Cloud Incident Response Playbook

VERSION 1.0

This document is provided for informational purposes only and provides guidelines and recommendations on how to prevent and respond to possible security incidents within your AWS cloud environment. It is intended purely as a guide and is not intended to be exhaustive. It represents AWS's current product offerings and practices as of the date of issue of this document and is subject to change. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services – always consult a trained cybersecurity professional for advice before making any business-critical decisions within your organisation. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

1 Executive Overview

Cloud Incident Response (IR) is crucial in the context of cloud computing for organizations to promptly mitigate a threat and reduce damaging impact to business services. As organizations increasingly rely on cloud services, incidents and security breaches can occur that may compromise data, disrupt services, or result in financial and reputational damage. Cloud IR is essential to ensure a swift and coordinated response, and to minimize the impact of incidents on cloud infrastructure and data.

This guide provides an overview of fundamentals in responding to security incidents within your AWS cloud environment. This guide is for those in technical roles and assumes that you are familiar with the general principles of information security, have a basic understanding of security incident response in your current on-premises environments, and have some familiarity with cloud services. For more information, please refer to AWS Security Incident Response Guide.

2 Incident Response Elements

Incident response is a coordinated and structured approach documented by the National Institute for Standards and Technology (NIST) Incident Response lifecycle: 1) preparation, 2) detection and analysis, 3) containment, eradication, and recovery and 4) post-incident activity. The goal is to effectively remove a threat from the organization’s computing environment while minimizing damage and restoring normal operations as quickly as possible.

Preparation – Identify roles and responsibilities and chain of command of the incident response team. Define what constitutes an incident, incident severity and prepare the team. Develop and prepare relevant technology components to aid the IR execution.

Detection and Analysis – Start the investigation upon detecting that an event has occurred or being alerted to one. Event validation should take place and a decision will be made to determine the severity of the event. Once an incident has been confirmed, this phase is used to scope the environment for additional indicators of compromise.

Containment – Contain the intrusion to minimize the impact scope and continued unauthorized access in the environment. Assess how attackers are maintaining persistence, laterally moving around the network, and how command and control communication with external source is established.

Eradication – Eliminate the threat, remove suspicious and unauthorized resources to return the environment to a safe state. Remediation plans are developed, and recommendations are implemented in a planned and controlled manner. Examples of such remediation actions:

- Blocking malicious IP address
- Implementing a “blackhole” for malicious domain names
- Resetting account credentials
- Reverting unauthorized configuration changes
- Patching vulnerable systems being exploited

Recovery – Restore back to normal business service operations. Besides restoring impacted systems from the last trusted backup, this phase will include improvement actions on the overall security of the network and to detect and prevent subsequent threat activities. Some recovery actions are:

- Rebuilding compromised systems
- Implementing centralised logging
- Establishing security awareness training program
- Enhancing network visibility

Post-Incident Activity – Verify that the incident has been mitigated, the threat has been removed, and that additional countermeasures have been implemented correctly. Assign designated owners to the prioritized action items identified in this retrospective review. This is a crucial step as part of continuous improvement.

3 Common Incident Response Scenarios

A.1 Threat Scenario 1 – Compromised AWS account IAM credentials

1. Step: Detection and Analysis
   - (i) Review Amazon GuardDuty for security alert findings related to compromised IAM credentials that provides information on the suspicious event.
     - a) CredentialAccess:IAMUser/AnomalousBehavior
     - b) Exfiltration:IAMUser/AnomalousBehavior
     - c) Impact:IAMUser/AnomalousBehavior
     - d) InitialAccess:IAMUser/AnomalousBehavior
     - e) Persistence:IAMUser/AnomalousBehavior
     - f) Policy:IAMUser/RootCredentialUsage
     - g) PrivilegeEscalation:IAMUser/AnomalousBehavior
     - h) Stealth:IAMUser/CloudTrailLoggingDisabled
     - i) Stealth:IAMUser/PasswordPolicyChange
     - j) UnauthorizedAccess:IAMUser/*
   - (ii) Review AWS CloudTrail logs on the API service calls made by the compromised IAM role, IAM user or IAM Access Key for suspicious events. Review CloudTrail for suspicious activity by searching for these events: `"ConsoleLogin"`, `"AssumeRole"`, `"GetFederationToken"`, `"GetSessionToken"`, `"CreateUser"`, `"CreateKeyPair"`, `"CreateAccessKey"`, `"CreateLoginProfile"`. Note that "userIdentity" should show up as "type": "Root" for the root user or "type": "IAMUser" for any local IAM users on the account.
   - (iii) Correlate the above CloudTrail findings with IAM Access Advisor data to identify services accessed by the compromised credentials and potential excessive permissions that can be removed or narrowed in scope.
   - (iv) Review IAM Access Analyzer (external access) for any unauthorized sharing of resources e.g. KMS-CMK key, S3, to unfamiliar external accounts. Focus on findings that indicate public access or access from other AWS accounts for possible signs of data exfiltration or lateral movement.
   - (v) Review Amazon RDS query logs (in CloudWatch log group) on the activities performed by the compromised RDS user using tools like Amazon Athena or third-party log analysis solutions.

2. Step: Containment
   - (i) Revoke the compromised IAM Role's active session. Go to AWS IAM and search for the compromised IAM role and select "Revoke Active Sessions". Alternatively, revoke the active sessions using AWS CLI:
     ```
     "aws sts revoke-role-session –role-name
     <role_name> --role-session-name <session_name>"
     ```

(ii) Deactivate the compromised long-term IAM Access Key (starts with AKIA). Alternatively, revoke the compromised access key using IAM console or AWS CLI:

```
“aws iam update-access-key –access-key-id
<access_key_id> --status Inactive –user-name
<user_name>”
```

(iii) Reset the password of the compromised IAM user and enforce MFA if not previously set.

(iv) Revoke any active IAM Identity Center sessions. Go to AWS IAM Identity Center Users page, select “Active > Sessions”. Select the specific listed session(s) and choose “Delete Session”

(v) Reset the affected RDS database admin (master) user password. Go to Amazon RDS console → select the RDS DB instance and choose **Modify**. → Enter the password in the **New Master Password** field. → Choose **Continue** and **Modify DB instance**.

```
aws rds modify-db-instance --db-instance-
identifier <db-identifier> --master-user-password
<new password>
```

(vi) Reset the affected RDS database user’s password. Check the specific database engine (Oracle, MSSQL, MySQL, PostgreSQL) for instructions.

3. Step: Eradication

(i) Delete unrecognized or unauthorized resources e.g. EC2 instances, security groups, IAM roles that were created or modified during the security event window.

(ii) Rotate impacted long-term IAM Access Keys by deactivating the existing ones and create a new long-term IAM Access Key.

(iii) Store credentials in AWS Secrets Manager for secure storage in an encrypted vault.

4. Step: Recovery

(i) Assess the impact of the compromise, including data loss, financial impact, and any compliance or regulatory implications.

(ii) Restore data or configurations from backups taken before the compromise.

(iii) Enable MFA for all IAM users and implement least privilege access policies.

A.2 Threat Scenario 2 – Web threats impacting publicly accessible service (Website Portal, API Gateway)

1. Step: Detection and Analysis

   (i) Review web server logs and/or API Gateway access logs to look for common attack signatures like SQL injection attempts, cross-site scripting (XSS) or suspicious requests from specific IP addresses or user-agents.

   (ii) Review AWS Web Application Firewall (WAF) metrics from CloudWatch log groups and query the WAF logs. Go to AWS WAF and Shield console → choose the WAF WebACL to review the dashboard on rules being triggered the most and investigate the corresponding requests to understand the nature of the attacks.

   Sample WAF CloudWatch Log Group Queries

   List top 20 WAF logs

   ```
   fields @timestamp, httpRequest.uri,
   httpRequest.country, httpRequest.clientIp,
   httpRequest.headers.1.value,
   terminatingRuleMatchData, action
   | sort @timestamp desc
   | limit 20
   ```

   Entries counted by a specified rule in a rule group

   ```
   fields @timestamp
   | filter (@message like
   'excludedRules":[{"exclusionType":"EXCLUDED_AS_COU
   NT","ruleId":"NoUserAgent_HEADER"}]}' and @message
   like 'terminatingRuleId":"Default_Action"')
   | parse @message '"ruleId":*}]}' as
   ruleMatchDetails
   | display @timestamp, httpRequest.clientIp,
   httpRequest.country, ruleMatchDetails,
   httpRequest.requestId
   |limit 10
   ```

   (iii) Scan the origin application service using Amazon Inspector to identify potential vulnerabilities that may be exploited.

   (iv) Scan the origin application EC2 instance using Amazon GuardDuty Malware Protection or existing host-based anti-malware security agent to identify malicious artefacts.

2. Step: Containment

   (i) If there is no WAF rule, attach a protective WAF WebACL to the Elastic (Application) Load Balancer (ALB) of the website portal or API Gateway. Implement WAF rules that correspond to the technology stack of the publicly accessible service. Essential WAF rules include AWS Managed Core Rule Set, IP Reputation List, Rate Limit Rule, SQL Database rules and WAF Botnet Controls.

(ii) Create conditions in AWS WAF that match the unusual behaviour. Add those conditions to the WAF WebACL firstly in “COUNT” action mode for validation before progressing to “BLOCK” action mode.

(iii) Implement Rate Limiting rule (as rate-based rule) on ALB or API Gateway to prevent excessive requests and protect against potential denial-of-service (DoS) attacks. Set the appropriate threshold and window period of evaluation (1 or 5 minutes) firstly in “COUNT” action mode for validation before progressing to “BLOCK” action mode.

(iv) Consider Content Delivery Network (CDN) Caching services such as Amazon CloudFront to serve cached content to users, reducing the load on origin servers to mitigate any potential impact.

3. Step: Eradication

(i) Consider temporarily isolating compromised EC2 instances from the network or place them on a security group with restricted access to prevent the attacker from interacting with affected resources. For compromised containers, isolate or suspend them to prevent further execution of malicious code. To isolated the compromised EC2 or container, detach existing security group and attach a new security group without any rules for both inbound and outbound.

(ii) Reduce attack surface by adjusting the security group of the origin service to only receive connections from CloudFront. Set the source address in the ALB to AWS CloudFront prefix list.

(iii) Patch the origin application service to resolve known vulnerabilities.

(iv) For injection of malicious data into database, identify and remove affected records.

(v) Remove any malicious files that were download to the web server (EC2 instance) or other systems.

(vi) Revert unauthorized configuration changes.

4. Step: Recovery

(i) Re-run vulnerability scans or penetration tests to confirm that previously identified vulnerabilities are no longer present.

(ii) Restore the database from the last clean RDS backup snapshot if necessary

(iii) Rebuild or restore clean versions of impacted application files or libraries.

A.3 Threat Scenario 3 – Malware infection of EC2 instance

1. Step: Detection and Analysis

(i) Capture the metadata information of the compromised EC2 instance
- Instance Type and ID
- IP address(es)
- Security Group(s)
- VPC / Subnet ID
- Region
- AMI ID
- IAM Instance Role
- Launch Time
- Amazon GuardDuty finding

(ii) Get the System Log for the EC2 instance
- Go to Amazon EC2 → choose EC2 Instance → Actions → Monitor and Troubleshoot → Get system log

(iii) Review Amazon GuardDuty finding about the compromised EC2 instance to understand the source of alert, finding type and behaviour.

(iv) Run Amazon GuardDuty Malware Protection for EC2 on-demand scans of the compromised EC2 instance to identify potential threats in the EBS file system. Copy the instance-id of the compromised EC2 instance.
- Go to Amazon GuardDuty → Malware Protection for EC2 → Enter the arn of the compromised Amazon EC2 instance (format `arn:aws:ec2:ap-southeast-1:555555555555:instance/i-b188560f`) to start the on-demand scan. The scan results (Clean, Infected) shall provide information on the analysis outcome.

(v) Deregister EC2 instance if it is part of EC2 Auto-Scaling Group or a registered target instance of Elastic Load Balancer.
- Go to Amazon EC2 Auto Scaling → Instance management → select the compromised EC2 instance → choose Actions and Detach.
- Go to Amazon EC2 → choose Load Balancers → Edit Instances → deregister the compromised EC2 instance.

(vi) Create an Amazon EBS Snapshot of the compromised EC2 instance.
- Go to Amazon EC2 → Snapshots.
- Choose "Create Snapshot" → Select resource type (Volume) → Select volume of compromised EC2 instance.
- Enter description of snapshot and add tags to the EBS snapshot.
- Choose "Create Snapshot".

(vii) Review Amazon CloudTrail logs on the API service calls made by the EC2 IAM instance role for suspicious events such as `iam:CreateUser`, `iam:CreateAccessKey`, `ec2:AuthorizeSecurityGroupIngress`, `ec2:CreateSecurityGroup`, `s3:DeletePublicAccessBlock`.

2. Step: Containment

(i) Isolate the compromised EC2 instance.

- Go to Amazon EC2 → Select the compromised EC2 instance to isolate.
- Select "Security" → choose the security group → detach existing security group and attach security group "`isolate-ec2-sg`".

(ii) Remove the IAM instance role attached to the compromised EC2 instance to prevent access to AWS services.

- Go to Amazon EC2 → Select the compromised EC2 instance and choose "Actions" → Modify IAM role.
- Remove the attached IAM Role.

(iii) Isolate compromised resources from the public internet by modifying the security groups or network ACLs.

3. Step: Eradication/Remediation

(i) Patch the compromised EC2 or fix the specific application vulnerability.

(ii) Go to Systems Manager → Patch Manager → Choose the EC2 instance(s) to run the patch based on the OS Patch Baseline definition.

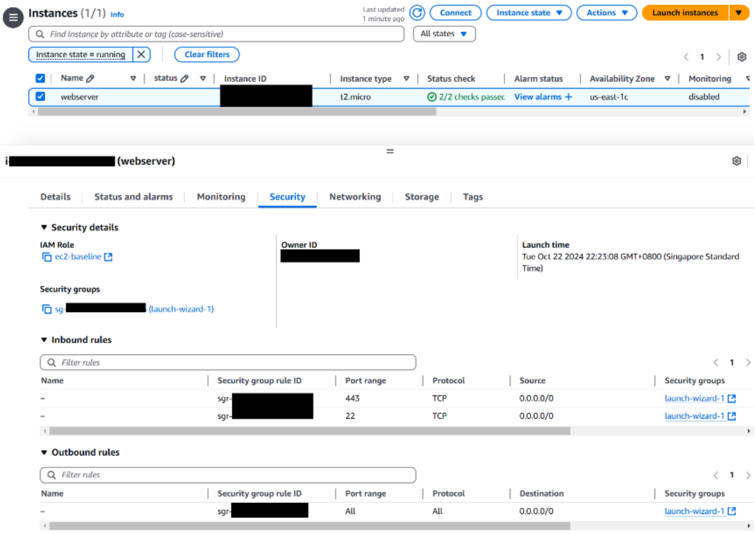
4. Step: Recovery

(i) Create a new EC2 instance from a trusted and recently patched AMI.

(ii) Restore from the last known trusted EBS snapshot.

Here's a general outline of the steps:

(a) Go to the Amazon EC2 dashboard and select the compromised EC2 instance to isolate.



☑	Name	status	Instance ID	Instance type	Status check	Alarm status	Availability Zone	Monitoring
☑	webserver		[redacted]	t2.micro	2/2 checks passec	View alarms +	us-east-1c	disabled

i[redacted] (webserver)

▼ Security details

IAM Role
ec2-baseline

Owner ID
[redacted]

Launch time
Tue Oct 22 2024 22:23:08 GMT+0800 (Singapore Standard Time)

Security groups
sg-[redacted] (launch-wizard-1)

▼ Inbound rules

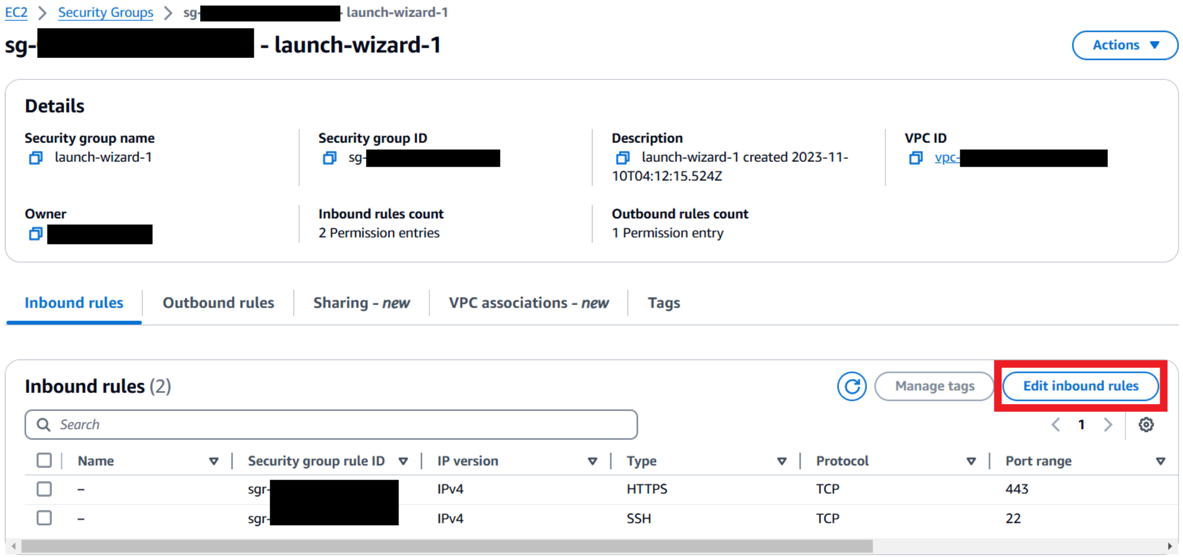
Name	Security group rule ID	Port range	Protocol	Source	Security groups
–	sgr-[redacted]	443	TCP	0.0.0.0/0	launch-wizard-1
–	sgr-[redacted]	22	TCP	0.0.0.0/0	launch-wizard-1

▼ Outbound rules

Name	Security group rule ID	Port range	Protocol	Destination	Security groups
–	sgr-[redacted]	All	All	0.0.0.0/0	launch-wizard-1

Figure 1 - EC2 Dashboard

(b) Modify its security group rules to restrict inbound and outbound traffic:



EC2 > Security Groups > sg-[redacted] - launch-wizard-1

sg-[redacted] - launch-wizard-1

Details

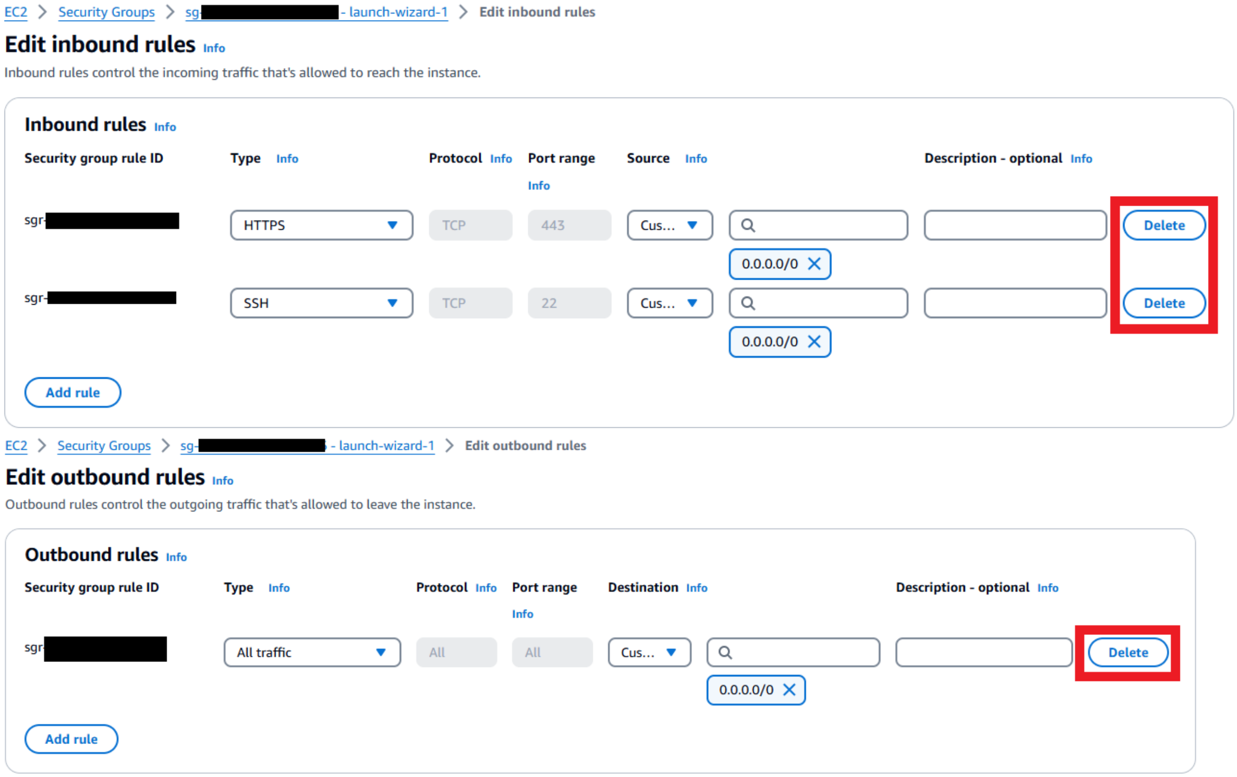
Security group name launch-wizard-1	Security group ID sg-[redacted]	Description launch-wizard-1 created 2023-11-10T04:12:15.524Z	VPC ID vpc-[redacted]
Owner [redacted]	Inbound rules count 2 Permission entries	Outbound rules count 1 Permission entry	

Inbound rules (2)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range
□	–	sgr-[redacted]	IPv4	HTTPS	TCP	443
□	–	sgr-[redacted]	IPv4	SSH	TCP	22

Figure 2 - EC2 Security Group

(c) After which, remove any inbound/outbound rules that were previously created:



EC2 > Security Groups > sg-[redacted] - launch-wizard-1 > Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type	Protocol	Port range	Source		Description - optional	
sgr-[redacted]	HTTPS	TCP	443	Cus...	0.0.0.0/0		Delete
sgr-[redacted]	SSH	TCP	22	Cus...	0.0.0.0/0		Delete

EC2 > Security Groups > sg-[redacted] - launch-wizard-1 > Edit outbound rules

Edit outbound rules Info

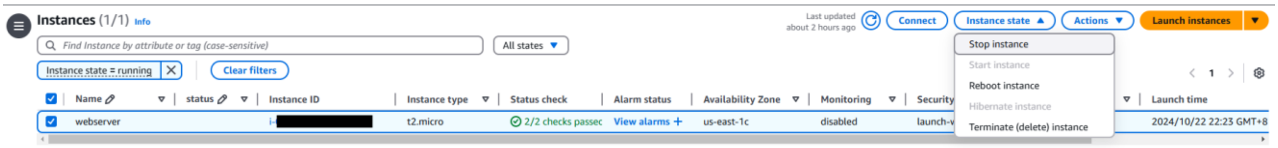
Outbound rules control the outgoing traffic that's allowed to leave the instance.

Outbound rules Info

Security group rule ID	Type	Protocol	Port range	Destination		Description - optional	
sgr[redacted]	All traffic	All	All	Cus...	0.0.0.0/0		Delete

Figure 3 - EC2 security group inbound and outbound rules

(d) Otherwise, you may stop the instance temporarily:

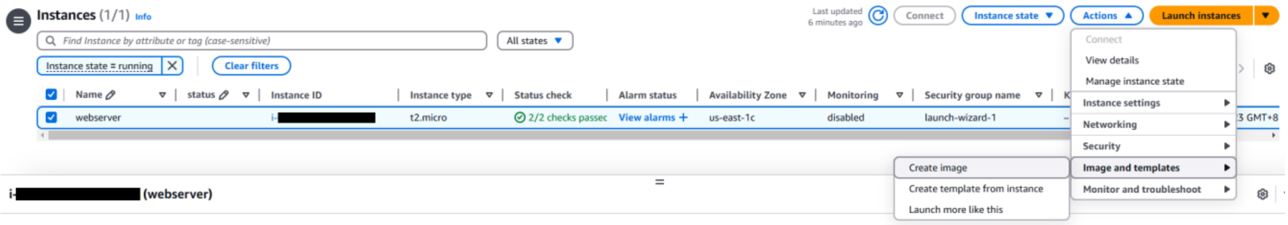


☑	Name	status	Instance ID	Instance type	Status check	Alarm status	Availability Zone	Monitoring	Security	Launch time
☑	webserver		i-[redacted]	t2.micro	2/2 checks passec	View alarms +	us-east-1c	disabled	launch-v	2024/10/22 22:23 GMT+8

Figure 4 - Stop EC2 instance

(e) Other optional steps

a. Create an AMI Image. Select the EC2 instance you want to image, click on the "Actions" dropdown and select "Create Image":

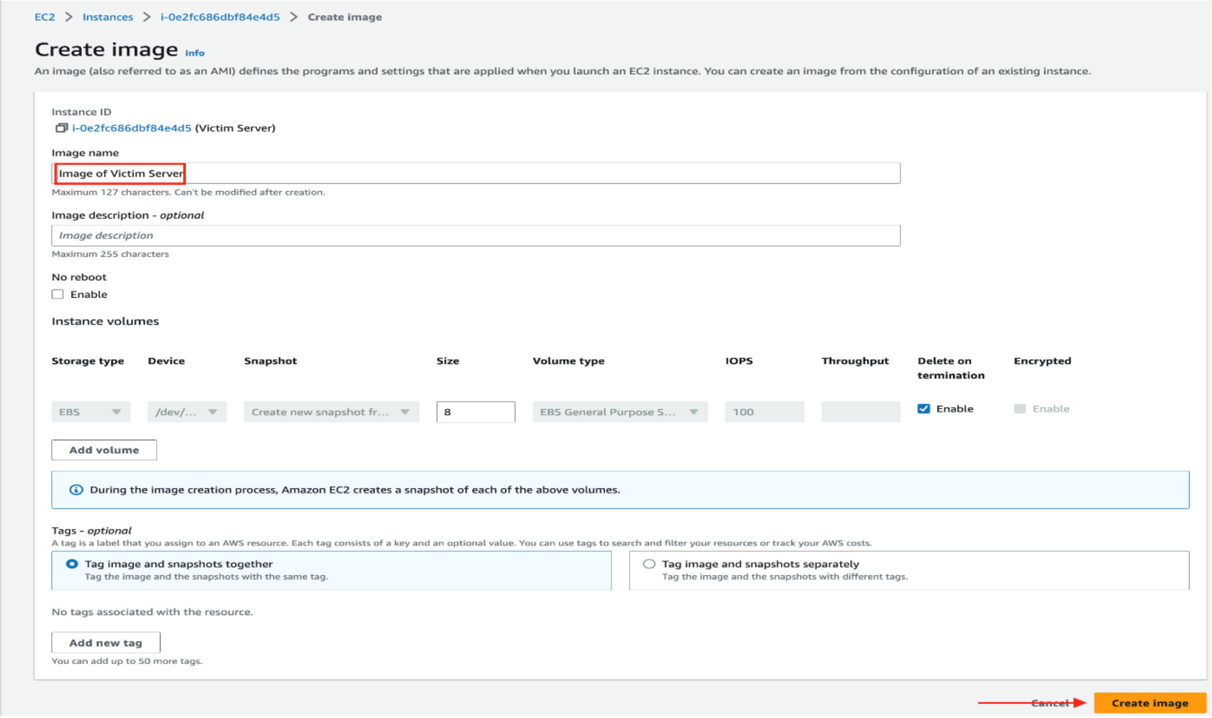


☑	Name	status	Instance ID	Instance type	Status check	Alarm status	Availability Zone	Monitoring	Security group name
☑	webserver		i-[redacted]	t2.micro	2/2 checks passec	View alarms +	us-east-1c	disabled	launch-wizard-1

i-[redacted] (webserver)

Figure 5 - Create image from compromised EC2

b. Provide a unique name and leave the default configuration as it is for the image. Select “No reboot” if you don't want your instance to be shut down:



EC2 > Instances > i-0e2fc686dbf84e4d5 > Create image

Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID
i-0e2fc686dbf84e4d5 (Victim Server)

Image name
Image of Victim Server
Maximum 127 characters. Can't be modified after creation.

Image description - *optional*
Image description
Maximum 255 characters

No reboot
☐ Enable

Instance volumes

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted
EBS	/dev/...	Create new snapshot fr...	8	EBS General Purpose S...	100		☑ Enable	☐ Enable

During the image creation process, Amazon EC2 creates a snapshot of each of the above volumes.

Tags - *optional*
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

● Tag image and snapshots together
Tag the image and the snapshots with the same tag.

○ Tag image and snapshots separately
Tag the image and the snapshots with different tags.

No tags associated with the resource.

You can add up to 50 more tags.

A.4 Additional incident response playbook resources

1. Library of AWS customer incident response playbooks: https://github.com/aws-samples/aws-customer-playbook-framework
    - (i) Responding to Ransomware attacks in the AWS cloud environment: https://github.com/aws-samples/aws-customer-playbook-framework/blob/main/docs/Responding_to_Ransom_in_AWS.md
    - (ii) Analyzing VPC Flow Logs: https://github.com/aws-samples/aws-customer-playbook-framework/blob/main/docs/Analyzing_VPC_Flow_Logs.md
    - (iii) Responding to Amazon Bedrock Security Events: https://github.com/aws-samples/aws-customer-playbook-framework/blob/main/docs/Bedrock_Response.md
    - (iv) Denial of Service / Distributed Denial of Service: https://github.com/aws-samples/aws-customer-playbook-framework/blob/main/docs/Denial_of_Service.md
    - (v) Unauthorized Network Changes: https://github.com/aws-samples/aws-customer-playbook-framework/blob/main/docs/Unauthorized_Network_Changes.md

4 Conclusion

It is important for you to consider the fundamental security incident response concepts for your AWS environment. By utilizing the existing controls, cloud features, and options for remediation, you can enhance the security of your cloud environment. You may start with small incremental steps with gradual incorporation of enhancements to your response procedures, so as to ensure that you are better prepared for any security incident.

For further assistance, all AWS customers can engage the AWS Customer Incident Response Team (CIRT) through an AWS support case. You can seek also assistance from SingCERT by reporting the incident via https://go.gov.sg/singcert-incident-reporting-form .

Note: Several steps in incident response may be highly technical. If necessary, organisations should consider engaging a cybersecurity services vendor to assist with the investigation and/or remediation. Please refer to this list of Cybersecurity Advisory and Consultancy service providers (if required): https://sgtech-prod-api.sgtech.org.sg/api/Common/GetPDF?type=artical&&fileName=f509e67b-1734-4f68-b03e-743fdd659e95.pdf.

Appendix

A.1 Log sources for commonly used AWS services

Service name	Evidence
AWS Cloud Trail	• CloudTrail Management Events • CloudTrail Data Events (S3, DynamoDB, Lambda) Note that by default, CloudTrail data events are not enabled.
Amazon Elastic Compute Cloud (EC2)	• CloudTrail • Elastic Block Storage (EBS) volume image • System-level logs (Application, Operation System)
AWS Lambda	• CloudTrail • CloudTrail – Data Events for Lambda (invocations) • Function Logs (CloudWatch)
Amazon Relational Database Service (RDS)	• CloudTrail • RDS Audit / Error Logs (CloudWatch) • RDS SQL query logs (CloudWatch) • Reference: Monitoring RDS log files
Amazon Redshift	• CloudTrail • Connection log • User log • User activity log • Reference: Database audit logging
Amazon DynamoDB	• CloudTrail • CloudTrail – Data Events for DynamoDB
Amazon Simple Storage Service (S3)	• CloudTrail • CloudTrail – Data Events for S3 • S3 Server access logs
Amazon Elastic File System	• CloudTrail
Amazon Virtual Private Cloud (VPC)	• CloudTrail • VPC Flow Logs (Network)

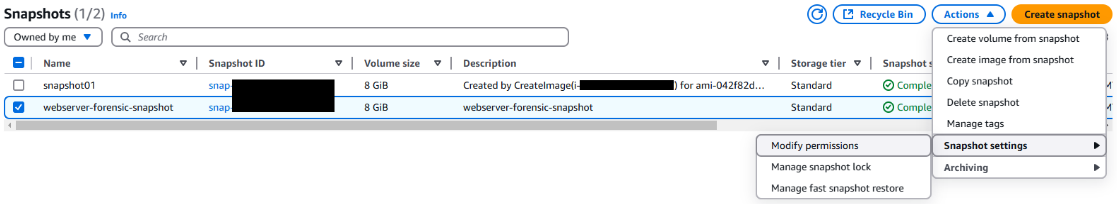
Service name	Evidence
Amazon Route 53 Resolver Resolver DNS Firewall	• CloudTrail • Route53 DNS query logs (CloudWatch) • Domain Registrations
Identity Access Management (IAM)	• CloudTrail • IAM Access Analyzer
AWS Web Application Firewall (WAF)	• WAF request logs (CloudWatch)

A.2 Explanation of different services (EC2 instance, RDS, Containers) actions available to the IR analyst

1. Amazon EC2 Virtual Machines

(a) This step is to perform digital forensics analysis of the EBS volume of affected EC2 instance using a separate AWS account. Otherwise, you may skip to part (b) to create an EC2 forensic instance.

(i) In the original account which hosts the compromised EC2 instance, navigate to the "Snapshots" section and locate the snapshot image that was created in the stage recovery of Threat Scenario 3. After which, click on "Actions" and choose "Modify Permissions" to share the snapshot with the separate AWS account that will perform the forensic analysis:

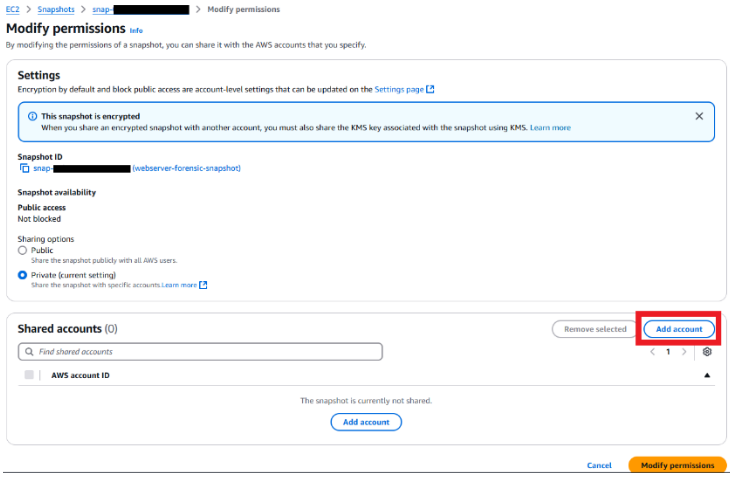

Snapshots (1/2) Info

Owned by me ▼ | Search

☐	Name	Snapshot ID	Volume size	Description	Storage tier	Snapshot s
☐	snapshot01	snap-[redacted]	8 GiB	Created by CreateImage(i-[redacted]) for ami-042f82d...	Standard	✓ Comple
☑	webserver-forensic-snapshot	snap-[redacted]	8 GiB	webserver-forensic-snapshot	Standard	✓ Comple

Figure 6 - Modify permissions of EBS snapshot

(ii) Select "Add account" and enter the AWS account ID of the recipient account:


EC2 > Snapshots > snap-[redacted] > Modify permissions

Modify permissions Info

By modifying the permissions of a snapshot, you can share it with the AWS accounts that you specify.

Settings

Encryption by default and block public access are account-level settings that can be updated on the Settings page

This snapshot is encrypted
When you share an encrypted snapshot with another account, you must also share the KMS key associated with the snapshot using KMS. Learn more

Snapshot ID
snap-[redacted] (webserver-forensic-snapshot)

Snapshot availability

Public access
Not blocked

Sharing options
○ Public
Share the snapshot publicly with all AWS users.
● Private (current setting)
Share the snapshot with specific accounts.Learn more

Shared accounts (0)

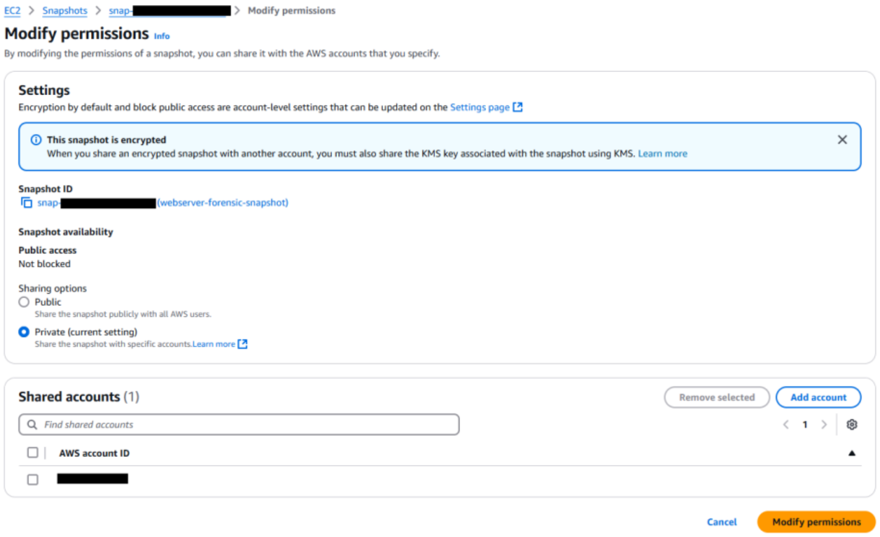
Find shared accounts

☐ AWS account ID

The snapshot is currently not shared.

Figure 7 - Add account to share EBS forensic snapshot

(iii) Once done, press on "Save changes" to share the snapshot with the separate AWS account that will perform the forensic analysis:



EC2 > Snapshots > snap-[redacted] > Modify permissions

Modify permissions Info

By modifying the permissions of a snapshot, you can share it with the AWS accounts that you specify.

Settings

Encryption by default and block public access are account-level settings that can be updated on the Settings page

This snapshot is encrypted
When you share an encrypted snapshot with another account, you must also share the KMS key associated with the snapshot using KMS. Learn more

Snapshot ID
snap-[redacted](webserver-forensic-snapshot)

Snapshot availability

Public access
Not blocked

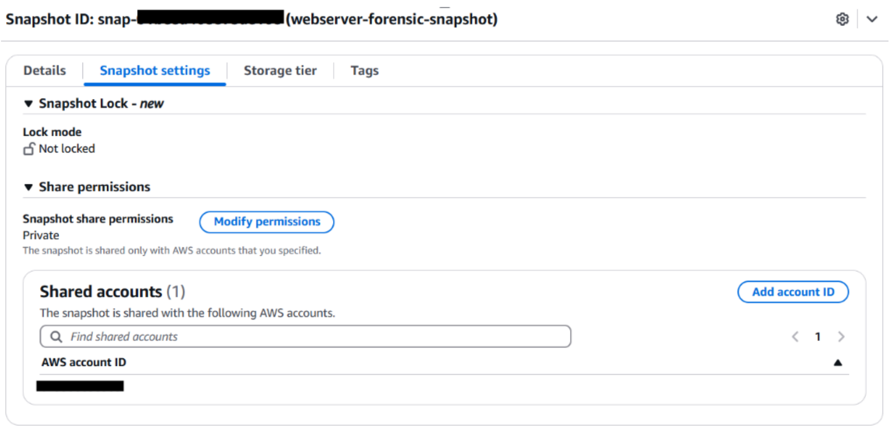
Sharing options
○ Public
Share the snapshot publicly with all AWS users.
● Private (current setting)
Share the snapshot with specific accounts.Learn more

Shared accounts (1)

□	AWS account ID
□	[redacted]

Figure 8 - Modify permissions of EBS forensic snapshot

(iv) To verify that the snapshot has been successfully shared, check if the receiving AWS account ID appears in the list of permissions for the snapshot:



Snapshot ID: snap-[redacted](webserver-forensic-snapshot)

▼ Snapshot Lock - *new*

Lock mode
Not locked

▼ Share permissions

Snapshot share permissions
Private
The snapshot is shared only with AWS accounts that you specified.

Shared accounts (1)
The snapshot is shared with the following AWS accounts.

AWS account ID
[redacted]

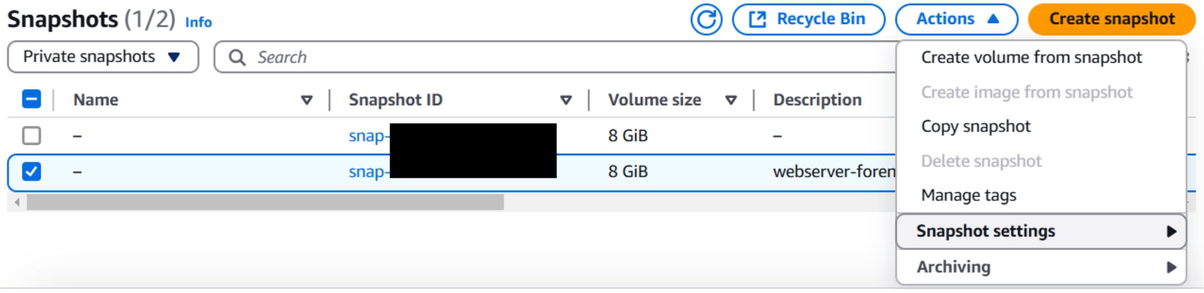
Figure 9 - Verify EBS forensic snapshot shared accounts setting

(v) Update the KMS-Customer Managed Key policy used for the EBS forensic snapshot encryption to allow a specific IAM principal role from the receiving AWS account to have KMS cryptographic operations.

```
{
    "Sid": "Allow receiving role use of the customer managed key",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::account-id:role/Admnistrator"
        ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}

{
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::account-id:role/Administrator"
        ]
    },
    "Action": [
        "kms:CreateGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": true
        }
     }
}
```

(b) In the receiving AWS account for forensic analysis, go to the Amazon EC2 Snapshots dashboard to copy snapshot of the shared EBS forensic snapshot. Encrypt the new copied snapshot using the KMS key within the receiving AWS account.

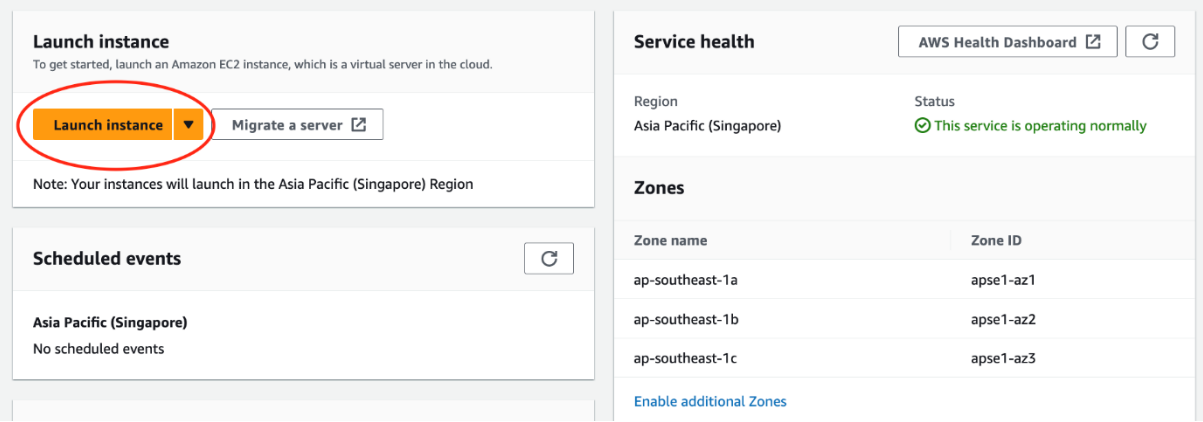


Snapshots (1/2) Info

	Name	Snapshot ID	Volume size	Description
□	–	snap-[redacted]	8 GiB	–
☑	–	snap-[redacted]	8 GiB	webserver-foren

Figure 10 - Copy snapshot of shared EBS forensic snapshot

(c) (i) Launch a EC2 Forensic Instance: Go to the Amazon EC2 dashboard and click "Launch Instance":



Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Note: Your instances will launch in the Asia Pacific (Singapore) Region

Scheduled events

Asia Pacific (Singapore)

No scheduled events

Service health

Region: Asia Pacific (Singapore)

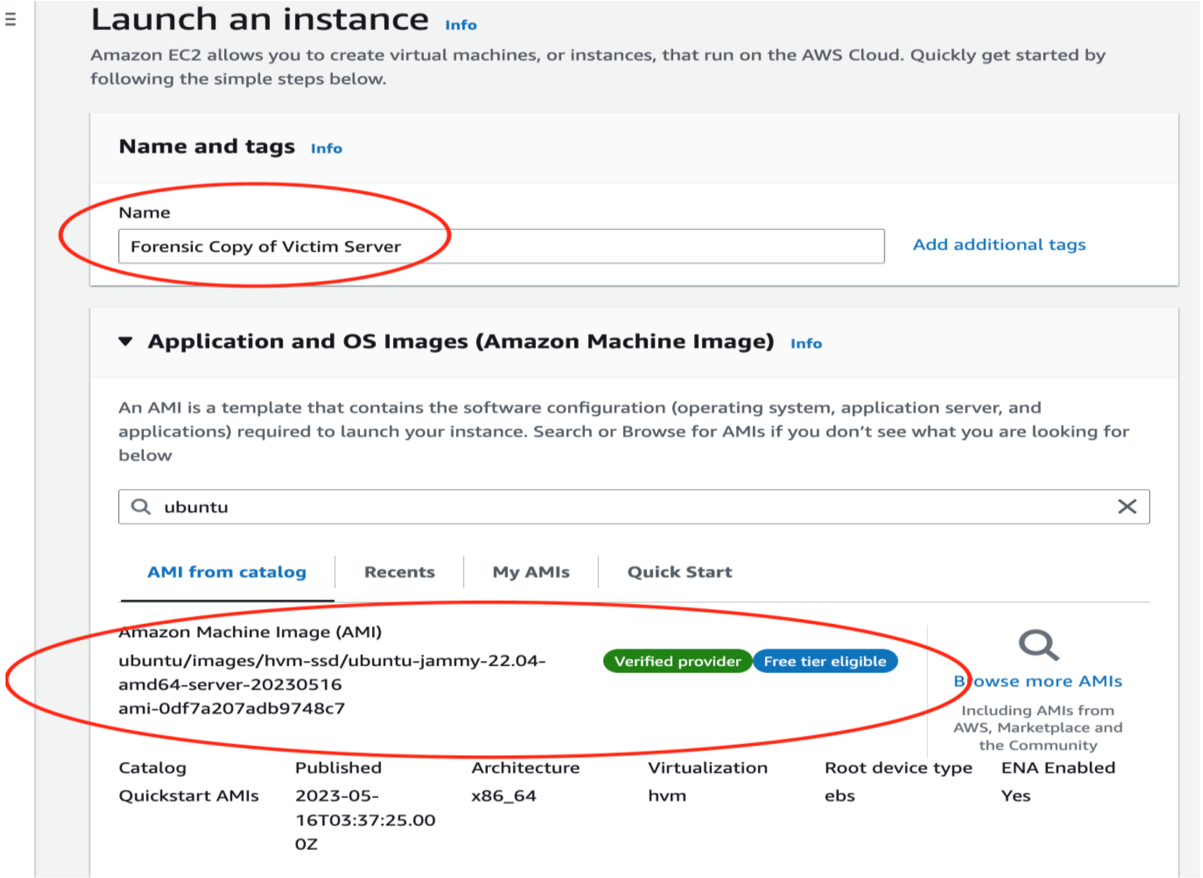
Status: This service is operating normally

Zones

Zone name	Zone ID
ap-southeast-1a	apse1-az1
ap-southeast-1b	apse1-az2
ap-southeast-1c	apse1-az3

Enable additional Zones

(ii) Enter a name for your forensic instance and under the "Choose an Amazon Machine Image (AMI)" section, select the appropriate AMI for your forensic instance:



Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: Forensic Copy of Victim Server

Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

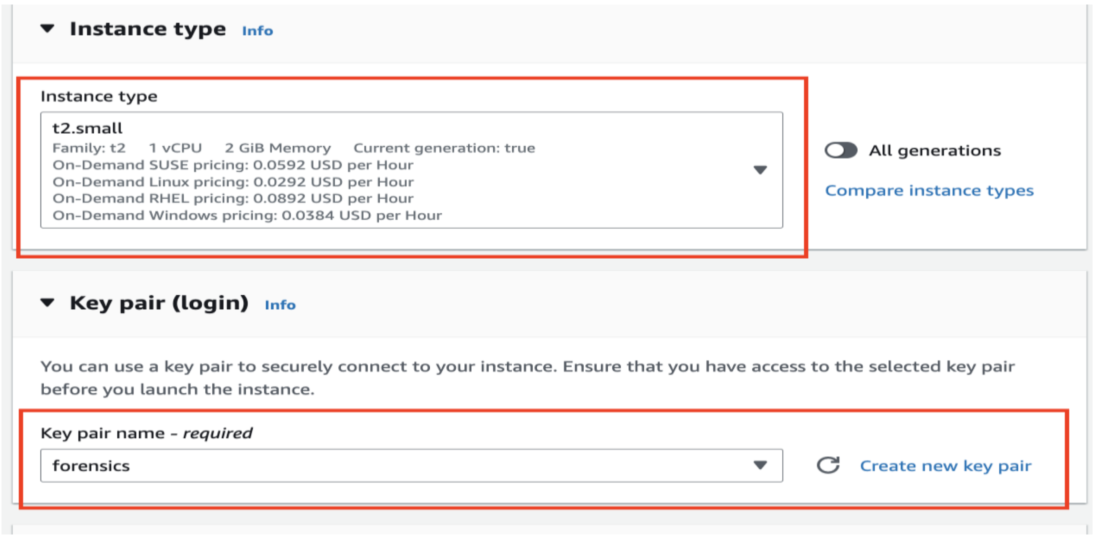
ubuntu

Amazon Machine Image (AMI)
ubuntu/images/hvm-ssd/ubuntu-jammy-22.04-amd64-server-20230516
ami-0df7a207adb9748c7

Catalog	Published	Architecture	Virtualization	Root device type	ENA Enabled
Quickstart AMIs	2023-05-16T03:37:25.000Z	x86_64	hvm	ebs	Yes

Figure 11 - Launch an EC2 forensic instance for analysis

(iii) Configure the instance type that fits your needs for the forensic analysis and choose an existing key pair or create a new one to securely access the forensic instance through SSH:



Instance type

Instance type: t2.small
Family: t2   1 vCPU   2 GiB Memory   Current generation: true
On-Demand SUSE pricing: 0.0592 USD per Hour
On-Demand Linux pricing: 0.0292 USD per Hour
On-Demand RHEL pricing: 0.0892 USD per Hour
On-Demand Windows pricing: 0.0384 USD per Hour

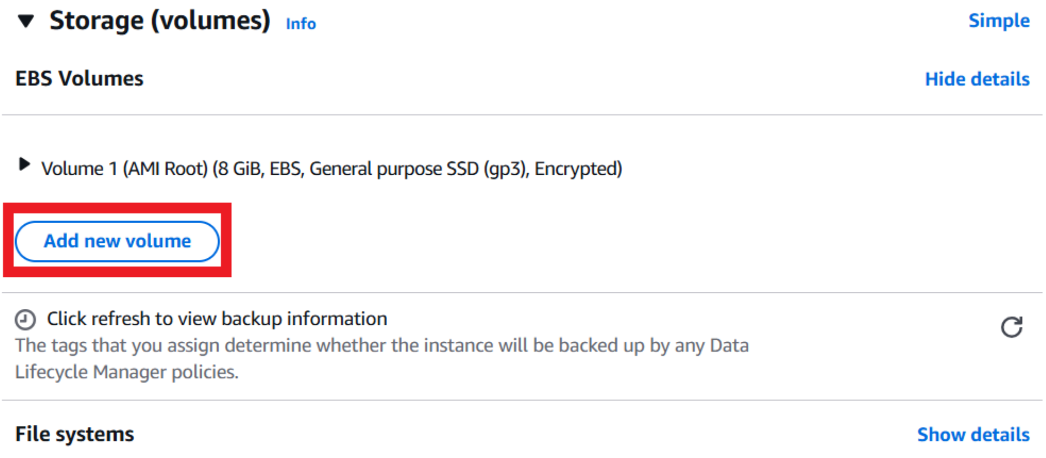
Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*: forensics

(iv) In the "Add Storage" section, you can leave the default root volume size as it is or modify it if needed. However, it is important to attach the forensic image (snapshot) of the original EC2 instance as an additional volume to the forensic instance.

Click on "Add New Volume":



▼ Storage (volumes) Info

Simple

EBS Volumes

Hide details

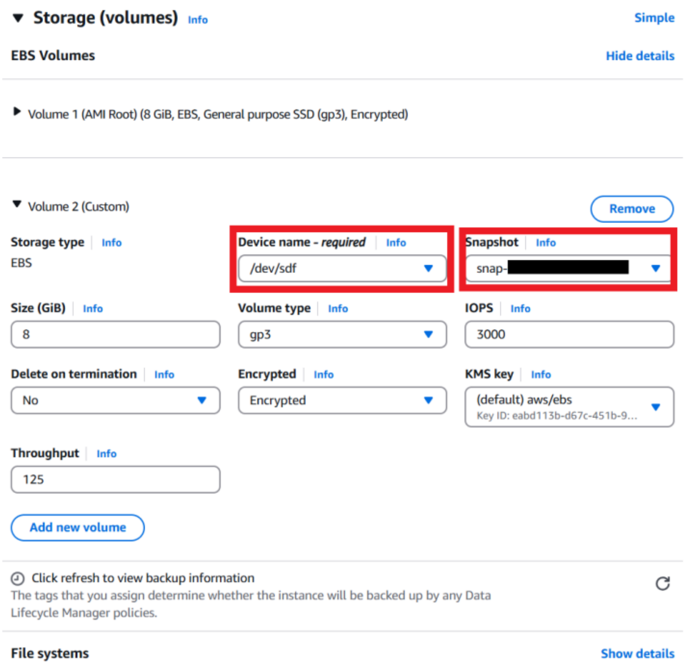
▶ Volume 1 (AMI Root) (8 GiB, EBS, General purpose SSD (gp3), Encrypted)

Click refresh to view backup information
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

File systems

Show details

(v) Select the forensic image snapshot from the list:



▼ Storage (volumes) Info

Simple

EBS Volumes

Hide details

▶ Volume 1 (AMI Root) (8 GiB, EBS, General purpose SSD (gp3), Encrypted)

▼ Volume 2 (Custom)

Storage type: EBS

Device name - *required*: /dev/sdf

Snapshot: snap-[redacted]

Size (GiB): 8

Volume type: gp3

IOPS: 3000

Delete on termination: No

Encrypted: Encrypted

KMS key: (default) aws/ebs
Key ID: eabd113b-d67c-451b-9...

Throughput: 125

Click refresh to view backup information
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

File systems

Show details

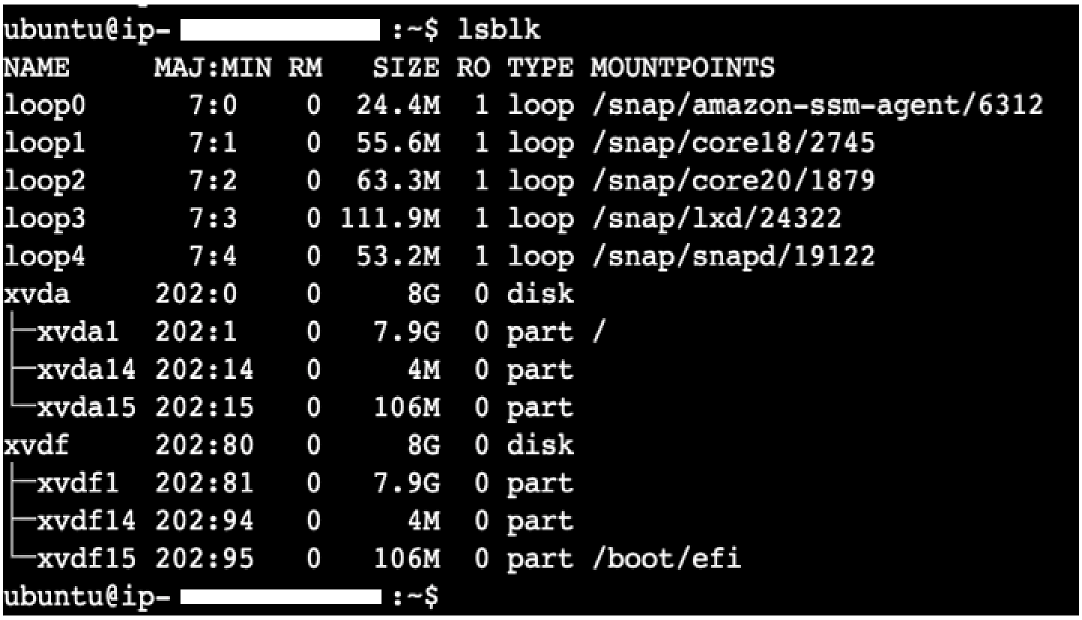
Figure 12 - Add EBS forensic snapshot volume to forensic EC2 instance

(vi) Attach a highly restricted security group to the EC2 Forensic instance.
(vii) Attach a highly restricted IAM EC2 instance role to the EC2 Forensic instance. E.g. attach specific Read-Only permission for a service.
(viii) Click on “Launch Instance” to launch the EC2 Forensic instance:

(d) (i) **IMPORTANT Read-Only Mount:** Once the forensic instance is launched and running, SSH into the forensic instance:

(ii) Use the following command “lsblk” to list all attached volumes on the instance:

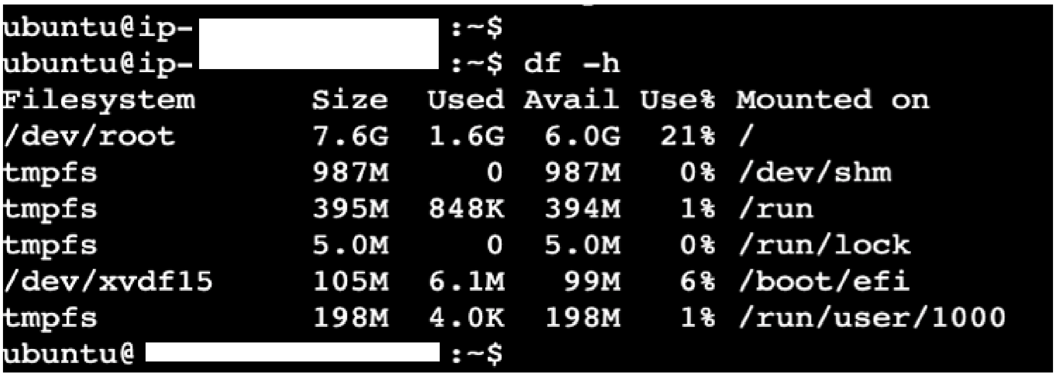
```
lsblk
```



```
ubuntu@ip-[redacted]:~$ lsblk
NAME      MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
loop0       7:0    0  24.4M  1 loop /snap/amazon-ssm-agent/6312
loop1       7:1    0  55.6M  1 loop /snap/core18/2745
loop2       7:2    0  63.3M  1 loop /snap/core20/1879
loop3       7:3    0 111.9M  1 loop /snap/lxd/24322
loop4       7:4    0  53.2M  1 loop /snap/snapd/19122
xvda      202:0    0     8G  0 disk
├─xvda1   202:1    0   7.9G  0 part /
├─xvda14  202:14   0     4M  0 part
└─xvda15  202:15   0   106M  0 part
xvdf      202:80   0     8G  0 disk
├─xvdf1   202:81   0   7.9G  0 part
├─xvdf14  202:94   0     4M  0 part
└─xvdf15  202:95   0   106M  0 part /boot/efi
ubuntu@ip-[redacted]:~$
```

(iii) If there are any existing mount points on the EC2 Forensic instance, check them using:

```
df -h
```



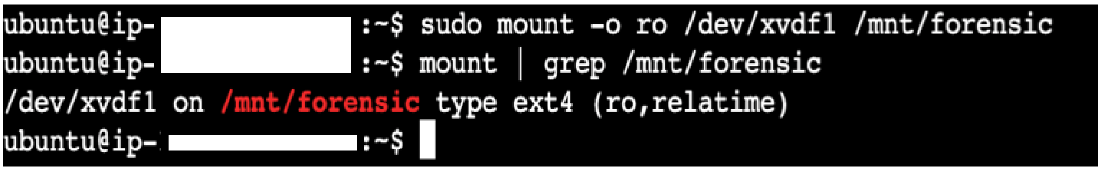
```
ubuntu@ip-[redacted]:~$
ubuntu@ip-[redacted]:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       7.6G  1.6G  6.0G  21% /
tmpfs           987M     0  987M   0% /dev/shm
tmpfs           395M  848K  394M   1% /run
tmpfs           5.0M     0  5.0M   0% /run/lock
/dev/xvdf15     105M  6.1M   99M   6% /boot/efi
tmpfs           198M  4.0K  198M   1% /run/user/1000
ubuntu@[redacted]:~$
```

(iv) If you have not created a specific directory as the mount point, create one to mount the attached forensic EBS volume:

```
sudo mkdir /mnt/forensic
```

(v) Mount the attached forensic EBS volume in read-only mode. For this example, the attached volume is identified as /dev/xvdf1 (this may vary depending on your instance). You can also verify if the volume has been mounted in **read-only** mode as seen below:

```
sudo mount -o ro /dev/xvdf1 /mnt/forensic
```



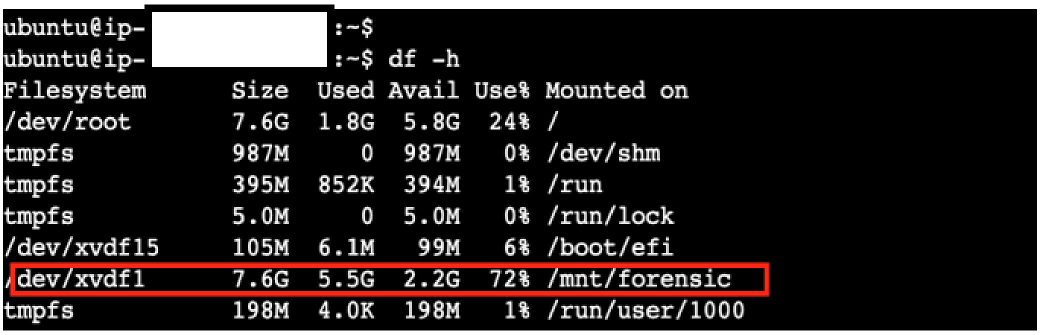
```
ubuntu@ip-[redacted]:~$ sudo mount -o ro /dev/xvdf1 /mnt/forensic
ubuntu@ip-[redacted]:~$ mount | grep /mnt/forensic
/dev/xvdf1 on /mnt/forensic type ext4 (ro,relatime)
ubuntu@ip-[redacted]:~$
```

(vi) Now, the attached "xvdf1" volume is accessible at "/mnt/forensic" in read-only mode, allowing forensic analysis without the risk of modifying the original evidence. Do remember to use the appropriate forensic analysis tools like "Autopsy," "The Sleuth Kit," or "Forensic Toolkit" to analyse the mounted volume such as examining the files, directories, and other artefacts. Do not forget to analyse the log files through the mounted volume using text editors or log analysis tools to identify for potential suspicious activities.

(e) *Optional only if you would like to transfer the mounted EBS volume to your local machine or virtual machine for analysis.

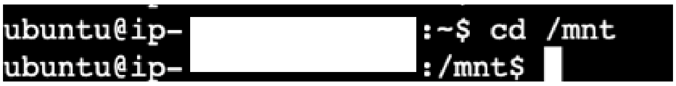
(i) Identify the mount point of the mounted EBS volume before you compress the data. Run the "df -h" command to list all mounted file systems and take note of the one that corresponds to the EBS volume:

```
df -h
```



```
ubuntu@ip-[redacted]:~$
ubuntu@ip-[redacted]:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       7.6G  1.8G  5.8G  24% /
tmpfs           987M     0  987M   0% /dev/shm
tmpfs           395M  852K  394M   1% /run
tmpfs           5.0M     0  5.0M   0% /run/lock
/dev/xvdf15     105M  6.1M   99M   6% /boot/efi
/dev/xvdf1      7.6G  5.5G  2.2G  72% /mnt/forensic
tmpfs           198M  4.0K  198M   1% /run/user/1000
```

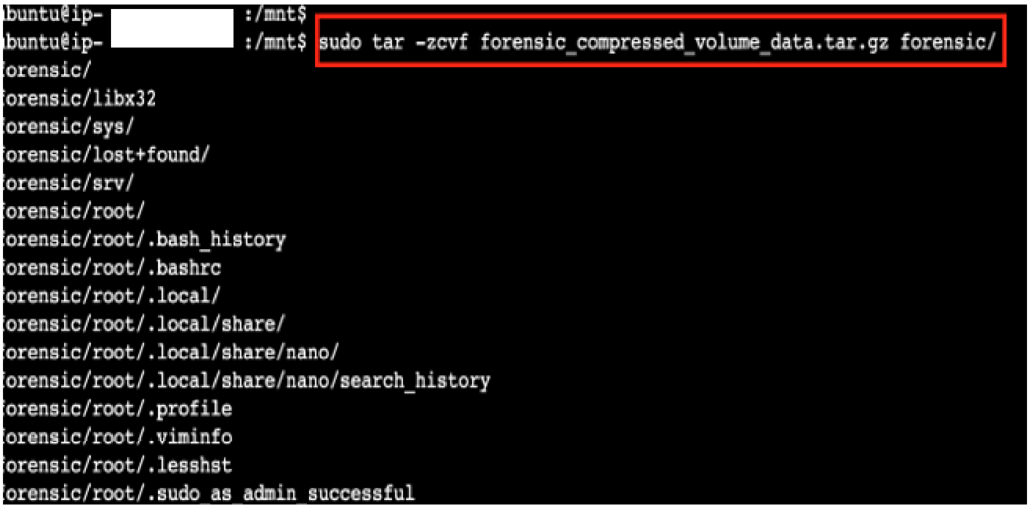
(ii) Navigate to the parent directory that contains the mounted EBS volume data:



```
ubuntu@ip-[redacted]:~$ cd /mnt
ubuntu@ip-[redacted]:/mnt$
```

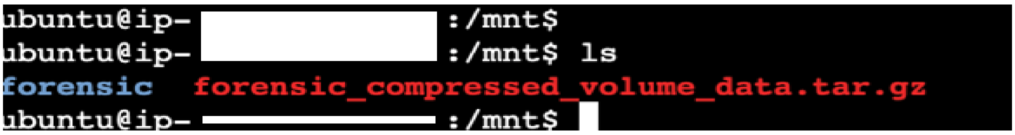
(iii) Prepare the data for transfer by compressing the relevant files and directories from the mounted EBS volume on your forensic EC2 instance. Use the "tar" command with the "-zcvf" parameters options to compress the entire data from the mounted EBS volume. To create a compressed archive (e.g., tar.gz) of the entire mounted volume, use the following command:

```
sudo tar -zcvf forensic_compressed_volume_data.tar.gz
forensic/
```



```
buntu@ip- [redacted] :/mnt$
buntu@ip- [redacted] :/mnt$ sudo tar -zcvf forensic_compressed_volume_data.tar.gz forensic/
orensic/
orensic/libx32
orensic/sys/
orensic/lost+found/
orensic/srv/
orensic/root/
orensic/root/.bash_history
orensic/root/.bashrc
orensic/root/.local/
orensic/root/.local/share/
orensic/root/.local/share/nano/
orensic/root/.local/share/nano/search_history
orensic/root/.profile
orensic/root/.viminfo
orensic/root/.lesshst
orensic/root/.sudo_as_admin_successful
```

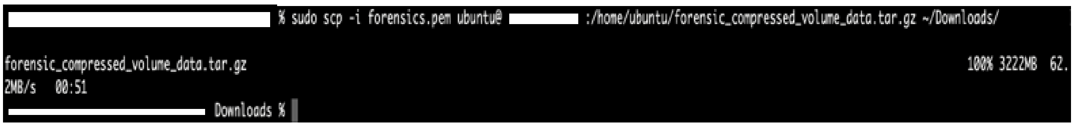
(iv) Depending on the size of the data being compressed, the process may take some time. Once done, you will have a compressed archive file containing all the data from the mounted EBS volume:



```
ubuntu@ip- [redacted] :/mnt$
ubuntu@ip- [redacted] :/mnt$ ls
forensic  forensic_compressed_volume_data.tar.gz
ubuntu@ip- [redacted] :/mnt$
```

(v) Now that the data is compressed into a tar archive, move the file to the home directory. Transfer it to your local machine or virtual machine environment by first copying the tar archive to S3 bucket and securely download from the console:

```
aws s3 cp forensic_compressed_volume_data.tar.gz s3-
evidence-bucket
```

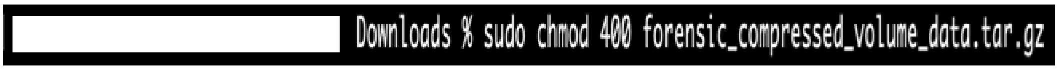


```
[redacted] % sudo scp -i forensics.pem ubuntu@ [redacted] :/home/ubuntu/forensic_compressed_volume_data.tar.gz ~/Downloads/

forensic_compressed_volume_data.tar.gz                                                        100% 3222MB  62.
2MB/s   00:51
[redacted] Downloads %
```

(vi) Ensure that the compressed archive is secured and accessible only by authorized users. You can adjust file permissions using the chmod command if needed:

```
sudo chmod 400 forensic_compressed_volume_data.tar.gz
```



```
[redacted] Downloads % sudo chmod 400 forensic_compressed_volume_data.tar.gz
```

(vii) The compressed archive is now available on your local machine in the specified destination directory. You can then proceed with the forensic analysis on the extracted data from the archive.

2. Amazon Relational Database Service (RDS)

(a) Revoke compromised access

(i) Review Amazon GuardDuty findings for related RDS resources.
(ii) Review CloudTrail and RDS audit logs to pinpoint the specific IAM users or roles used to access the RDS instance during the incident.
(iii) Revoke or modify the permissions of the identified IAM entities to prevent further unauthorised access to the RDS instance.
(iv) Revoke the privileges of the compromised database user within the RDS instance.
(v) Rotate the RDS admin user's (and database system users) credential and securely manage in AWS Secrets Manager (encrypted storage vault).

(b) Assess and remove unauthorised changes

(i) Review RDS query logs to identify unauthorised changes made to the database schema and data.
(ii) Use Database Activity Streams for more insights on the actions performed by the compromised IAM entity.
(iii) Restore database using point-in-time recovery (PITR) or by restoring from a clean backup.

(c) Address data exfiltration or tampering

(i) Analyse RDS access logs to identify unauthorised data access or attempts to exfiltrate data from the database.
(ii) Run database-specific integrity checks to identify any data corruption or inconsistencies that might have resulted from the attack.

(d) Strengthen RDS security

(i) Implement IAM authentication for database access to eliminate the need to store database credentials within any program code.
(ii) Rotate database credentials for master users and other privileged database users.
(iii) Review and tighten database user privileges to ensure only minimum necessary permissions to perform their tasks.
(iv) Review and tighten security group attached to RDS database to restrict the source to specific consumers (either by security group reference or IP address). Avoid wide IP address range.
(v) Refer to Security in Amazon RDS for best practice guidance.

3. Containers (Amazon Elastic Container Service ECS/ Elastic Kubernetes Service EKS)

(a) Revoke compromised access

(i) Review Amazon GuardDuty findings for related ECS/EKS resources.

(ii) Analyse CloudTrail logs and Kubernetes audit logs to pinpoint the specific IAM roles or service accounts used by the compromised containers to assess AWS resources

(iii) Update IAM policies or Kubernetes RBAC configurations to revoke or limit the permissions of the identified roles or service accounts, preventing further unauthorised access.

(b) Assess and address unauthorised changes

(i) Review container logs to identify any unauthorised actions taken within the containers

(ii) Analyse Kubernetes audit logs to identify any changes made to Kubernetes objects during the compromise.

(c) Isolate and redeploy affected containers:

(i) Scale down or pause deployments to temporarily limit the impact and prevent further exploitation.

(ii) Update the packages in container images to resolve known vulnerabilities. Check that dockerfile includes "`RUN apt-get update && apt-get upgrade`" to upgrade the packages in your images.

(iii) Redeploy the affected containers from known good images to ensure that these are free of any malicious code or vulnerabilities.

(iv) Review and tighten security group attached to container cluster to restrict the source to specific consumers (either by security group reference or IP address). Avoid wide IP CIDR ranges.

(v) Review ECS task definition to ensure that containers do not run-in privileged mode and do not have write access to root file system.

(vi) Review container dockerfile configuration to ensure that containers run as non-root user by having USER directive statement.

(vii) Refer to Security in Amazon Elastic Container Service for best practice guidance.

A.3 Differences in Incident Response

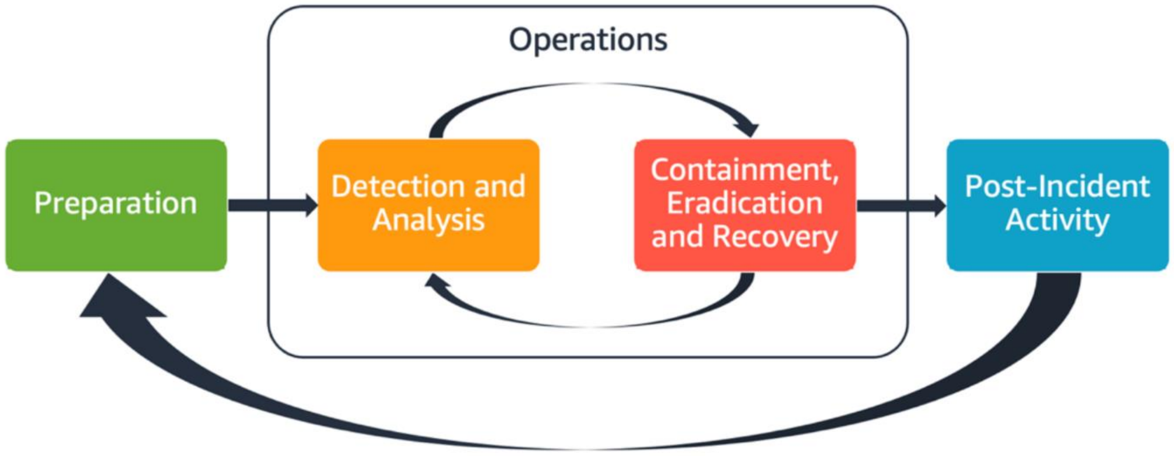
To understand the deviations from traditional on-premises response and their impact to AWS incident response program, we need to understand the core AWS incident response design principles. The foundation of an incident response program in the cloud is Preparation, Operations, and Post-Incident Activity.

- **Preparation** – To ensure your incident response team is ready to identify and address incidents within your AWS cloud environment, it is essential to put in place both preventive (e.g. restrictive IAM permissions and network security) and detective controls (e.g. alerts). Pre-provision appropriate access for the IR team to the required tools to perform log analysis and execute response/containment actions on the impacted services. Prepare comprehensive IR playbooks with clear prescriptive steps on **who** is responsible for taking the **action** and provide clarity on **how** the steps should be taken.

- **Operations** – Handle security events and possible incidents by following the phases of incident response outlined by NIST: detect, analyse, contain, eradicate, and recover.

- **Post-incident activity** – Carry out retrospective review of the security event with focus on how to improve time to detection, time to respond and identify opportunities for automation. Assess what additional threat detection rules and/or countermeasures may be required. Assign the improvement actions to designated owners to foster continuous improvement.

The following diagram shows how these stages align with the NIST incident response lifecycle, but with operations encompassing detection and analysis with containment, eradication, and recovery.



Key differences of incident response in your AWS cloud compared to on-premises

Incident response is a crucial component of a cybersecurity strategy, whether implemented on-premises or in the cloud. The goal is to protect data confidentiality, integrity, and availability through security principles like least-privilege and defense-in-depth. These principles are supported by various incident response patterns, including log retention, selecting alerts based on threat modelling, developing playbooks, and integrating security information and event management (SIEM) systems. However, when these patterns are implemented in the cloud, there are notable differences. This section highlights the key distinctions in incident response within the AWS cloud. It is important to understand how security operations and incident response differ in the cloud to effectively build response capabilities in your AWS cloud environment.

This section provides information on each of these differences, along with the core incident response design principles specific to AWS.

Preparation

In order to effectively prepare for events, incorporating diversity into our teams and response plans can have a significant impact. By assembling a team with diverse perspectives, we increase the likelihood of identifying blind spots that may have gone unnoticed and discovering innovative solutions that may not have been otherwise considered.

Preparation is done across three domains:

- **People** – The process of preparing your personnel for a security incident requires the identification of the appropriate stakeholders involved in incident response and providing them with training on incident response procedures and cloud technology.
- **Process** – To adequately prepare your processes for a security incident, it entails the documentation of architectures, the creation of comprehensive incident response plans, and the development of playbooks to ensure consistent and effective response to security events.
- **Technology** – To effectively prepare your technology for a security incident, it involves configuring access controls, gathering, and monitoring essential logs, implementing efficient alert systems, and establishing response and investigative capabilities. An example would be AWS Organizations which helps oversee and control your AWS resources as they grow and scale. By consolidating your AWS accounts under an AWS Organization, you gain the ability to manage them centrally.

Operations (Core concepts of IR)

Operations is the core component in incident response. It consists the following five phases: *detection*, *analysis*, *containment*, *eradication*, and *recovery*.

Detection	Detect a potential security event.
Analysis	Assess the security event and determine the scope of the security event.

Containment	Reduce and limit the impact scope of the security event.
Eradication	Delete unauthorized resources or artifacts associated with the security event. Implement measures to address the root cause (e.g. application vulnerability) that caused the security event.
Recovery	Restore systems to known safe state and monitor them to ensure that the threat does not resurface.

The below should serve as guidance when customers respond to and operate on security events across various AWS services.

<u>Operations (Detection)</u>

An alert is the core component of the detection phase. It triggers an alert to initiate the incident response procedure based on suspicious activities within the AWS account.

Actions taken by a user, role or an AWS service are recorded as events in AWS CloudTrail. Network VPC Flow Logs capture information about the IP traffic going to and from the network interfaces in the VPC. AWS service logs (e.g. Lambda, WAF) are stored in Amazon CloudWatch log groups and/or S3 for long-term archival storage.

Amazon GuardDuty is a threat detection service that continuously monitors your AWS accounts and workloads for malicious activity. Amazon GuardDuty is designed to analyses multiple AWS data sources including AWS CloudTrail, VPC Flow Logs, Amazon Route53 Resolver DNS query logs, Amazon S3 data events, Amazon EKS and Amazon ECS container audit logs to identify threats. When you enable Amazon GuardDuty across the regions that are in use, you do not need to enable any of the log sources, it automatically starts the analysis and processing of the log sources to generate relevant security findings.

For continuous compliance monitoring and aggregation of security alerts, enable AWS Security Hub with security standards such as AWS Foundational Security Best Practices (FSBP) and external compliance frameworks from National Institute of Standards and Technology (NIST). This provides you with a comprehensive view of your cloud security posture that will be important in responding to various threat scenarios. Misconfigurations and compliance gaps are often exploited by threat actors.

Create a new security group (e.g. “isolate-ec2-sg“) that blocks all inbound and outbound traffic. Remove the default “allow all” rules in both inbound and outbound rules. This security group is used to implement network isolation of Amazon VPC resources.

<u>Operations (Analysis)</u>

Logs, query capabilities, and threat intelligence are a few of the supporting components required by the analysis phase. The logs utilized for detection purposes are also used for analysis and will need to be incorporated and configured with appropriate querying tools.

Operations (Collect and Analyse Forensic Evidence)

Forensics can be used to examine resources across your AWS environment. For an Amazon EC2 instance, it entails preserving and analysing the evidence to identify potential security incidents or unauthorized access.

Operations (Relevant Artefacts to Look Out For)

The investigation phase typically involves considering different types and sources of data relevant to security incidents. For your AWS environment, control planes provide the administrative APIs used to create, read/describe, update, delete, and list (CRUDL) resources. These API calls are captured in Amazon CloudTrail. Users, with direct control over infrastructure, can collect and analyse these CloudTrail logs for local analysis or query them through native AWS services. Data plane is what provides the primary function of the service. Example of data plane activities include getting and putting objects in an S3 bucket (Amazon S3 access logs), Route53 answering DNS queries, Amazon DynamoDB queries and AWS Lambda functions invocations. Direct querying and acquisition of relevant information play a crucial role in investigation. Within the application layer, covering databases and Amazon EC2 instances, users need to collect and analyse data consistently with the resource type and intended analysis approach. For example, when dealing with an Amazon EC2 instance, a step-by-step response order includes acquiring instance metadata, enabling instance protections and tags, acquiring disk (Amazon EBS snapshots), acquiring memory, optional live response/artifact collection, decommissioning the instance, isolating or containing the instance, and making a decision on the responder's choice. This detailed response order reflects the layered responsibilities and access control associated with your AWS resources, allowing users to take specific actions at the infrastructure level for effective incident response.

Containment

Containment is the implementation of remediation measures during a security incident, to reduce the extent of the impact and confine the consequences of unauthorized access within the environment. If the compromised entity is an Amazon VPC resource e.g. Amazon EC2, use network security groups and IAM identity policies to exercise containment. If the compromised entity is an AWS service resource e.g. Amazon S3, Amazon DynamoDB, use IAM resource policies and Amazon VPC endpoint policies for containment.

Source containment is the implementation of filtering or routing mechanisms in an environment to restrict access to resources from a particular source IP address or a specific network range. Abuse of compromised long-term IAM access key by an external actor will require the key to be disabled. Additionally, IAM identity policies are adjusted to restrict access to AWS resources and API services. For instance, set the IAM policy to only allow selected service actions e.g. s3:GetObject for a particular resource e.g. S3 bucket.

Resource containment helps to prevent unauthorized usage of a resource by employing access containment techniques that restrict the functions and IAM principals that have access to the resource. Additionally, IAM resource policies are adjusted to restrict API service calls only from your source Amazon VPC endpoint. For a publicly accessible service, AWS WAF WebACL rules would be attached to the service to block exploitation attempts.

Eradication

Eradication is the effective resolution of the root cause, including suspicious or unauthorized resources in efforts to return the account to a known safe state. In the context of AWS resources, eradication of the root cause will involve analysing available logs or automated tools like Amazon CloudWatch logs and Amazon GuardDuty. These events serve as a basis for determining the necessary remedial actions required to restore the environment to a known secure state. For exploited application vulnerabilities, the effective resolution is to patch the vulnerable application library or code.

Recovery

Recovery involves several steps, including restoring systems to a known secure state, ensuring the safety and integrity of backups before initiating the restoration process, conducting thorough testing to confirm the proper functioning of the systems after restoration, and patching vulnerabilities associated with the security incident.

Post-Incident Activity

As the threat landscape evolves continuously, it is essential for organizations to adapt and enhance their ability to safeguard their environments effectively. A crucial aspect of continuous improvement is to learn from past incidents and simulations to strengthen capabilities in detecting, responding to, and investigating potential security incidents. This iterative process helps mitigate vulnerabilities, reduce response time, and expedite the return to secure operations. Therefore, it also verifies that your organization remains prepared with the latest capabilities and knowledge to effectively respond, no matter what the situation is.

Example of such approaches include the following:

- **Establish a Framework for Learning from Incidents:**
    - Establish a framework for capturing lessons learned to enhance incident response capabilities and prevent the incident from happening again.

- **Use Indicators of Compromise (IOCs):**
    - Establish a framework for gathering, managing, and utilising IOCs to improve detection capabilities and streamline investigations. Integrate IOCs into the analysis and investigation stages of incident response procedures.

- **Education Training**
    - Ensure ongoing education and training for incident response staff and stakeholders.