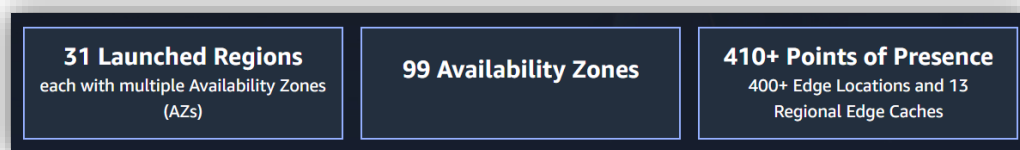


Fig: AWS Global Infrastructure.

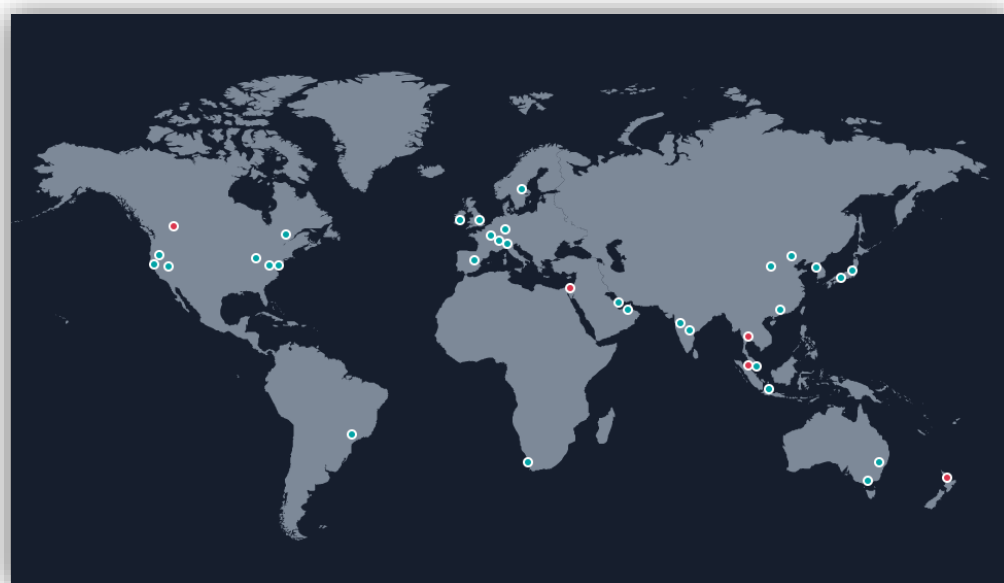


Fig: Location of AWS Regions Worldwide.

Security in AWS

Security is a top priority for AWS (Amazon Web Services), and the platform provides a comprehensive set of security features and services to help users protect their data, applications, and infrastructure. Here are some key aspects of security in AWS:

➤ **Data Encryption:**

AWS supports encryption at rest and in transit. Users can encrypt their data stored in AWS services like Amazon S3, Amazon EBS, and Amazon RDS using AWS Key Management Service (KMS) or their encryption keys. Additionally, AWS provides SSL/TLS encryption for data in transit.

➤ **Identity and Access Management:**

AWS Identity and Access Management (IAM) enables users to manage access to AWS services and resources. IAM allows users to create and manage users, groups, and roles with fine-grained permissions. Multi-factor authentication (MFA) can be enabled for additional security.

➤ **Network Security:**

AWS offers Virtual Private Cloud (VPC), which allows users to create isolated virtual networks and control network traffic flow. Users can configure security groups and network access control lists (ACLs) to define firewall rules and restrict network access. AWS also provides distributed denial-of-service (DDoS) protection with services like AWS Shield and AWS WAF.

➤ **Compliance and Governance:**

AWS adheres to various compliance frameworks and industry standards, including ISO 27001, SOC 1/2/3, HIPAA, GDPR, and more. Users can utilize services like AWS Config, AWS CloudTrail, and AWS Trusted Advisor to monitor and enforce compliance with best practices and security controls.

➤ **Security Monitoring and Logging:**

AWS CloudTrail provides detailed logs of API calls and activity within AWS accounts. Amazon GuardDuty offers intelligent threat detection for identifying malicious activity and unauthorized behaviour. Amazon CloudWatch allows users to collect and analyse logs and metrics to monitor the security and performance of their AWS resources.

➤ **Incident Response and Automation:**

AWS provides services like AWS Systems Manager, AWS Config, and AWS CloudFormation that enable automated security and compliance remediation. AWS also offers Incident Response services and guidance to help users respond to security incidents effectively.

➤ **Secure Compute Services:**

AWS offers secure compute services like Amazon EC2 (Elastic Compute Cloud), where users can apply security best practices such as using secure AMIs (Amazon Machine Images), patch management, and network isolation.

AWS Lambda, the serverless service, benefits from built-in security features and automatic scaling.

➤ **Auditing and Penetration Testing:**

AWS allows users to conduct security assessments by performing vulnerability scans, penetration testing, and security audits. AWS provides a wide range of resources and guidelines to ensure such activities are conducted safely and with prior authorization.

It's important to note that while AWS provides a secure infrastructure, it is the responsibility of users to properly configure and secure their own applications and data deployed on AWS. Following AWS security best practices and guidelines is crucial to maintain a secure environment.

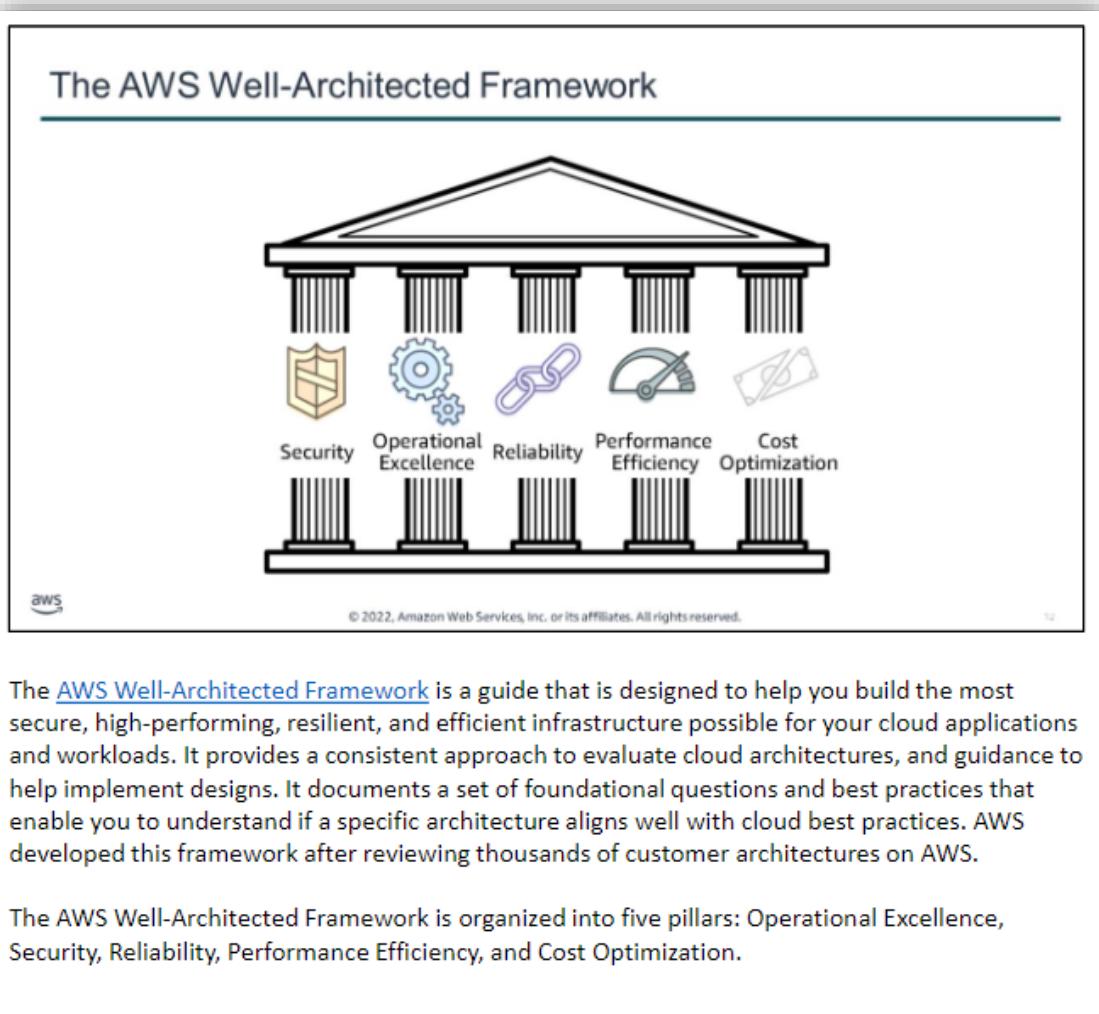


Fig: A representation of AWS Well-Architected Framework

Concept of shared responsibility model in AWS:

The shared responsibility model is a security framework used by AWS (Amazon Web Services) to clarify the division of security responsibilities between AWS and its customers. Under this model, both AWS and customers have distinct security responsibilities depending on the type of service being used. The model helps ensure that security measures are in place to protect AWS infrastructure and customer data.

An overview of the shared responsibility model in AWS:

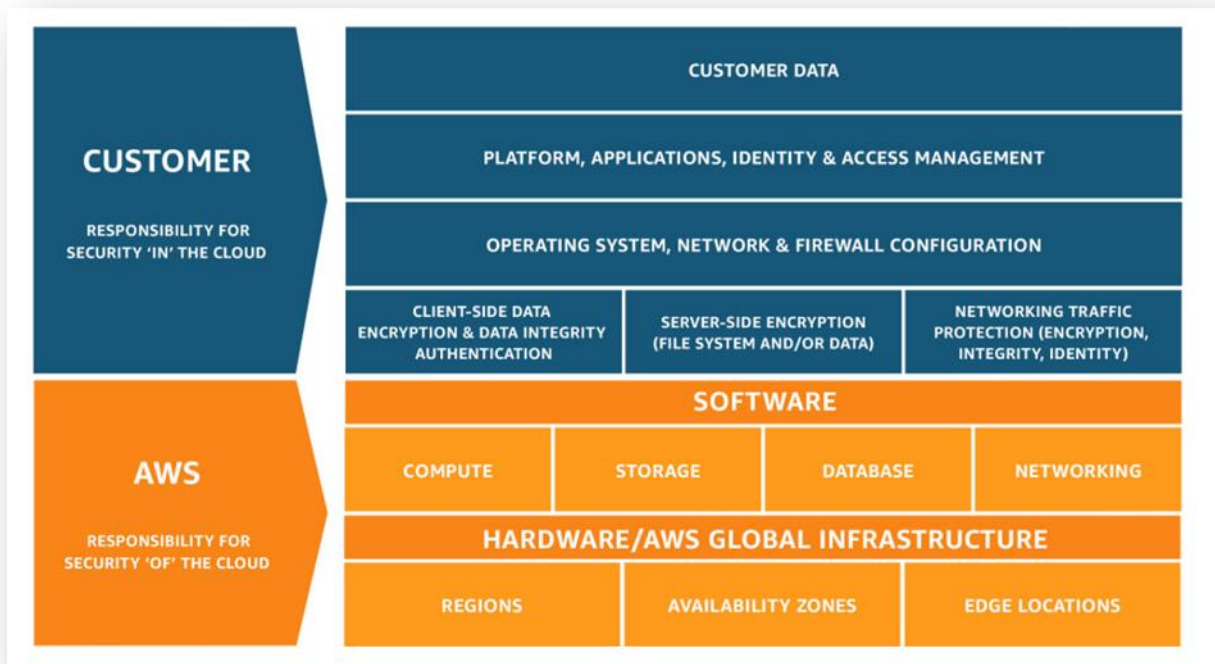


Fig: Table representing shared responsibility model in AWS.

AWS Responsibility:

- AWS is responsible for the security of the cloud infrastructure, including the physical data centres, networking, and virtualization layers.
- AWS ensures the availability and reliability of its global infrastructure, including power and cooling systems, networking equipment, and hardware maintenance.
- AWS manages and secures the underlying infrastructure services like EC2, S3, RDS, and others, ensuring they are patched, updated, and protected against common security threats.
- AWS provides core security features such as DDoS protection, encryption options, and identity and access management (IAM) tools.

Customer Responsibility:

- Customers are responsible for securing their applications, operating systems, data, and configurations deployed on AWS services.
- Customers have control over their AWS accounts and are responsible for managing user access, permissions, and credentials using AWS IAM.
- Customers are responsible for applying security patches and updates to their EC2 instances, virtual machines, and operating systems.
- Customers must implement network security measures such as configuring security groups, network ACLs, and firewall rules to control inbound and outbound traffic.
- Customers are responsible for encrypting sensitive data, both in transit and at rest, using AWS-provided encryption mechanisms or their own encryption tools.
- Customers are responsible for managing and securing their applications, including application-level access controls, authentication, and data encryption within the application.

The specific division of responsibilities may vary depending on the AWS services being used. For example, when using AWS RDS (Relational Database Service), AWS manages the infrastructure and underlying database engine, while the customer is responsible for configuring and securing the database instances, managing user access, and encrypting data.

It's important for customers to understand and fulfill their part of the shared responsibility model by following best practices and implementing appropriate security measures within their AWS deployments. AWS provides extensive documentation, guidelines, and security features to assist customers in achieving a secure environment.

1.2 Problem Statement

The Vulnerability Aspect:

Running in AWS alone, however, is not enough to protect the systems from ransomware or other cybersecurity threats. The user is responsible for securing the aspects of the environment one controls. (Concept of AWS Shared Responsibility Model) AWS provides users with robust tools to keep their environment secure at scale. In this project, we will explore how some of these tools can enhance our ability to protect against ransomware, detect malicious activity, and respond to malicious activity if it is detected.

Some Ransomware Artifacts:

American oil pipeline system Colonial Pipeline, Acer, Chicago-based CNA Financial Corp, Kia Motors, and many Indian companies are the victims of Ransomware attacks.

The shattering fact about Ransomware attack

- The average ransom paid in 2020 was \$312,493 USD.
- The largest recorded ransom paid was \$4.5 million USD.
- In 2022, organizations all around the world detected 493.33 million ransomware attacks.
- If you pay the ransom, there's a reported 97% chance that you'll get an activation key that would decrypt your data.
- However, 46% of the time, companies report there's some level of corruption. And, of course, restoration isn't immediate, so there's still a massive impact.
- 80% of paying organizations reportedly get hit again. It makes sense. Once you've paid out, organizations reveal their weakness.

According to Gartner, by 2025, ransomware attacks are expected to increase by 700% and at least 75% of IT organizations will face one or more attacks.

- Gartner, January 2021.

S3 buckets are advertised by AWS as extremely durable. Recent Ermetic research found that misconfigurations of S3 buckets and access-related factors made exposure to potential ransomware in the real-world sample studied extremely common. This potential risk calls for organizations to take urgent action to correct any such S3 bucket misconfigurations and access-related factors.

- Ermetic research

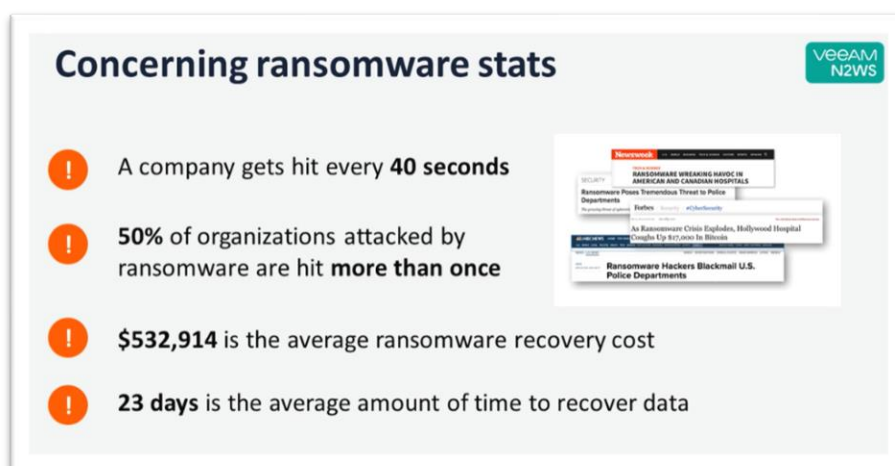


Fig: Some of the Ransomware Statistics

The biggest cost isn't the ransom

Downtime creates **loss of productivity of users and responders**, exposure of **sensitive data**, **loss of revenue** – current and future costs include:

- Data damage
- Restoration of host systems and data
- Downtime due to attacks (no productivity/revenue)
- Forensic investigation
- Damage to the reputations of victims

**Loss of productivity & non-availability is the primary business impact of ransomware*

Fig: Various losses due to Ransomware

1.3The Technology

Cloud computing is the delivery of computing resources, including servers, storage, databases, networking, software, and analytics, over the Internet on a pay-per-use basis. Instead of owning and maintaining physical infrastructure, users can access and utilize these resources provided by a cloud service provider.

The choice between cloud computing and on-premises computing depends on factors such as scalability needs, cost considerations, resource control requirements, geographic distribution, security and compliance requirements, and the organization's overall IT strategy. Many organizations adopt a hybrid approach, leveraging a mix of cloud and on-premises infrastructure to meet their specific needs.

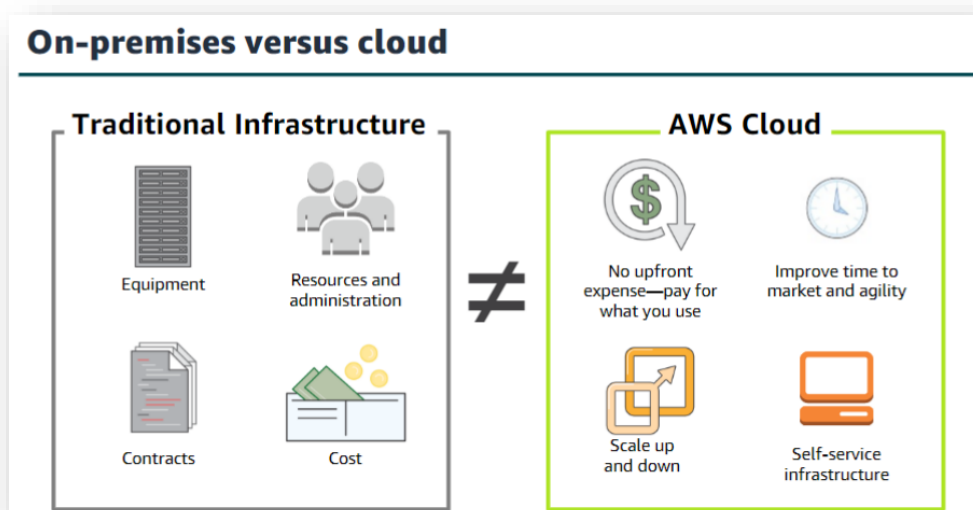


Fig: Comparisons between On-premises versus Cloud Environment

In AWS, a solution architecture refers to the design and structure of a solution that leverages various AWS services to address specific business requirements or solve a particular problem. It involves designing the components, infrastructure, data flow, integration points, and security considerations to create a scalable, reliable, and efficient solution.



AWS Services to be Used

Amazon Elastic Compute Cloud (Amazon EC2) offers the broadest compute platform, with the deepest instance choice to match your workload's needs.

Amazon's simple storage service with Object storage is built to retrieve any amount of data from anywhere.

AWS Cloud Formation:

It allows you to define your infrastructure as code using a declarative JSON or YAML template format, which can be version-controlled, reviewed, and deployed repeatedly.

AWS Lambda:

It is a serverless compute service that allows you to run your code without provisioning or managing servers. With Lambda, you can focus on writing the application logic and let AWS handle the underlying infrastructure, scaling, and availability.

Amazon GuardDuty:

It correlates activity in the AWS environment with threat intelligence from multiple sources that provide additional risk context and anomaly detection.

IAM Access Analyser:

IAM Access Analyzer is a feature of AWS Identity and Access Management (IAM) that helps you analyse and evaluate the access policies in your AWS environment. It helps you identify any unintended or overly permissive access permissions that could pose security risks.

Amazon Macie:

It can identify sensitive data, classify, and label it, and track its location and access.

AWS Security Hub:

It automates response and remediation. AWS Security Hub is a cloud security posture management service that performs security best practice checks, aggregates alerts and enables automated remediation.

AWS Backup:

Centrally manage and automate data protection. AWS Backup is a cost-effective, fully managed, policy-based service that simplifies data protection at scale.

AWS Backup Vault Lock:

AWS Backup Vault Lock enforces a *write-once, read-many* (WORM) setting for all the backups you store and create in a backup vault. With AWS Backup Vault Lock, you can add an additional layer of defence that protects backups (recovery points) in your backup vaults from inadvertent or malicious delete operations and updates.

AWS Systems Manager:

It centralizes operational data from multiple AWS services and automates tasks across your AWS resources. You can create logical groups of resources such as applications, different layers of an application stack, or production versus development environments. It gains operational insights into AWS and on-premises resources. AWS Systems Manager is a secure end-to-end management solution for hybrid cloud environments.

AWS CloudTrail:

It logs all API calls. AWS CloudTrail monitors and records account activity across AWS infrastructure, giving you control over storage, analysis, and remediation actions

Amazon Server-Side Encryption with Amazon Simple Storage Service (Amazon S3) managed encryption keys (SSE-S3) can digitally sign and encrypt the logs and store them in a secure Amazon S3 bucket.

Amazon VPC:

Amazon VPC or virtual private cloud flow Logs monitor all network activity going in and out of VPC. Amazon VPC gives full control over our virtual networking environment, including resource placement, connectivity, and security.

Amazon Inspector:

Amazon Inspector automatically discovers workloads, such as Amazon EC2 instances, containers, and Lambda functions, and scans them for software vulnerabilities and unintended network exposure.

Amazon CloudWatch:

It monitors your AWS environment and generates alerts. Amazon CloudWatch Observes and monitors resources and applications on AWS, on-premises, and on other clouds. Amazon CloudWatch collects and visualizes real-time logs, metrics, and event data in automated dashboards to streamline our infrastructure and application maintenance.

a) Hardware Requirements

No matter how many services AWS offers, you still require some amount of hardware to use the services. The amount of hardware you require when working with services in the cloud is minimal because the AWS hardware does all the heavy lifting. When working with services locally, you need additional hardware because AWS is no longer doing the heavy lifting for you. Therefore, you should consider different hardware requirements depending on where you host the AWS service.

AWS abstracts the underlying hardware, the hardware requirements for your local devices or infrastructure still play a role in effectively utilizing AWS services. Assessing and considering the hardware requirements based on your application's needs and where you host the AWS service is indeed an important consideration.

b) Software Requirements (Front End & Back End Tools)

1. AWS-security-assessment-solution/CloudFormation-Templates/Self ServiceSec.yml

AWSTemplateFormatVersion: 2010-09-09

Description: Cloudformation templates for

<https://github.com/aws-labs/aws-security-assessment-solution>

Parameters:

TemplateS3Bucket:

Description: The name of the bucket you created to upload the project files.

Type: String

Default: YourBucketNameHere

RansomwareChecks:

Description: "Enable Ransomware checks"

Type: String

Default: 'false'

AllowedValues:

- 'true'
- 'false'

CommonSecurityMistakesChecks:

Description: "Check for common security mistakes as per <https://www.youtube.com/watch?v=tmuCIE3nWIk> "

Type: String

Default: 'false'

AllowedValues:

- 'true'
- 'false'

Conditions:

This will enable the optional ransomware specific modules

EnableRansomwareChecks: !Equals

- !Ref RansomwareChecks
- 'true'

This will enable the optional common security mistake checks specific modules

EnableCommonSecurityMistakesChecks: !Equals

- !Ref CommonSecurityMistakesChecks
- 'true'

Resources:

Basic VPC required for the Security Review. This will create a VPC with 2 subnets in a single AZ. It will also create a NATGateway and a few VPC Endpoints.

rVPCStack:

Type: AWS::CloudFormation::Stack

Properties:

TemplateURL: !Join

- "
 - - 'https://'
 - !Ref TemplateS3Bucket
 - '.s3.'
 - !Ref "AWS::Region"
 - '.amazonaws.com/SelfServiceSecVPC.yml'
- # This will create a single Role for EC2 Instance, leverage a few AWS Managed Policies and a few custom policies so Prowler and ScoutSuite can run.

rIAMStack:

Type: AWS::CloudFormation::Stack

Properties:

TemplateURL: !Join

```

- "
- - 'https://'
- !Ref TemplateS3Bucket
- '.s3.'
- !Ref "AWS::Region"
- '.amazonaws.com/SelfServiceSecIAM.yml'
Parameters:
SelfServiceSecS3Bucket: !GetAtt
- rS3Stack
- Outputs.SelfServiceSecS3Bucket
#This will create a Bucket where the output of Prowler and
ScoutSuite will be delivered
rS3Stack:
Type: AWS::CloudFormation::Stack
Properties:
TemplateURL: !Join
- "
- - 'https://'
- !Ref TemplateS3Bucket
- '.s3.'
- !Ref "AWS::Region"
- '.amazonaws.com/SelfServiceSecS3.yml'
#This will create a single Instance using Amazon Linux 2, and
install/deploy Prowler and ScoutSuite
rEC2Stack:
Type: AWS::CloudFormation::Stack
Properties:
TemplateURL: !Join
- "
- - 'https://'
- !Ref TemplateS3Bucket
- '.s3.'
- !Ref "AWS::Region"
- '.amazonaws.com/SelfServiceSecEC2.yml'
Parameters:
SelfServiceSecVPCID: !GetAtt
- rVPCStack
- Outputs.SelfServiceSecVPCID
SubnetAID: !GetAtt
- rVPCStack
- Outputs.SubnetAID
InstanceRoleName: !GetAtt
- rIAMStack
- Outputs.InstanceRoleName
SelfServiceSecS3Bucket: !GetAtt
- rS3Stack
- Outputs.SelfServiceSecS3Bucket
#This will enable the optional ransomware checks
rRansomwareChecks:
Condition: EnableRansomwareChecks
Type: AWS::CloudFormation::Stack
Properties:
TemplateURL: !Join
- "
- - 'https://'

```

- !Ref TemplateS3Bucket
- '.s3.'
- !Ref "AWS::Region"
- '.amazonaws.com/SelfServiceSecRansomware.yml'

Parameters:

InstanceRoleARN: !GetAtt

- rIAMStack
- Outputs.InstanceRoleARN

TemplatesS3Bucket: !Ref TemplateS3Bucket

SelfServiceSecS3Bucket: !GetAtt

- rS3Stack
- Outputs.SelfServiceSecS3Bucket

#This will enable the optional checks for common security mistakes

rCommonSecurityMistakesChecks:

Condition: EnableCommonSecurityMistakesChecks

Type: AWS::CloudFormation::Stack

Properties:

TemplateURL: !Join

- "
- - 'https://'
- !Ref TemplateS3Bucket
- '.s3.'
- !Ref "AWS::Region"
- '.amazonaws.com/SelfServiceSecCommonSecurityMistakes.yml'

Parameters:

InstanceRoleARN: !GetAtt

- rIAMStack
- Outputs.InstanceRoleARN

TemplatesS3Bucket: !Ref TemplateS3Bucket

SelfServiceSecS3Bucket: !GetAtt

- rS3Stack
- Outputs.SelfServiceSecS3Bucket

Outputs:

ReportBucketName:

Description: The name of the newly generated S3 Bucket with reports from the tool

Value: !GetAtt rS3Stack.Outputs.SelfServiceSecS3Bucket

2. CloudFormation-Templates/SelfServiceSecEC2.yml

AWSTemplateFormatVersion: '2010-09-09'

Description: AWS CloudFormation template to launch an Instance for self service Security Review

Parameters:

SelfServiceSecVPCID:

Type: String

SubnetAID:

Type: String

SelfServiceSecS3Bucket:

Type: String

InstanceRoleName:

```

Type: String
LatestAmild:
Type: 'AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>'
Default: '/aws/service/ami-amazon-linux-latest/amzn2-ami-hvm-
x86_64-gp2'
Resources:
rInstanceSG:
Type: AWS::EC2::SecurityGroup
Properties:
GroupDescription: "SelfServiceSec Security Instance SG"
VpcId: !Ref SelfServiceSecVPCID
Tags:
- Key: Name
Value: "SelfServiceSecSecurityGroup"
rEC2Instance:
Type: AWS::EC2::Instance
Properties:
InstanceType: m5a.large
Tags:
- Key: Name
Value: "SelfServiceSecSecurityInstance"
SecurityGroupIds:
- !GetAtt "rInstanceSG.GroupId"
ImageId: !Ref 'LatestAmild'
SubnetId: !Ref SubnetAID
IamInstanceProfile: !Ref InstanceRoleName
# This will update the Instance, pull Prowler, run and then write to
the output to the S3 Bucket
UserData:
Fn::Base64: !Sub |
#!/bin/bash
yum -y update
yum groupinstall "Development Tools" -y
sudo yum -y install gcc openssl-devel bzip2-devel libffi-devel
wget https://www.python.org/ftp/python/3.9.16/Python-3.9.16.tgz
tar xzf Python-3.9.16.tgz
cd Python-3.9.16/
./configure --enable-optimizations
sudo make altinstall
export PATH=$PATH:/usr/local/bin
pip3.9 install prowler
cd
aws configure set default.region ${AWS::Region}
prowler aws | tee prowler-output.txt
aws s3 cp prowler-output.txt s3://${SelfServiceSecS3Bucket}
aws s3 cp output s3://${SelfServiceSecS3Bucket} --recursive
/sbin/init 0

```

3. CloudFormation-Templates/SelfServiceSecS3.yml

AWSTemplateFormatVersion: 2010-09-09
Description: Cloudformation templates to create a new Bucket for self service Security Reviews
Resources:
Create a random name S3 bucket for the outputs to be placed
rCentralizedBucket:
DeletionPolicy: Delete
Type: 'AWS::S3::Bucket'
Properties:
BucketEncryption:
ServerSideEncryptionConfiguration:
- ServerSideEncryptionByDefault:
SSEAlgorithm: AES256
Outputs:
SelfServiceSecS3Bucket:
Description: 'Bucket Name'
Value: !Ref rCentralizedBucket

4. CloudFormation-Templates/SelfServiceSecRansomware.yml

AWSTemplateFormatVersion: 2010-09-09
Description: Cloudformation templates to create ransomware checks in the account. <https://github.com/awslabs/aws-security-assessment-solution/>
Parameters:
InstanceRoleARN:
Type: String
SelfServiceSecS3Bucket:
Type: String
TemplatesS3Bucket:
Type: String
Conditions:
CreateGovCloudResources: !Equals [!Ref "AWS::Partition", aws]
Currently two features are not supported in GovCloud so we omit these two checks below with this partition check.

Resources:
libs:
Type: AWS::Lambda::LayerVersion
Properties:
LayerName: boto3
Description: Dependencies for the ransomware modules
Content:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/boto3_layer.zip"
CompatibleRuntimes:
- python3.9
- python3.8
- python3.7

rLambdaCheckForLogging:
Type: "AWS::Lambda::Function"
Properties:

Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/check_logging.py.zip"
FunctionName: "LambdaCheckForLogging"
Handler: "lambda_function.lambda_handler"
MemorySize: 256
Layers:
- !Ref libs
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 900
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForLogging:
Type: AWS::CloudFormation::CustomResource
Properties:
ServiceToken: !GetAtt rLambdaCheckForLogging.Arn

rLambdaCheckForRoute53DNSFirewall:
Type: "AWS::Lambda::Function"
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key:
"modules/RansomwareDetection/check_route53_dns_firewall.py.zip"
FunctionName: "LambdaCheckForRoute53DNSFirewall"
Handler: "lambda_function.lambda_handler"
MemorySize: 128
Layers:
- !Ref libs
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 600
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForRoute53DNSFirewall:
Type: AWS::CloudFormation::CustomResource
Properties:
ServiceToken: !GetAtt rLambdaCheckForRoute53DNSFirewall.Arn

rLambdaCheckForUnusedAccessKeys:
Type: "AWS::Lambda::Function"
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket

S3Key:
"modules/RansomwareDetection/check_unused_access_keys.py.zip"
FunctionName: "LambdaCheckForUnusedAccessKeys"
Handler: "lambda_function.lambda_handler"
MemorySize: 128
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 600
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForUnusedAccessKeys:
Type: AWS::CloudFormation::CustomResource
Properties:
ServiceToken: !GetAtt rLambdaCheckForUnusedAccessKeys.Arn

rLambdaCheckForDNSSEC:
Type: "AWS::Lambda::Function"
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/check_dnssec.py.zip"
FunctionName: "LambdaCheckForDNSSEC"
Handler: "lambda_function.lambda_handler"
MemorySize: 128
Layers:
- !Ref libs
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 600
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForDNSSEC:
Type: AWS::CloudFormation::CustomResource
Properties:
ServiceToken: !GetAtt rLambdaCheckForDNSSEC.Arn

rLambdaCheckFor2k8:
Type: "AWS::Lambda::Function"
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/check_for_outdated_os.py.zip"
FunctionName: "LambdaCheckFor2k8"

Handler: "lambda_function.lambda_handler"
MemorySize: 128
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 90
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckFor2k8:
Type: AWS::CloudFormation::CustomResource
Properties:
ServiceToken: !GetAtt rLambdaCheckFor2k8.Arn

rLambdaCheckForBackups:
Type: "AWS::Lambda::Function"
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/check_backup.py.zip"
FunctionName: "LambdaCheckForBackups"
Handler: "lambda_function.lambda_handler"
MemorySize: 128
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 90
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForBackups:
Type: AWS::CloudFormation::CustomResource
Properties:
ServiceToken: !GetAtt rLambdaCheckForBackups.Arn

rLambdaCheckForGuardDuty:
Type: "AWS::Lambda::Function"
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/check_guarddduty.py.zip"
FunctionName: "LambdaCheckForGuardDuty"
Handler: "lambda_function.lambda_handler"
MemorySize: 128
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 90
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:

Mode: "PassThrough"

Environment:

Variables:

reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForGuardDuty:

Type: AWS::CloudFormation::CustomResource

Properties:

ServiceToken: !GetAtt rLambdaCheckForGuardDuty.Arn

rLambdaCheckForEBSSnapshot:

Type: "AWS::Lambda::Function"

Properties:

Code:

S3Bucket: !Ref TemplatesS3Bucket

S3Key: "modules/RansomwareDetection/check_ebs_snapshots.py.zip"

FunctionName: "LambdaCheckForEBSSnapShots"

Handler: "lambda_function.lambda_handler"

MemorySize: 128

Role: !Ref InstanceRoleARN

Runtime: "python3.8"

Timeout: 90

DeadLetterConfig: !Ref "AWS::NoValue"

TracingConfig:

Mode: "PassThrough"

Environment:

Variables:

reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForEBSSnapshot:

Type: AWS::CloudFormation::CustomResource

Properties:

ServiceToken: !GetAtt rLambdaCheckForEBSSnapshot.Arn

rLambdaCheckForS3StaleAccess:

Type: "AWS::Lambda::Function"

Condition: CreateGovCloudResources

Properties:

Code:

S3Bucket: !Ref TemplatesS3Bucket

S3Key: "modules/RansomwareDetection/check_s3_stale_access.py.zip"

FunctionName: "LambdaCheckForS3StaleAccess"

Handler: "lambda_function.lambda_handler"

MemorySize: 128

Role: !Ref InstanceRoleARN

Runtime: "python3.8"

Timeout: 180

DeadLetterConfig: !Ref "AWS::NoValue"

TracingConfig:

Mode: "PassThrough"

Environment:

Variables:

reportbucket: !Ref SelfServiceSecS3Bucket

xLambdaCheckForS3StaleAccess:

```

Type: AWS::CloudFormation::CustomResource
Condition: CreateGovCloudResources
Properties:
ServiceToken: !GetAtt rLambdaCheckForS3StaleAccess.Arn

rLambdaCheckForSSMUse:
Type: "AWS::Lambda::Function"
Condition: CreateGovCloudResources
Properties:
Code:
S3Bucket: !Ref TemplatesS3Bucket
S3Key: "modules/RansomwareDetection/check_ssm_used.py.zip"
FunctionName: "LambdaCheckForSSMUse"
Handler: "lambda_function.lambda_handler"
MemorySize: 128
Role: !Ref InstanceRoleARN
Runtime: "python3.8"
Timeout: 90
DeadLetterConfig: !Ref "AWS::NoValue"
TracingConfig:
Mode: "PassThrough"
Environment:
Variables:
reportbucket: !Ref SelfServiceSecS3Bucket
xLambdaCheckForSSMUse:
Type: AWS::CloudFormation::CustomResource
Condition: CreateGovCloudResources
Properties:
ServiceToken: !GetAtt rLambdaCheckForSSMUse.Arn

```

c) Operating System Requirements

Amazon Web Services (AWS), gives us the flexibility to choose the operating system (OS) that best suits our needs when deploying and managing our infrastructure. AWS supports a wide range of operating systems, including both Linux and Windows variants. We can choose any operating system in AWS based on our usage requirements & necessities.

1.4 Objectives of the Project

Ransomware has been a prominent threat to enterprises since the late 1990s. In 2020, the FBI's IC3 (Internet Crime Complaint Centre) received 2,474 complaints identified as ransomware with adjusted losses of over \$29.1 million. Those complaints, however, represent only the attacks reported to IC3. The actual number of ransomware attacks and costs are much higher.

Cloud security at AWS is the highest priority. AWS has technology that helps customers protect and recover their systems from ransomware attacks, including services and features that provide infrastructure and data backups, the AWS-Well Architected Tool, and strong security controls using AWS security services. In this workshop, we will learn some of the basics of how to protect, detect, and respond to ransomware using AWS Services.

2. Literature Survey / Related Works

Security and compliance resources

- Security and compliance resources AWS Security Reference Architecture (AWS SRA)
- The European Union Agency for Cybersecurity (ENISA)
- U.S. Cybersecurity & Infrastructure Security Agency (CISA)

2.1 Literature Review

- ❖ Ermetic Whitepaper: Misconfigurations Leading to AWS S3 Ransomware Exposure
<https://l.ermetic.com/wp-aws-s3-ransomware-exposure-report>
- ❖ Securing your AWS Cloud environment from ransomware
https://d1.awsstatic.com/WWPS/pdf/AWSPS_ransomware_ebook_Apr-2020.pdf
- ❖ Assess your security posture to identify and remediate security gaps susceptible to ransomware
https://aws.amazon.com/blogs/publicsector/assess-your-security-posture-identify-remediate-security-gaps-ransomware/?sc_channel=SM&sc_campaign=Webinar_2021_vid&sc_publisher=YouTube&sc_medium=video&sc_content=Video10331&sc_detail=JmWbiCLqLag%20&sc_country=US
- ❖ Protecting against ransomware
<https://aws.amazon.com/security/protecting-against-ransomware/>
- ❖ Ransomware recovery AWS Elastic Disaster Recovery | Ransomware Recovery | AWS (amazon.com)
- ❖ Best Practices for Security, Identity, & Compliance Security, Identity & Compliance | AWS Architecture Centre (amazon.com)
- ❖ AWS Security Incident Response Guide
https://d1.awsstatic.com/whitepapers/aws_security_incident_response.pdf
- ❖ Classic Intrusion Analysis Frameworks for AWS Environments: Application and Enhancement Classic Intrusion Analysis Frameworks for AWS Environments: Application and Enhancement - AWS Whitepapers (amazon.com)
- ❖ Base paper
<https://aws.amazon.com/blogs/apn/protect-detect-and-respond-to-ransomware-with-presidiosransomware-mitigation-kit/>
- ❖ AWS White Paper Security Pillar - AWS Well-Architected Framework - Security Pillar (amazon.com)

- ❖ Introducing AWS Backup Audit Manager
<https://aws.amazon.com/about-aws/whats-new/2021/08/aws-backup-audit-manager/>
- ❖ <https://n2ws.com/blog/aws-disaster-recovery/how-to-laugh-at-ransomware-cyber-criminals-with-aws-backup>
- ❖ The anatomy of ransomware event targeting data residing in Amazon S3
<https://aws.amazon.com/blogs/security/anatomy-of-a-ransomware-event-targeting-data-in-amazon-s3/>

2.2 Existing Systems

Ensuring full security in the cloud can be challenging due to several factors:

- Complexity: Cloud environments are complex and dynamic, often involving numerous interconnected services, virtual machines, containers, and networking components. Managing security across such a dynamic and distributed infrastructure requires a deep understanding of the cloud provider's offerings, security configurations, and the potential risks associated with each component.
- Shared Responsibility: Cloud service providers like AWS operate on a shared responsibility model, where they provide security of the underlying infrastructure, while customers are responsible for securing their applications, data, and user access. This shared responsibility can lead to potential gaps or misconfigurations if customers fail to understand and implement their part of the security responsibilities correctly.
- Misconfigurations: Misconfigurations are one of the most common causes of security incidents in the cloud. Due to the self-service nature of cloud environments, misconfigurations can occur at various levels, including access controls, network configurations, storage settings, and encryption. A single misconfiguration can lead to data exposure, unauthorized access, or other security vulnerabilities.
- Rapid Development and Deployment: Cloud environments allow for rapid development, deployment, and scaling of applications. While this agility brings significant benefits, it can also increase the likelihood of security oversights. Fast-paced development cycles and the need to quickly provision resources may result in security controls being overlooked or bypassed.
- Evolving Threat Landscape: The threat landscape is constantly evolving, with new attack vectors, vulnerabilities, and techniques emerging regularly. Cloud environments are not immune to these threats, and security measures must continually adapt and evolve to counter new risks. Staying up to date with the latest security best practices, monitoring for emerging threats, and promptly applying security patches and updates are essential but can be challenging to manage effectively.

- **Insider Threats and User Errors:** While cloud providers have robust security measures in place, insider threats and user errors remain significant concerns. Malicious insiders or compromised user accounts can bypass security controls and gain unauthorized access to data or systems. Additionally, accidental user errors, such as misconfigured permissions or accidental data exposure, can also result in security breaches.

To address these challenges, organizations need to adopt a holistic and layered approach to cloud security. This includes implementing strong access controls, regularly auditing and monitoring configurations, employing encryption and secure network architectures, conducting regular security assessments, and providing ongoing security training to users and administrators. Collaborating with cloud providers, leveraging their security tools and services, and working with experienced security professionals can also help enhance the overall security posture in the cloud.

It's important to have a comprehensive security strategy that combines the built-in security measures provided by AWS with additional layers of defence, including regular backups, user education, monitoring, and incident response plans tailored to ransomware threats.

2.3 Proposed System-Planning

Let's consider a business case of a famous café which is expanding its business by venturing into online order delivery service. This Café is doing really well in their business & the management decides to expand their business in the horizon of online domain to explore the limitless possibilities assisted by technology.



Fig: Representative image of a café business that is planning to go online operationally.

The café is migrating its online system into the cloud now.



Fig: Representational image of the café migrating its online system into the cloud.

Eventually after their huge success and brand reputation the Café decides to go global offering franchise on demand to willing partnership seekers.



Fig: Representational image of a business going global using AWS cloud services.

2.4 Motivation for Implementation:

Ransomware has been a prominent threat to enterprises since the late 1990s. In 2020, the FBI's IC3 (Internet Crime Complaint Centre) received 2,474 complaints identified as ransomware with adjusted losses of over \$29.1 million. Those complaints, however, represent only the attacks reported to IC3. The actual number of ransomware attacks and costs are much higher.

Protecting your cloud system from ransomware is crucial for several reasons:

- **Data Loss and Downtime:** Ransomware attacks can encrypt or delete your data, rendering it inaccessible or permanently lost. This can result in significant business disruption, operational downtime, and potential financial losses. Protecting your cloud system helps safeguard your critical data and minimize the risk of data loss or extended periods of service unavailability.
- **Financial Impact:** Ransomware attacks often come with a demand for a ransom payment to restore access to your data or systems. Paying the ransom is not recommended, as it encourages criminal activity and does not guarantee recovery. By protecting your cloud system against ransomware, you reduce the chances of falling victim to such attacks and mitigate the potential financial impact.
- **Reputation and Customer Trust:** A ransomware attack can tarnish your organization's reputation and erode customer trust. If your customers' sensitive data is compromised or inaccessible due to a ransomware incident, they may lose confidence in your ability to protect their information. Protecting your cloud system demonstrates your commitment to data security and helps maintain trust with your customers.
- **Compliance Requirements:** Many industries have strict regulatory requirements concerning data protection and security, such as the General Data Protection Regulation (GDPR) in the European Union or the Health Insurance Portability and Accountability Act (HIPAA) in the healthcare sector. Failure to adequately protect your cloud system from ransomware attacks can lead to compliance violations, legal consequences, and potential financial penalties.
- **Business Continuity:** Ransomware attacks can disrupt your business operations, causing significant disruptions and delays. By implementing robust security measures for your cloud system, you ensure business continuity and minimize the impact of potential ransomware incidents. This includes having proper backup and disaster recovery plans in place to recover data and systems quickly.
- **Prevention of Data Theft:** In some cases, ransomware attacks may be a cover for data theft. Attackers may exfiltrate sensitive information before encrypting it and use the threat of exposure as leverage to extort a ransom payment. Protecting your cloud system from ransomware reduces the risk of unauthorized data access and theft.

3. System Design Methodology

3.1 System Architecture Diagram

The chosen architecture suits a restaurant or cafeteria type large-scale business model.

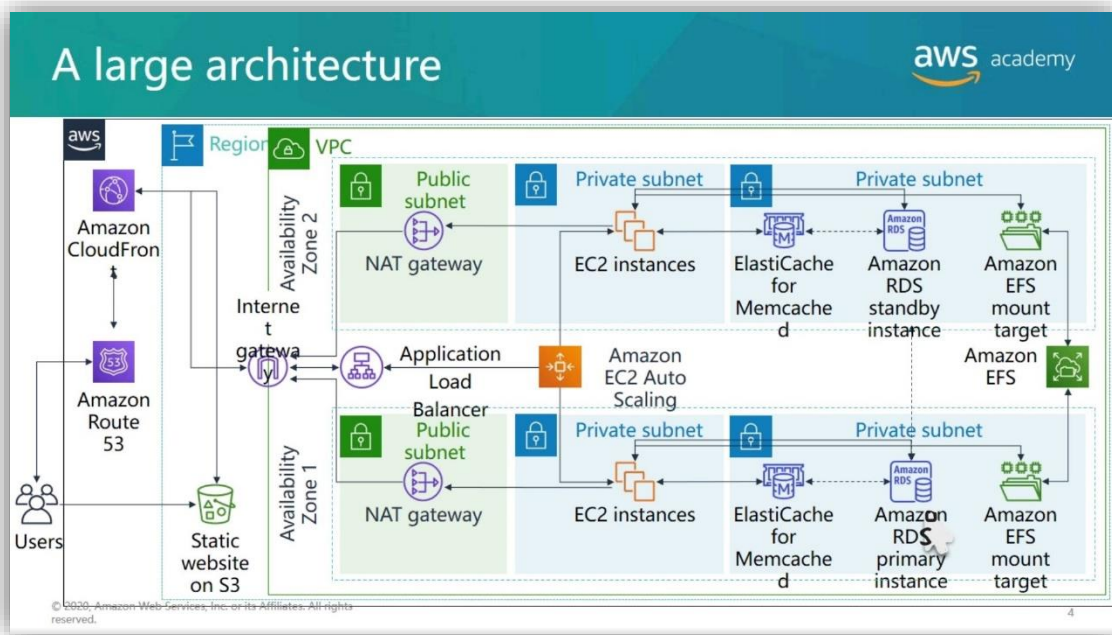


Fig: A representational large-scale architecture.

3.2 Description of work

Detecting ransomware in an Amazon Web Services (AWS) environment requires a multi-layered approach that combines proactive security measures, real-time monitoring, and incident response capabilities. Here are some steps you can take to detect and mitigate ransomware in AWS:

1. **Implement strong access controls:** Ensure that you follow the principle of least privilege, granting users and services only the permissions they require to perform their tasks. Regularly review and update access policies to prevent unauthorized access to critical resources.
2. **Enable CloudTrail:** AWS CloudTrail provides detailed logs of API activity within your AWS account. By enabling CloudTrail, you can track and monitor actions taken on your resources, including changes made by potential ransomware attacks.
3. **Enable AWS GuardDuty:** It is a threat detection service that uses machine learning algorithms to analyse AWS account activity and identify malicious behaviour. It can help detect potential indicators of ransomware activity, such as unusual API calls or compromised instances.

4. **Implement AWS Config Rules:** AWS Config allows you to define and enforce rules for resource configurations. Create custom Config Rules that check for specific ransomware-related configurations, such as public access to critical storage buckets or unusual EC2 instance configurations.
5. **Use AWS Security Hub:** Security Hub provides a centralized view of security alerts and findings across your AWS accounts. It aggregates findings from multiple security services, including GuardDuty and AWS Config, and provides a comprehensive overview of your security posture.
6. **Monitor VPC Flow Logs:** Enable VPC Flow Logs to capture network traffic metadata. Analyse the logs for any suspicious or unexpected network traffic patterns that could indicate ransomware activity.
7. **Implement threat intelligence feeds:** Utilize threat intelligence feeds to enhance your security monitoring. These feeds provide information about known malicious IP addresses, domains, and file hashes associated with ransomware attacks. You can use AWS services like AWS Lambda and AWS Systems Manager to automate the ingestion and analysis of threat intelligence data.
8. **Implement anomaly detection:** Leverage AWS services like Amazon Macie or third-party tools to identify anomalous behaviour in your AWS environment. Look for unusual file access patterns, encryption activities, or unexpected changes in data behaviour that may indicate the presence of ransomware.
9. **Regularly patch and update:** Keep your AWS resources, including operating systems, applications, and security software, up to date with the latest patches and updates. This helps protect against known vulnerabilities that ransomware may exploit.
10. **Prepare an incident response plan:** Develop an incident response plan that outlines the steps to be taken in the event of a ransomware attack. Define roles and responsibilities, establish communication channels, and practice incident response scenarios regularly.
11. **AWS Backup:**
Centrally manage and automate data protection. It's a cost-effective, fully managed, policy-based service that simplifies data protection at scale.
12. **AWS Backup Vault Lock:**
It enforces a *write-once, read-many* (WORM) setting for all the backups you store and create in a backup vault. With AWS Backup Vault Lock, you can add an additional layer of defence that protects backups (recovery points) in your backup vaults from inadvertent or malicious delete operations and updates.

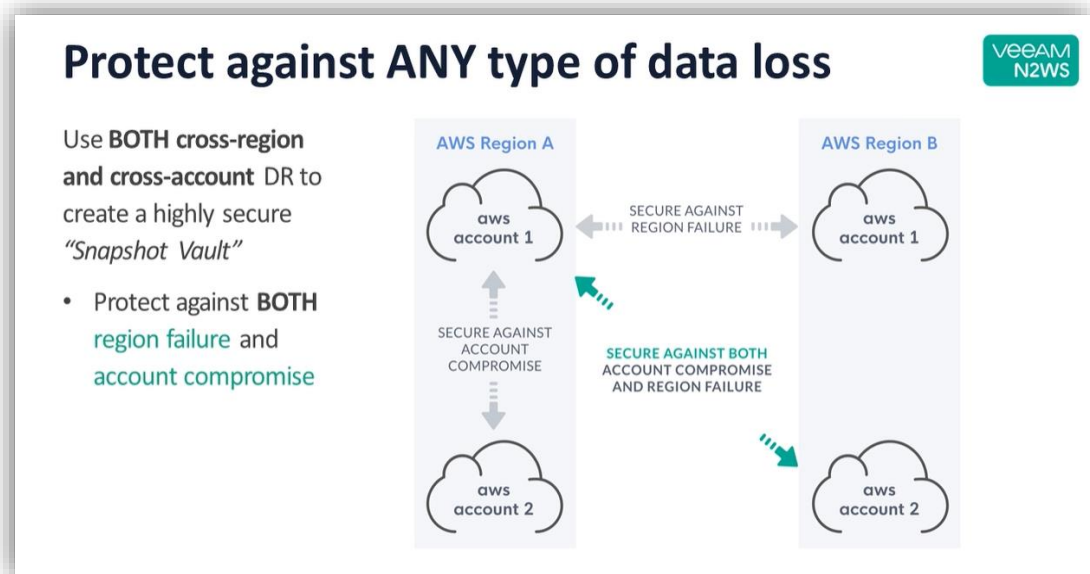


Fig: Pictorial representation of Cross Region & Cross account DR

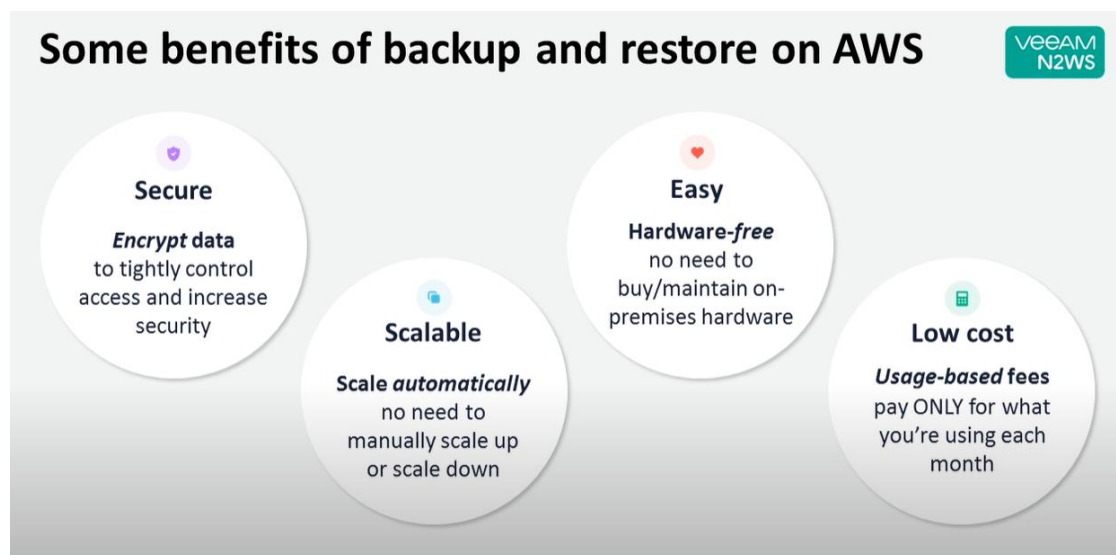


Fig: Benefits of backup & restore in AWS

3.3 Some Usable Algorithms in the project

Fraud Detection Algorithms used in AWS:

AWS Fraud Detector uses algorithms such as random cut forests, gradient boosting, and neural networks to detect and prevent fraud in real-time.

There are several algorithms and techniques that can be employed to detect ransomware in an AWS (Amazon Web Services) environment.

Here are a few approaches that can be used:

- **Anomaly Detection:** Implement anomaly detection algorithms to identify unusual patterns of behaviour or file access. Ransomware often exhibits abnormal behaviour, such as rapid encryption of files or accessing files outside the normal user behaviour. Anomaly detection algorithms can help detect such activities and trigger alerts.
- **Machine Learning (ML) Models:** Train ML models on historical data to identify patterns and characteristics of ransomware. These models can analyse file access patterns, system calls, network traffic, and other relevant data to detect potential ransomware activities. ML algorithms such as Random Forest, Support Vector Machines, or Neural Networks can be used for this purpose.
- **Signature-based Detection:** Maintain a signature database of known ransomware strains and use signature-based detection algorithms to scan files and processes for matching signatures. This approach is effective for detecting known ransomware variants but may not be as effective against new or evolving strains.
- **Behavioural Analysis:** Monitor system and application behaviour for signs of ransomware activity. This can include analysing system logs, file access patterns, privilege escalation attempts, and suspicious network traffic. By establishing a baseline of normal behaviour, deviations can be detected and flagged as potential ransomware activities.
- **File Integrity Monitoring (FIM):** Employ FIM tools or algorithms to monitor file integrity and detect unauthorized changes or encryption. Ransomware typically modifies or encrypts files, and FIM algorithms can detect these changes and trigger alerts or actions.
- **Heuristic Analysis:** Develop heuristic algorithms that analyse file behaviour, execution patterns, and system interactions to identify potential ransomware activity. Heuristics can detect ransomware based on its behaviour rather than relying on specific signatures.
- **CloudTrail Analysis:** Leverage AWS CloudTrail logs, which provide a record of AWS API activity, to monitor for suspicious activities related to ransomware. Analysing CloudTrail logs can help identify unauthorized access attempts, unusual API calls, or changes to security policies.

It's important to note that combining multiple detection techniques and continuously updating and refining the algorithms is crucial for effective ransomware detection in an AWS environment. Additionally, proactive security measures such as regular backups, strong access controls, and employee awareness and training are also vital to minimize the impact of ransomware attacks.

4.Implementation / Results

4.1 Experimental Setup

Detecting ransomware in AWS (Amazon Web Services) requires a comprehensive approach that involves monitoring various aspects of your environment. Here is an experimental setup that you can use to detect ransomware in AWS:

1. **One Time Check:** A program file to do the one-time check is uploaded to the S3 bucket. The S3 URL of this bucket is used along with AWS Cloud Formation by creating a stack. It creates many resources using IaC which checks for Aws core services if they are enabled, it checks if data protection is enabled, if EBS volumes are running on no snapshots, if they are running on outdated OS, if EC2 instances management is not done properly
2. **AWS CloudTrail Logging: Enable** AWS CloudTrail logging to capture all API calls and activities within your AWS account. This includes activities related to EC2 instances, S3 buckets, security groups, and other resources. By monitoring CloudTrail logs, you can detect any suspicious activities that may indicate ransomware activity, such as unexpected modifications or deletions of resources.
3. **IAM Access Analyser:** It provides automated reasoning and identifies potential security risks by continuously analysing resource policies and access control lists (ACLs) associated with your AWS Identity and Access Management (IAM) roles, Amazon S3 buckets, and AWS KMS keys.
4. **Amazon GuardDuty:** Activate Amazon GuardDuty, a threat detection service that uses machine learning and anomaly detection algorithms to identify potential security issues. GuardDuty can analyse events from CloudTrail logs, VPC Flow Logs, and DNS logs to detect known ransomware behaviours, unauthorized access attempts, or data exfiltration attempts.
5. **VPC Flow Logs:** Enable VPC Flow Logs for your AWS Virtual Private Cloud (VPC) to capture network traffic metadata. Analysing VPC Flow Logs can help identify unusual network communication patterns or large data transfers, which might indicate ransomware activities.
6. **AWS Config Rules:** Utilize AWS Config Rules to monitor compliance and detect changes to critical security settings. You can create custom rules to check for specific ransomware indicators, such as changes to EC2 instance configurations or unauthorized modifications to security groups.
7. **Amazon Macie:** Enable Amazon Macie, a data classification and security service that uses machine learning to automatically discover, classify, and protect sensitive data stored in AWS. Macie can help identify potential ransomware targets by monitoring data access patterns, unusual file modifications, or encryption of sensitive files.
8. **AWS Security Hub:** It acts as a central hub for aggregating, organizing, and prioritizing security alerts and findings from various AWS services, as well as from third-party security tools. With AWS Security Hub, you can gain insights into your security and compliance status by analyzing data from AWS services

such as Amazon GuardDuty, Amazon Inspector, Amazon Macie, and AWS Firewall Manager.

9. **AWS Backup:** AWS Backup is a fully managed backup service provided by Amazon Web Services (AWS) that simplifies the process of protecting your data and applications stored in AWS. It allows you to centrally manage and automate backup tasks across multiple AWS services, ensuring the durability, availability, and integrity of your backups.

10. AWS Backup Vault Lock:

AWS Backup Vault Lock adds an additional layer of defence that protects backups (recovery points) in the backup vaults from inadvertent or malicious delete operations and updates that shorten or otherwise alter their retention period. It also helps in enforcing retention periods, prevents early deletions by privileged users (including the AWS account root user), and meets organization's data protection policies and procedures.

11. **Security Information and Event Management (SIEM) Integration:** Integrate your AWS environment with a SIEM solution, such as Splunk, Elastic Stack (ELK), or AWS-native services like AWS Security Hub or AWS CloudWatch. Configure log ingestion from various AWS services to the SIEM, and create custom detection rules or use pre-built threat intelligence feeds to identify ransomware-related activities.
12. **File Integrity Monitoring (FIM):** Deploy FIM tools or agents on your EC2 instances to monitor file system changes. These tools can alert you if there are unexpected modifications or encryption of files, which are typical behaviours of ransomware. Tools like OSSEC, Tripwire, or AWS-native services like Amazon Inspector can help with FIM.
13. **Endpoint Detection and Response (EDR):** Implement an EDR solution like AWS-native Amazon Detective or third-party solutions to monitor the behaviour of your EC2 instances. EDR tools can detect suspicious activities, such as abnormal process execution, network connections, or changes to system configurations, which might indicate ransomware activity.
14. **Regular Vulnerability Scanning:** Conduct regular vulnerability scanning of your AWS resources using tools like AWS Inspector or third-party vulnerability scanners. This can help identify potential weaknesses or misconfigurations that could be exploited by ransomware attackers.
15. **User Behaviour Analytics (UBA):** Utilize UBA tools or services to analyse user behaviour within your AWS environment. Unusual user activities, such as access from unfamiliar locations or at unusual times, might indicate compromised credentials or malicious activities related to ransomware.
16. **Incident Response and Automation:** Establish an incident response plan specific to ransomware incidents. Define procedures for containing and eradicating ransomware, as well as for recovering affected systems and data. Leverage automation tools like AWS Lambda or AWS Systems Manager Automation to streamline and accelerate incident response processes.

The below-mentioned experimental setup should be supplemented with the above-given good security practices, such as enforcing strong access controls, regular backups, and employee security awareness training. Additionally, staying up to date with the latest security threats and patching vulnerabilities promptly is crucial for maintaining a secure AWS.

1.Protect with AWS Systems Manager:

Reducing the probability of a ransomware incident is top of mind for information security professionals. Attack surface reduction and vulnerability management are key elements of a successful security program. In this builder session, we will look at how AWS Systems Manager helps with systems hardening via patch management and configuration management.

AWS Systems Manager helps maintain security and compliance by scanning your instances against your patch, configuration, and custom policies. You can define patch baselines, maintain up-to-date anti-virus definitions, and enforce firewall policies. You can also remotely manage your servers at scale without manually logging in to each server. Systems Manager also provides a centralized store to manage your configuration data, whether it's plain text, such as database strings, or secrets, such as passwords. AWS Systems Manager can also be helpful in an incident response scenario if your traditional server management infrastructure becomes unavailable.

Patching

Systems patching is the most basic component of a vulnerability management program. (Note: While we do not cover it as a part of this builder session, AWS Inspector can help you with vulnerability scanning in AWS) AWS Systems Manager gives you the ability to deploy patches to your EC2 instances and on-premises systems. You can even patch systems that are isolated from the internet or the rest of your network by creating a VPC endpoint.

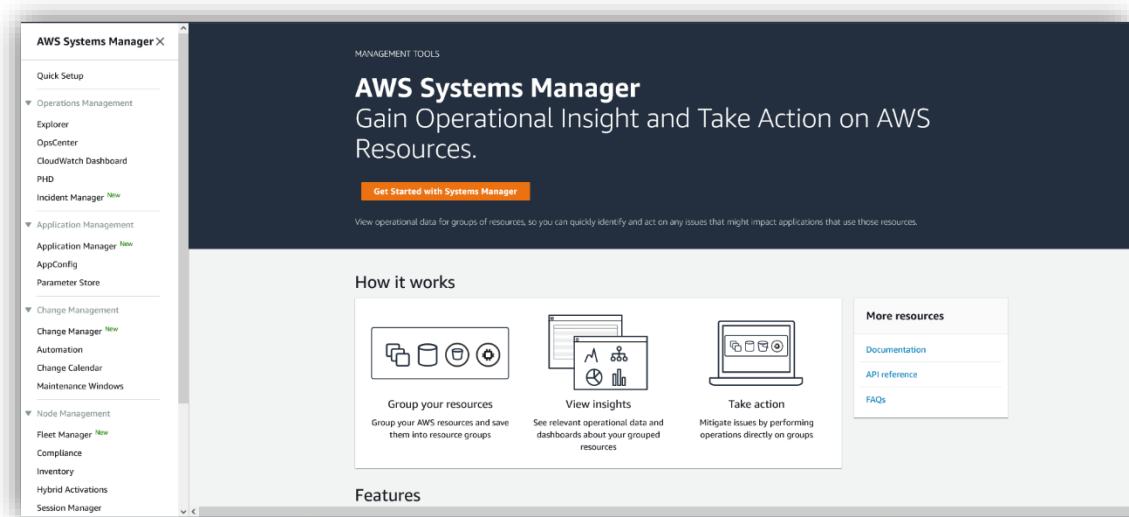
AWS Systems Manager Agent (SSM Agent) is preinstalled, by default, on the following Amazon Machine Images (AMIs):

- Amazon Linux
- Amazon Linux 2
- Amazon Linux 2 ECS-Optimized Base AMIs
- macOS 10.14.x (Mojave), 10.15.x (Catalina), and 11.x (Big Sur)
- Ubuntu Server 16.04, 18.04, and 20.04
- Windows Server 2008-2012 R2 AMIs published in November 2016 or later
- Windows Server 2016 and 2019

The focus of this task will be on Windows.

(See documentation for more information if you need to manually install the SSM agent)



- In AWS Management Console – go to Systems Manager



- Go to Systems Manager > Fleet Manager

Fleet Manager is a capability of AWS Systems Manager, is a unified user interface (UI) experience that helps you remotely manage your server fleet running on AWS, or on premises. With Fleet Manager, you can view the health and performance status of your entire server fleet from one dashboard. You can also gather data from individual instances to perform common troubleshooting and management tasks from the console.

Review the two instances enrolled for management:

Instance ID	Instance name	SSM Agent ping status	Operating System	SSM Agent version
	WebServer2	Online	Microsoft Windows Server 2019 Datacenter	3.0.1124.0
	WebServer1	Online	Microsoft Windows Server 2019 Datacenter	3.0.1124.0

- Go to Systems Manager > Patch Manager

Patch Manager, a capability of AWS Systems Manager, automates the process of patching managed instances with both security related and other types of updates. You can use Patch Manager to apply patches for both operating systems and applications. You can use Patch Manager to install Service Packs on Windows instances and perform minor version upgrades on Linux instances.

- Click on “View Predefined patch baselines”

predefined patch baselines.

[View predefined patch baselines](#)

- Click on the search bar and select “Operating System” and “Windows” as the filter
- Click on Default Windows patch baseline > AWS-DefaultPatchBaseline

pb-[REDACTED]	AWS-DefaultPatchBaseline	Default Patch Baseline Provided by AWS.	Windows	✔ Yes
---------------	--------------------------	--	---------	-------

Review approval rules

- Click on “Actions” and set it as the default patch baseline.

Edit Delete Actions ▲

Set default patch baseline

Modify patch groups

Set default patch baseline

Baseline ID

arn:aws:ssm:us-east-1:[REDACTED]:patchbaseline/pb-09ca3fb51f0412ec3

Baseline name

AWS-DefaultPatchBaseline

Operating system

Windows Server

Is default

Yes

Cancel

Set default

- Click on “Set default”
- Go back to Patch Manager and click “Patch Now”

Patch Manager provides options to scan your instances and report compliance on a schedule, install available patches on a schedule, and patch or scan instances on demand whenever you need to. You can also generate patch compliance reports that are sent to an Amazon Simple Storage Service (Amazon S3) bucket of your choice. You can generate one-time reports, or generate reports on a regular schedule. For a single instance, reports include details of all patches for the instance. For a report on all instances, only a summary of how many patches is missing is provided.

- Under Patching Operations, select “Scan”
- Under Instances to patch click on "Patch only the target instances I specify"
- Under Target Selection click on “Choose instances manually” – and select the two web servers

Patch instances now

Basic configuration

Scan for missing patches or install patches, with or without rebooting. For more patching options, use the [Configure patching](#) page.

Patching operation

☒ Scan

☐ Scan and install

Instances to patch

Choose whether to patch all instances or only the instances you specify

☐ Patch all instances

☒ Patch only the target instances I specify

Target selection

Choose a method for selecting targets.

☐ Specify instance tags

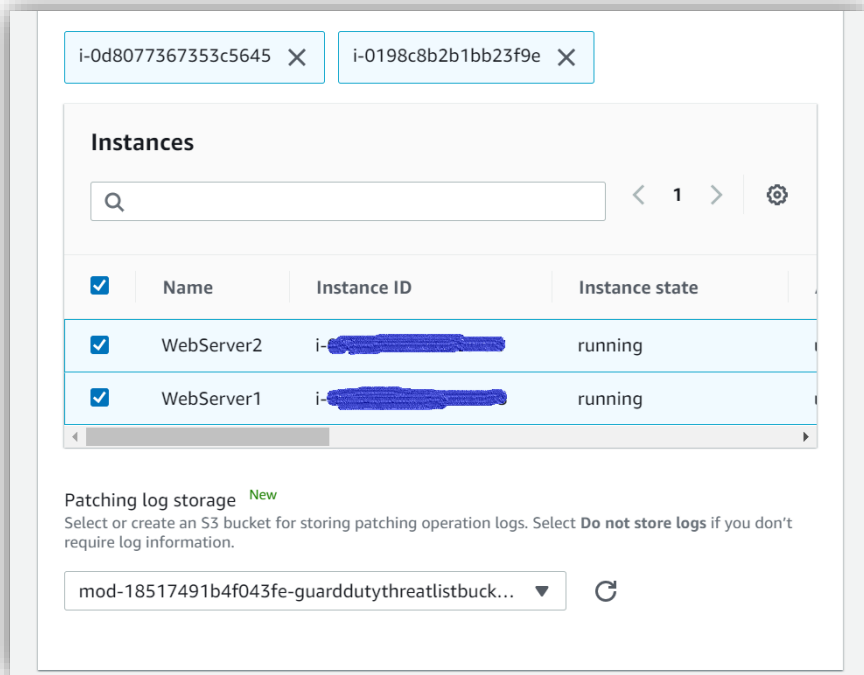
Specify one or more tag key-value pairs to select instances that share those tags.

☒ Choose instances manually

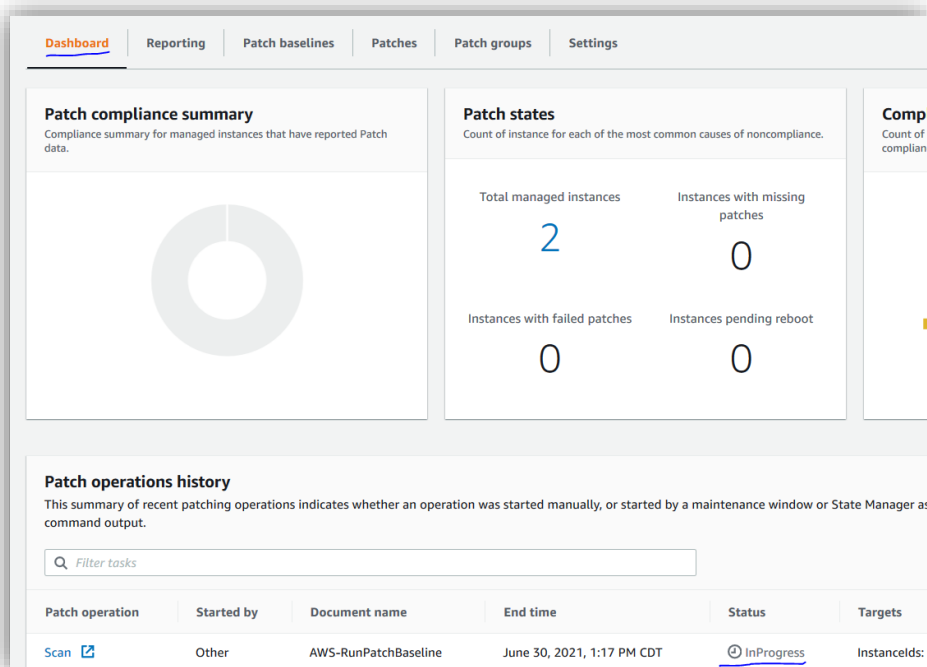
Manually select the instances you want to register as targets.

☐ Choose a resource group

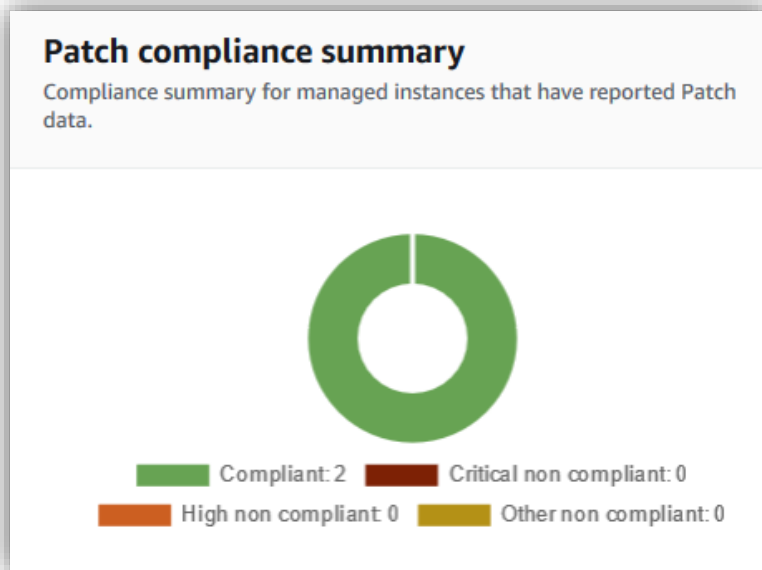
Choose a resource group that includes the resources you want to target.



- Click on “Patch now”
- Click on “Patch Manager” and scroll down on the dashboard and look at Patch Operation History to see the scan in progress



- Once the status turns to “Success”, scroll up to see Patch compliance summary



The scan will complete and show both instances compliant. If missing patches are identified – you can go back through the process and this time choose to install the missing patches.

Let's schedule regular patching to keep our systems up to date – click on “Configure Patching”

- Under Instances to Patch, click on “Select instances manually” – and select the two web servers
- Under Patching Schedule, Select “Schedule in a new Maintenance Window”
- Select “Use a CRON schedule builder”
- Select the maintenance window to run every Sunday at 2:00 and set the duration to last 3 hours

How do you want to specify a patching schedule?

☐ Select an existing Maintenance Window

☒ Schedule in a new Maintenance Window

☐ Skip scheduling and patch instances now

How do you want to specify a Maintenance Window schedule?

☒ Use a CRON schedule builder

☐ Use rate schedule builder

☐ Enter a CRON/Rate expression

Maintenance Window run frequency

☐ Every 12 hours

☒ Every at

Maintenance Window duration

Maximum number of hours to allow a Maintenance Window to run.

Enter a number between 1 and 24

Maintenance Window name

Enter a name between 3 and 128 characters. Valid characters include: a-z, A-Z, 0-9, and _.-

Explore other scheduling options

- Select “Scan and install” under Patching operation
- Click “Configure Patching” at the bottom of the page

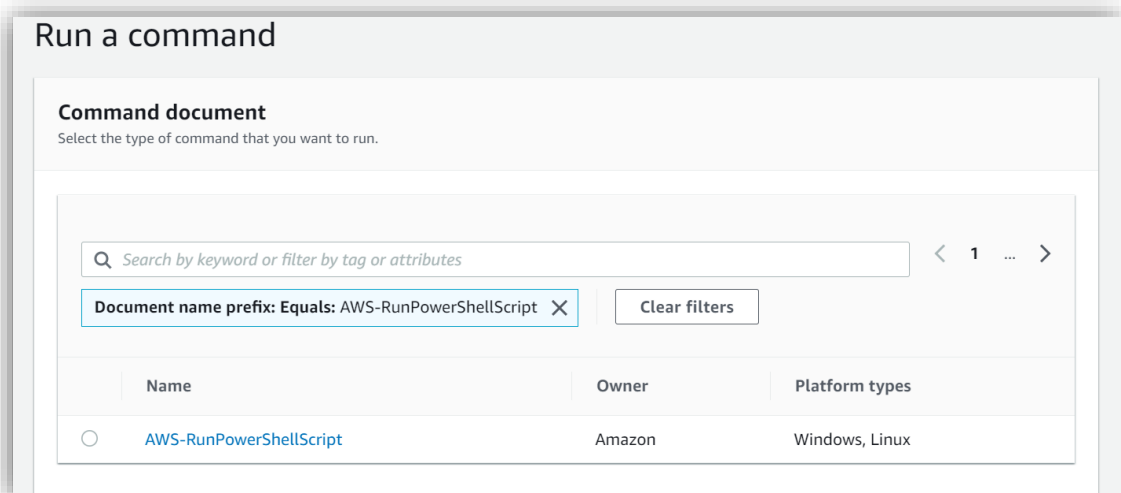
Registry or Configuration changes

Sometimes a registry or a configuration change needs to be made to a system after patching to close out a vulnerability or to make the patch effective. Systems Manager provides a simple way to accomplish that at-scale.

- Go to System Manager > Run Command

Using Run Command, a capability of AWS Systems Manager, you can remotely and securely manage the configuration of your managed instances. Run Command allows you to automate common administrative tasks and perform one-time configuration changes at scale. You can use Run Command from the AWS Management Console, the AWS Command Line Interface (AWS CLI), AWS Tools for Windows PowerShell, or the AWS SDKs. Administrators typically use Run Command to perform tasks on their managed instances: install or bootstrap applications, build a deployment pipeline, capture log files when an instance is removed from an Auto Scaling group, and join instances to a Windows domain.

- Click on “Run Command”
- Review available commands
- Click on the search bar and select “Document name prefix”, “Equals” and type “AWS-RunPowerShellScript” as the filter

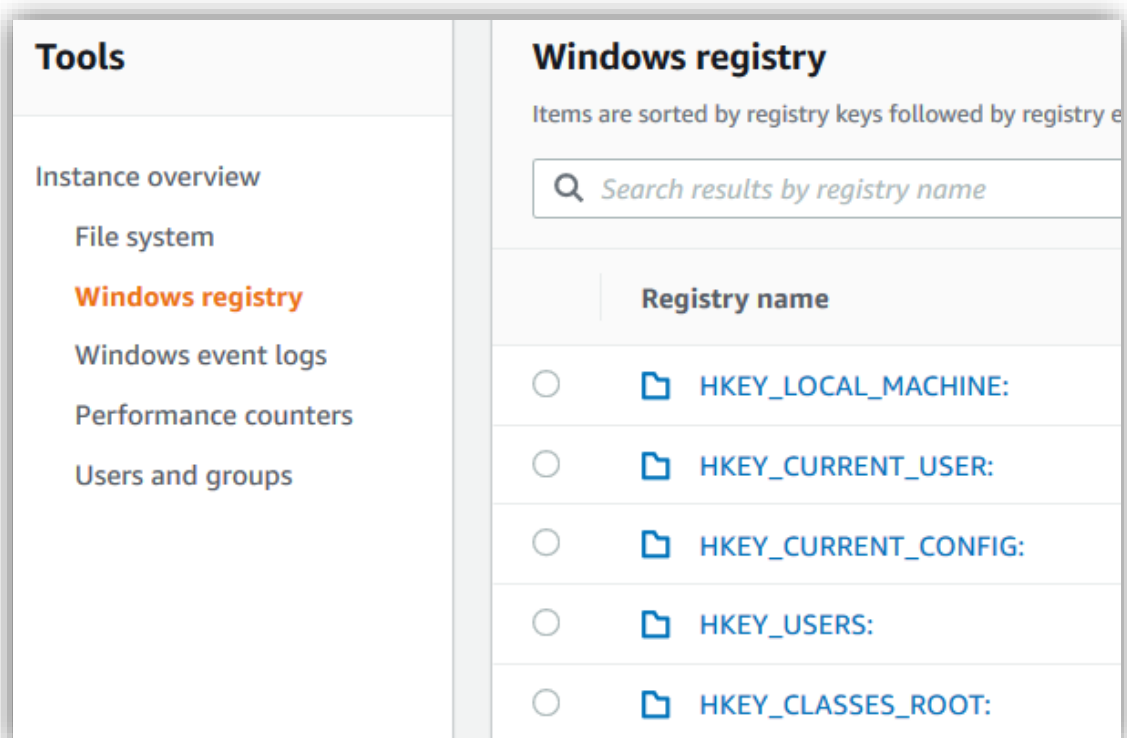


- Select AWS-RunPowerShellScript
- Under command parameters section copy/paste below Powershell script to create a test registry entry:

`new-item -path hklm:\\software -name Test`

```
new-itemproperty -path hklm:\\software\\test -Name Test1 -Value "Builder"
```

- Under Targets, select “Choose instances manually” – and select the two web servers
- Uncheck “Enable an S3 bucket” in the Output options section
- You could log results of script execution there or in CloudWatch if needed
- Click “Run”
- Hit “Refresh” on the page to see the script execute successfully
- To view the registry entry, you just added – go to Fleet Manager and click on either of the instances
- Under the instance – click on Windows Registry



- Drill-down to HKEY_LOCAL_MACHINE: > SOFTWARE > Test (You may have to go to 2nd page of the results)
- Key Test and property Test1 with value of “Builder” is what we created by executing the PowerShell script

Registry name	Registry type	Registry value
Test1	String	Builder

This ability to execute commands on your server fleet at scale gives you the ability to accomplish other tasks – such as installing EDR (Endpoint Detection & Response) tools or updating signature files.

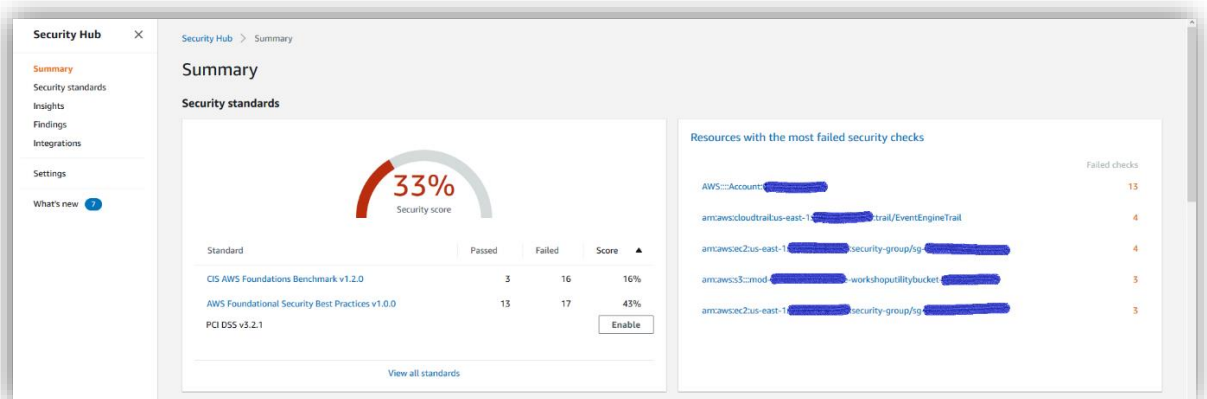
2.Detect using AWS Security Hub, AWS GuardDuty, CloudWatch, Systems Manager:

At a high level, there are two types of detective capabilities useful in reducing the risk of a major ransomware incident. First – detection of vulnerabilities/misconfigurations that may lead to a compromise and second – detection of potentially malicious activities/compromise. AWS provides tools that enable both types of detections. Inspector, for example, can help with vulnerability scanning. Security Hub integration with Endpoint Security systems can help detect malicious activity occurring on that endpoint. In this section we will use Security Hub to detect a misconfigured security group and integration between Security Hub and GuardDuty to detect an EC2 instance communicating with a known malicious IP address.

Security Hub

- In AWS Management Console – go to Security Hub

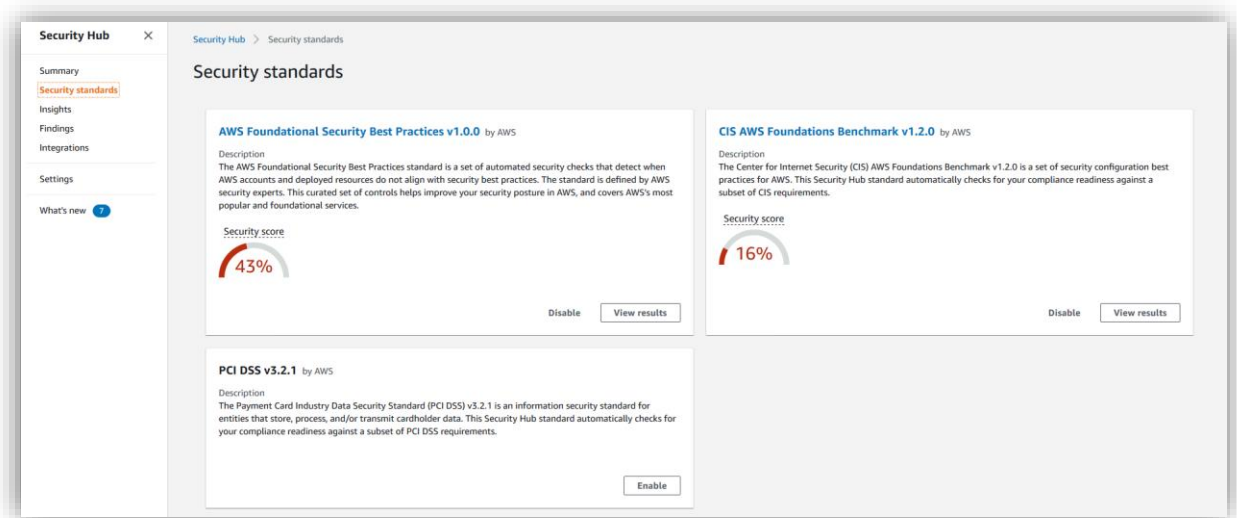
You may get a warning message *AWS Config is not appropriately enabled on some accounts. AWS Config is required for Security Hub's security checks. Review remediation steps for the related findings for CIS 2.5. If you recently enabled AWS Config, note that it can take up to 12 hours for Security Hub to detect the change.* Please ignore this message as there is no impact the labs



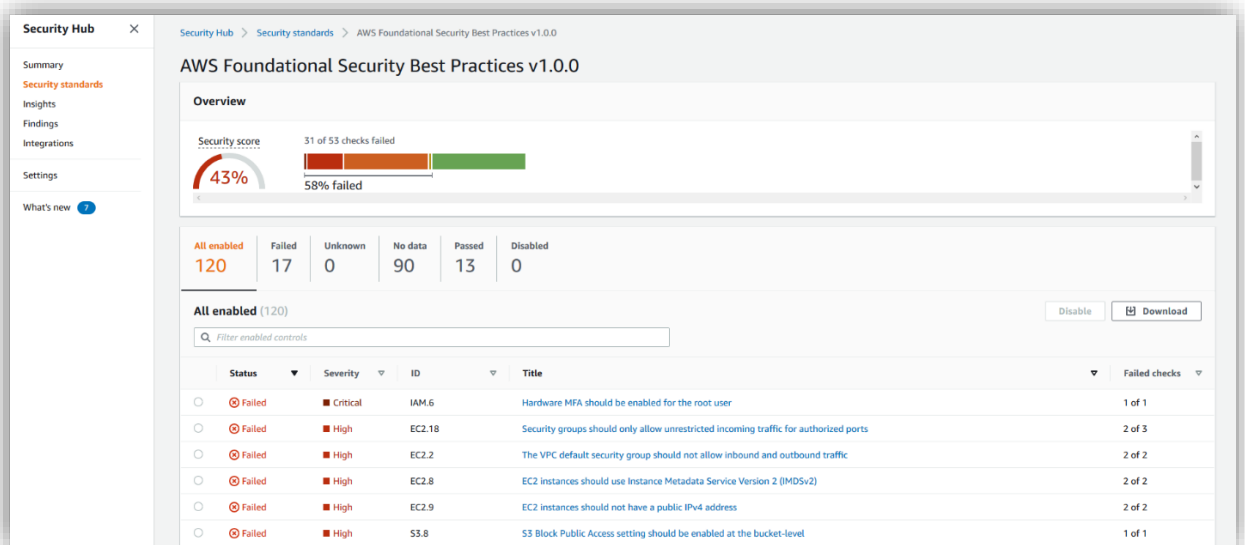
AWS Security Hub gives you a comprehensive view of your security alerts and security posture across your AWS accounts. AWS provides a range of powerful security tools at your disposal, from firewalls and endpoint protection to vulnerability and compliance scanners. Oftentimes this leaves your team switching back-and-forth between these tools to deal with hundreds, and sometimes thousands, of security alerts every day. 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, Amazon Macie, AWS Identity and Access Management (IAM) Access Analyzer, AWS Systems Manager, and AWS Firewall Manager, as well as from

AWS Partner Network (APN) solutions. AWS Security Hub continuously monitors your environment using automated security checks based on the AWS best practices and industry standards that your organization follows. You can also act on these security findings by investigating them in Amazon Detective or by using Amazon CloudWatch Event rules to send the findings to ticketing, chat, Security Information and Event Management (SIEM), Security Orchestration Automation and Response (SOAR), and incident management tools or to custom remediation playbooks.

- Click on “Security standards” on the left side and look at the Security Score for AWS Foundational Security Best Practices and CIS AWS Foundations Benchmark.



- Click on “View results” under AWS Foundational Security Best Practices
- You will see the list of controls with their status and severity. In your actual AWS environments, further exploration can be done on the failed controls along with remediation actions.



- Click in “Integrations” to look at and accept findings from other AWS services or from third-party integrations.

Security Hub Security Hub > Integrations

Integrations

Accept findings from other AWS services or from third-party integrations. You can also send findings from Security Hub to some integrations.

Filter integrations

AWS: Audit Manager

Description
AWS Audit Manager continuously audits your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

Type of integration
Receives findings from Security Hub

Categories
Governance, Risk, Compliance (GRC)

How to send findings to this integration
The integration is automatically enabled when you enable the service. No other configuration besides turning on the service is required. [Go to service homepage](#)

Status
After you follow the configuration instructions, Security Hub automatically sends findings to this service.

AWS: Chatbot

Description
AWS Chatbot is an interactive agent that makes it easy to monitor and interact with your AWS resources in your Slack channels and Amazon Chime chat rooms.

Type of integration
Receives findings from Security Hub

Categories
Instant messaging

How to send findings to this integration
The integration is automatically enabled when you enable the service. No other configuration besides turning on the service is required. [Go to service homepage](#)

Status
After you follow the configuration instructions, Security Hub automatically sends findings to this service.

AWS: Firewall Manager

Description
AWS Firewall Manager is a security management service that makes it easier to centrally configure and manage AWS WAF rules across your accounts and applications.

Type of integration
Sends findings to Security Hub

Categories
Enterprise Firewalls and Intrusion Prevention Systems (IPS), Web Application Firewall (WAF), DDoS Protection

How to receive findings from this integration
The integration is automatically enabled when you enable the service. No other configuration besides turning on the service is required. [Go to service homepage](#)

Status
Accepting findings. [See findings](#) Stop accepting findings

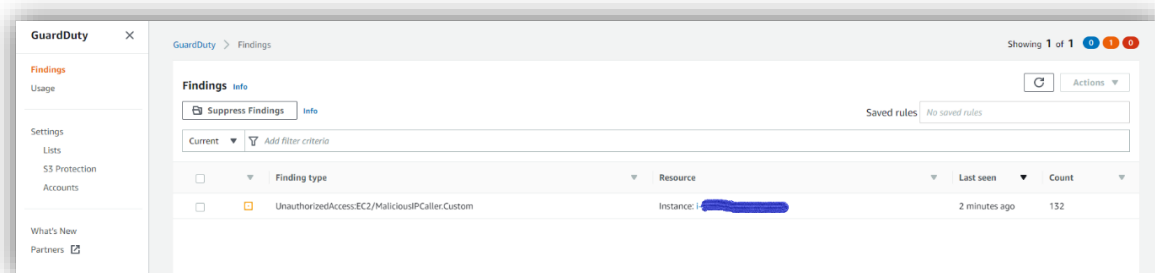
AWS Security Hub can aggregate security finding data from several AWS services and from supported AWS Partner Network (APN) security solutions. This aggregation provides a comprehensive view of security and compliance across your AWS environment. You can also send findings that are generated from your own custom security products.

- Under Integrations click on “See findings” icon for GuardDuty to proceed directly to GuardDuty console.
- Note that it may take up to 2 hours to see findings and results.

GuardDuty

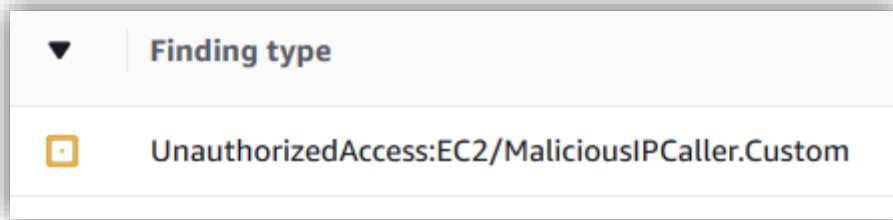
GuardDuty can detect machines that communicate with known malicious IP addresses including C2C communication. GuardDuty findings can be centrally aggregated in Security Hub. In this section we will find one of our EC2 instances talking to a known malicious IP address.

- In AWS Management Console – go to GuardDuty



Amazon GuardDuty is a threat detection service that continuously monitors for malicious activity and unauthorized behaviour to protect your AWS accounts, workloads, and data stored in Amazon S3. With the cloud, the collection and aggregation of account and network activities is simplified, but it can be time consuming for security teams to continuously analyse event log data for potential threats. With GuardDuty, you now have an intelligent and cost-effective option for continuous threat detection in AWS. The service uses machine learning, anomaly detection, and integrated threat intelligence to identify and prioritize potential threats. GuardDuty analyzes tens of billions of events across multiple AWS data sources, such as AWS CloudTrail event logs, Amazon VPC Flow Logs, and DNS logs. With a few clicks in the AWS Management Console, GuardDuty can be enabled with no software or hardware to deploy or maintain. By integrating with Amazon CloudWatch Events, GuardDuty alerts are actionable, easy to aggregate across multiple accounts, and straightforward to push into existing event management and workflow systems.

- In your GuardDuty console you will see a custom finding for unauthorized access on an EC2 from a malicious IP.



- Click on the finding name and review the finding details on the right. Note that the disallowed IP address is from a CustomThreatList – and that IP address is being accessed by one of your instances repeatedly (Count property indicates how many times the instance attempted to access this IP address)
- Scroll down to see other finding-related information
- Review the CustomThreatList by clicking on Lists > CustomThreatList

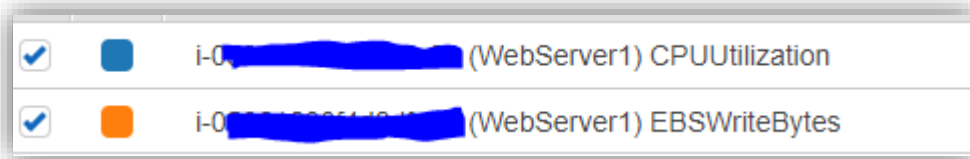
List name	List file URL	Format	Active	
CustomThreatList	https://s3.amazonaws.com/...	TXT	<input checked="" type="checkbox"/>	✎ ✕

- You can view the file – NOTE: IP addresses in the file are NOT actually malicious – they are being used to generate findings for this session only. You can use custom threat lists in your environment for testing or to enhance GuardDuty's built-in threat intelligence

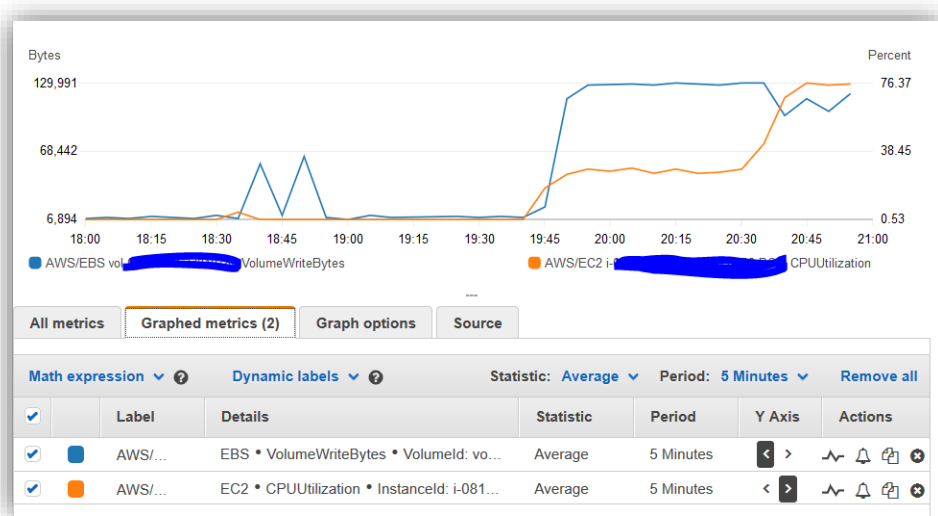
CloudWatch

A sudden increase in EC2 instance's resource utilization, specifically – CPU and Disk – may indicate malicious activity on that instance. CloudWatch metrics allow you to track instance resource utilization and allows you to set alerts when anomalous activity is detected. We are going to take a look at resource utilization by WebServer1 – since that's where we saw potentially malicious communication coming from.

- Go to CloudWatch > Under Metric > All Metrics, and select EC2 > Per-Instance Metrics, then select EBSWriteBytes and CPUUtilization metric for WebServer1



- Click on the “Graphed metrics” tab to display the two selected metrics
- Click on the “>” sign in the Y-Axis column for CPU Utilization metric to allow it to display on the same graph as EBSWriteBytes metric allowing for scale.



CloudWatch provides you with a way to enable anomaly detection alarms for your workloads. This allows you to set alarms for anomalous behavior without having to fine-tune thresholds. To try this in your environment. Click on the “heartbeat” line next to CPU Utilization metric

- Click on “Edit model” and review customization options, then click “Cancel”
- Click on the “Bell” icon to set an alarm for anomalous behavior detection
- Review alarm options

Conditions

Threshold type

☐ Static
Use a value as a threshold

☒ Anomaly detection
Use a band as a threshold

Whenever CPUUtilization is...

Define the alarm condition

☒ Outside of the band
> or < threshold

☐ Greater than the band
> threshold

☐ Lower than the band
< threshold

Anomaly detection threshold

Based on a standard deviation. Higher number means thicker band, lower number means thinner band.

2

Must be a positive number

► Additional configuration

Cancel Next

- We are not going to set an anomaly alarm during this session because it will take time for the system to learn “normal” behaviour for your environment – but it’s a useful tool to utilize for operational and security anomaly detection. To learn more about using anomaly detection and how it works in CloudWatch, please visit [Amazon CloudWatch Anomaly Detection](#) page. Click “Cancel” to close the window.

Systems Manager

Now that we have identified a suspicious EC2 instance – we want to investigate the instance further to decide if we need to invoke our IR (Incident Response) procedures.

- Go to Systems Manager > Session Manager and click on “Start Session”

Session Manager is a fully managed AWS Systems Manager capability that allows you to manage your Amazon EC2 instances, on-premises instances, and virtual machines (VMs) through an interactive one-click browser-based shell or through the AWS Command Line Interface (AWS CLI). Session Manager provides secure and auditable instance management without the need to open inbound ports, maintain bastion hosts, or manage SSH keys. Session Manager also allows

you to comply with corporate policies that require controlled access to instances, strict security practices, and fully auditable logs with instance access details

- Select the affected instance (It will be WebServer1) and click “Start Session”
- Once the session starts – type in “whoami” at the prompt to see the user context the session launched as:

```
PS C:\Windows\system32> whoami
ec2amaz-... \ssm-user
```

- Note that ssm-user is a local user created by System Manager to initiate Session Manager sessions. On Windows systems the account password is changed every time a new session is started.
- Session Manager provides you a secure way to manage instances without opening network ports or setting up bastion hosts, and without exposing privileged domain credentials to the OS on AD Domain joined servers.
- You can use the Session Manager to perform additional investigation on the machine. For example – search for files on the system that may be indicators of compromise
- Execute the following command to find files with extension .zzz in the folder c:\data (note: we’re using the folder c:\data to save time, but you can search the entire drive if needed)

```
get-childitem -path c:\data\ -include *.zzz -Recurse -ErrorAction SilentlyContinue
```

- The command should produce a list of files with extension .zzz – we will take it as a confirmation that the instance was compromised and proceed to response

3.Respond with Systems Manager:

Your response to the detected incident will depend on your organizations Incident Response (IR) procedures, which themselves are influenced by IR priorities, legal obligations, etc. AWS provides you with tools that enable automation of some of the steps in your IR plan – which greatly relieves the pressure on the response team. Note – in this builder session we do not cover backup/recovery, but in the context of ransomware it is very important to have tested offline backups that can be used to recover your data and systems in

case of a ransomware incident. AWS Backup enables you to centralize and automate data protection across AWS services – it is a cost-effective, fully managed, policy-based service that simplifies data protection at scale.

Systems Manager

A common set of steps you may want to take when you detect a compromised EC2 system is to:

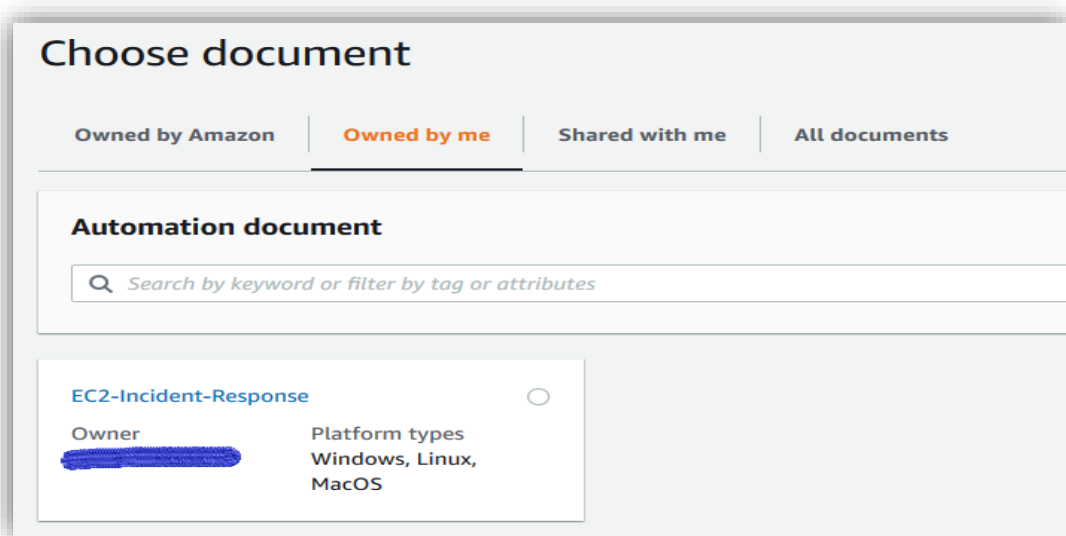
- Tag the instance to indicate that it's compromised or under investigation
- Preserve the instance by enabling termination protection and disabling "DeleteOnTermination" setting
- Isolate the instance from the network
- Preserve data by creating a snapshot of all attached EBS volumes

In this section, we will go through how to use Systems Manager and custom automation to automate these steps.

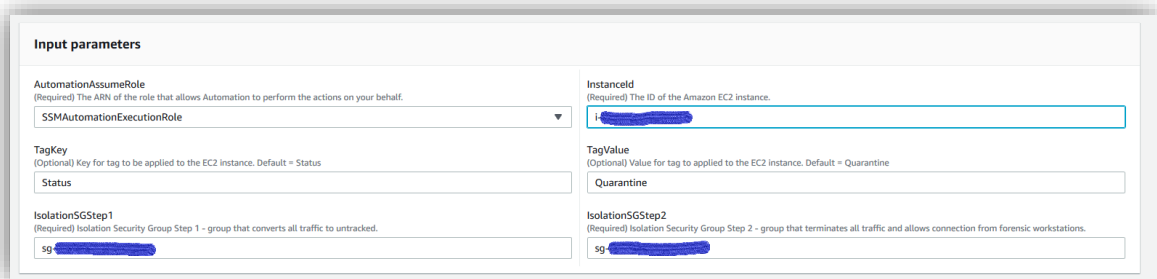
- Go to Systems Manager > Automation

Automation, a capability of AWS Systems Manager, simplifies common maintenance and deployment tasks of Amazon EC2 instances and other AWS resources.

- Click on "Execute Automation"
- Click on "Owned by me" tab at the top of the page



- Note that this is a custom document specifically created for this workshop – review the document’s content by clicking on the document name and then clicking on the Content tab
- When ready – click on “Execute Automation” button



Input parameters

AutomationAssumeRole
(Required) The ARN of the role that allows Automation to perform the actions on your behalf.
SSMAutomationExecutionRole

InstanceId
(Required) The ID of the Amazon EC2 instance.
i-[redacted]

TagKey
(Optional) Key for tag to be applied to the EC2 instance. Default = Status
Status

TagValue
(Optional) Value for tag to be applied to the EC2 instance. Default = Quarantine
Quarantine

IsolationSGStep1
(Required) Isolation Security Group Step 1 - group that converts all traffic to untracked.
sg-[redacted]

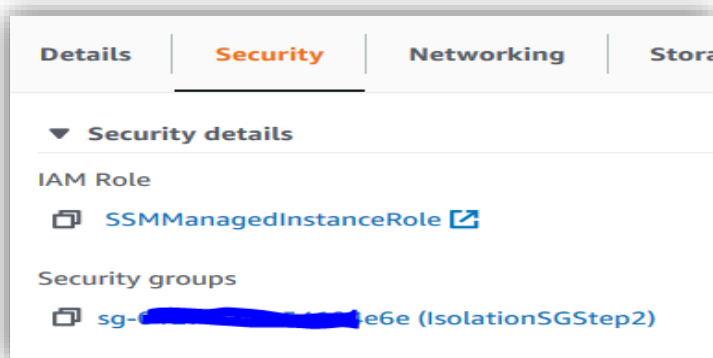
IsolationSGStep2
(Required) Isolation Security Group Step 2 - group that terminates all traffic and allows connection from forensic workstations.
sg-[redacted]

- Under Input parameters – select SSMAutomationExecutionRole as the role to perform the automation
- Open [AWS EC2 console](#) on separate browser tab to get EC2 instance ID and Security Groups IDs
- Go back to Systems Manager tab on your browser and enter instance Id of the compromised instance
- Enter Security Group ID for security group with the name “IsolationSGStep1” for Step 1, and “IsolationSGStep2” for Step 2

[redacted]' and VPC 'vpc-[redacted]2fa7'. The second is 'IsolationSGStep1' with ID 'sg-[redacted]' and VPC 'vpc-[redacted]2fa7'." data-bbox="171 557 862 602"/>

sg- [redacted]	IsolationSGStep2	vpc- [redacted] 2fa7	Isolation Step 2 - Bloc...
sg- [redacted]	IsolationSGStep1	vpc- [redacted] 2fa7	Isolation Step 1 - Conv...

- Click “Execute”
- Go to EC2 and click on WebServer1 instance – see that it now has “IsolationSGStep2” security group applied to it

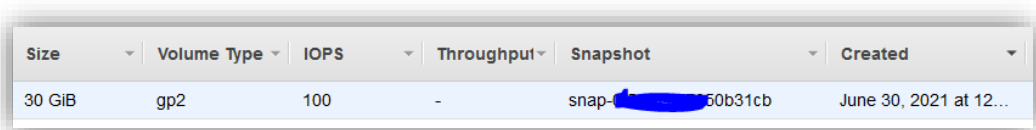


Important:

Please note that IsolationSG1 is attached to terminate any ongoing connections to the instance. If IsolationSG2 is attached directly - that has no incoming and outgoing rules, all future connections to the instance will be terminated however existing connections will not be terminated. Hence, we attach the SG1 under step1 which converts connections to untracked and on step2 we attach SG2 that blocks the connections which are untracked from continuing as well as any new connections. The end effect is to terminate not just future connections but ongoing connections as well.

For more information, please visit [Security group connection tracking](#)

- Click on Storage and then Volume ID
- See that a snapshot has been created for the volume:



Size	Volume Type	IOPS	Throughput	Snapshot	Created
30 GiB	gp2	100	-	snap-0b31cb	June 30, 2021 at 12:00

4.2 Comparative study

AWS provides a variety of resources and content to help educate and guide users on protecting against ransomware.

- **AWS Security Blog:**

The AWS Security Blog covers a wide range of security topics, including ransomware. It offers in-depth articles, case studies, best practices, and guidance on securing AWS environments, protecting against ransomware, and incident response. You can find specific articles on ransomware prevention, mitigation, and recovery by searching the AWS Security Blog archives.

- **AWS Security Documentation:**

The official AWS Security Documentation provides comprehensive information on security features, services, and best practices. It covers topics such as access management, network security, encryption, incident response, and compliance. The documentation includes specific guidance on protecting against ransomware and securing various AWS services.

- **AWS Whitepapers:**

AWS offers a range of whitepapers covering various security topics. The "AWS Security Best Practices" whitepaper is particularly relevant, as it provides guidance on securing AWS environments and mitigating common security risks, including ransomware.

- **AWS re:Invent Security Sessions:**

AWS re:Invent is an annual conference where AWS showcases its latest offerings and provides educational sessions. It features sessions focused on cloud security, including topics related to ransomware prevention, incident response, and security best practices. Recordings of past sessions are available on the AWS re:Invent website.

These resources provide valuable insights and guidance on protecting against ransomware in AWS environments. It's recommended to explore these materials, stay updated with the latest AWS announcements and security updates, and consult with AWS experts or security professionals for a comprehensive approach to ransomware prevention and mitigation in your specific AWS deployment.

4.3 Proposed Results

In this report we saw some basics about AWS Security Services and how they can be leveraged to Protect, Detect and Respond to Security events and ransomware. As with any other incident response scenarios, mitigating ransomware threats requires a strong preventive strategy.

- Patching and vulnerability management are core to preventing ransomware attacks. AWS Services like Systems manager - Patch Manager can help us with setting up constant schedules for patching operating systems.
- SecurityHub can identify misconfigured resources and aggregate security finding data from several AWS services to provide a comprehensive view of

security and compliance across your AWS environments. GuardDuty can help detect anomalous network behavior and other security threats through a GuardDuty finding indicating a potential compromised system.

- Implementing and testing an incident response playbook using automations to quickly respond to a ransomware attack is key to mitigation. Systems manager automation runbook helps define the actions to isolate and protect the evidence on a compromised system. This helps in further investigation on the compromised system.

Above are some of the capabilities of AWS Services to protect our systems and data.

5. Conclusion

In conclusion, undertaking an AWS project involves leveraging the cloud services and resources provided by Amazon Web Services to achieve specific data and system safety goals or address security issues related to ransomware attacks. The success of an AWS project depends on careful planning, effective implementation, and ongoing management. Here are some key points to consider in concluding an AWS project of this type:

- Objectives and Deliverables
- Implementation and Execution
- Performance and Optimization
- Business Impact
- Lessons Learned
- Documentation and Handover
- Training and Support
- Future Roadmap

By reviewing these aspects, we can effectively conclude an AWS project and ensure that the project's outcomes align with the initial objectives. This allows for continuous improvement, optimization, and growth within the environment.

6. References

- Securing your AWS Cloud environment from ransomware

<https://d1.awsstatic.com/whitepapers/Security/ransomware-risk-management-on-aws-using-csf.pdf>

- Ransomware recovery

<https://aws.amazon.com/cloudendure-disaster-recovery/ransomware-recovery/>

- Best Practices for Security, Identity, & Compliance

<https://aws.amazon.com/architecture/security-identity-compliance>

- AWS Security Incident Response Guide

https://d1.awsstatic.com/whitepapers/aws_security_incident_response.pdf

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

<https://docs.aws.amazon.com/whitepapers/latest/classic-intrusion-analysis-frameworks-for-aws-environments/classic-intrusion-analysis-frameworks-for-aws-environments.pdf>

- Ransomware Risk Management on AWS Using the NIST Cyber Security Framework (CSF)

<https://d1.awsstatic.com/whitepapers/Security/ransomware-risk-management-on-aws-using-csf.pdf>