

## Project2: Cloud Security Builder

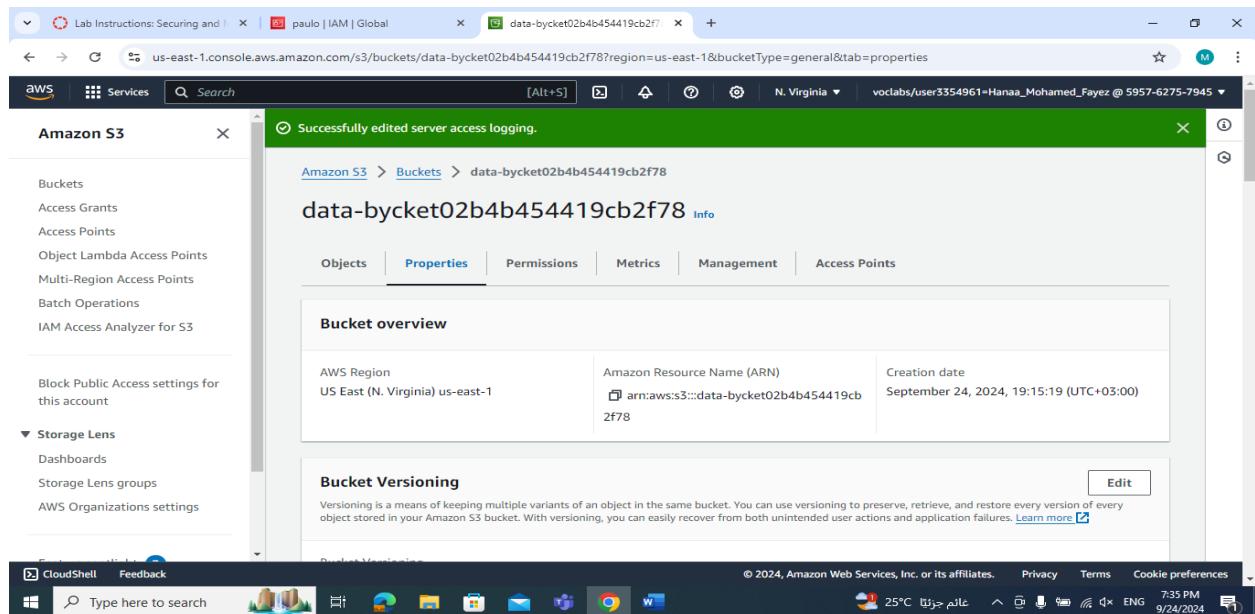
In this project, the challenge is to create resources in AWS and to implement security on them.

- Secure access to objects in an Amazon Simple Storage Service (Amazon S3) bucket.
- Secure network access to virtual network.
- Encrypt data at rest by using AWS Key Management Service (AWS KMS) on an Amazon Elastic Block Store (Amazon EBS) volume.
- Manage encryption keys by using AWS KMS.
- Create a monitoring and incident response system by using Amazon CloudWatch and AWS

### Phase 1: Securing data in Amazon S3

#### Task 1.1: Create a bucket, apply a bucket policy, and test access

Create a new bucket named **data-byucket02b4b454419cb2f78**



## Upload an object to the *data-bucket*

The screenshot shows the AWS S3 console interface. At the top, a green banner displays the message "Upload succeeded". Below this, the "Summary" section shows the destination as "s3://data-bucket02b4b454419cb2f78" with one file uploaded successfully (1 file, 13.0 B (100.00%)) and zero files failed. The "Files and folders" tab is selected, showing a table with one item: "myfile.txt.txt" (text/plain, 13.0 B, Succeeded). The browser's address bar shows the URL <https://us-east-1.console.aws.amazon.com/s3/upload/data-bucket02b4b454419cb2f78?region=us-east-1&bucketType=general>.

Verify the S3 access level of both the paulo and mary user logins

The screenshot shows the AWS S3 console interface for the bucket "data-bucket02b4b454419cb2f78". A service menu is open on the left, showing options like Buckets, Access Grants, Access Points, Object Lambda, Multi-Region Access Points, Batch Operations, and IAM Access Analyzer for S3. The main area shows the "Objects (1) Info" section with one object: "myfile.txt.txt" (txt, 13.0 B, Standard storage class). The browser's address bar shows the URL <https://us-east-1.console.aws.amazon.com/s3/buckets/data-bucket02b4b454419cb2f78?region=us-east-1&tab=objects>.

The screenshot shows the AWS S3 console for a bucket named 'data-bucket02b4b454419cb2f78'. A service menu is open over the left sidebar, with the 'Next' button highlighted. The main content area displays the 'Objects' tab with a message about insufficient permissions to list objects.

## Task 1.2: Enable versioning and object-level logging on a bucket

Enable versioning on the *data-bucket*

The screenshot shows the 'Bucket Properties' page for the 'data-bucket02b4b454419cb2f78' bucket. Under the 'Bucket Versioning' section, it is set to 'Enabled'. The 'Multi-factor authentication (MFA) delete' section is set to 'Disabled'. The 'Tags (0)' section shows no tags associated with the resource.

## Enable server access logging on the *data-bucket*

The screenshot shows the AWS S3 console with the 'Server access logging' tab selected. A success message at the top states 'Successfully edited server access logging.' The configuration section shows 'Server access logging' is 'Enabled' and the 'Destination bucket' is set to 's3://s3-objects-access-log-02b4b454419cb2f78'. A note indicates that requests for access to the bucket can be checked via CloudWatch. Below this, the 'AWS CloudTrail data events' section is shown, with a note to configure CloudTrail data events to log Amazon S3 object-level API operations. The Windows taskbar at the bottom shows various pinned icons and the date/time as 9/25/2024.

## Task 1.3: Implement the S3 Inventory feature on a bucket

The screenshot shows the AWS S3 console with the 'Inventory configurations' tab selected. A success message at the top states 'Inventory successfully created.' It notes that it may take up to 48 hours to deliver the first report. Another message indicates a 'Bucket policy successfully created' with a link to view it. The 'Create inventory configuration' button is highlighted in orange. The table below shows one existing inventory configuration named 'Inventory' with details: Enabled, Entire bucket scope, destination s3://s3-inventory..., daily frequency, and Apache Parquet format. The Windows taskbar at the bottom shows various pinned icons and the date/time as 9/24/2024.

## Task 1.4: Confirm that versioning works as intended

The screenshot shows the AWS S3 console interface. On the left, the navigation pane includes 'Buckets', 'Storage Lens', and 'CloudShell'. The main area displays 'Objects (3) Info' for a bucket named 'data-bycket02b4b454419cb2f78'. A table lists three objects: 'customers.csv.xlsx' (version fMy7rTUCI74, type xlsx, size 9.3 KB), 'customers.csv.xlsx' (version JAyuUrnwKAv, type xlsx, size 9.3 KB), and 'myfile.txt' (version a03bmjbeqv5, type txt, size 13.0 B). The 'Show versions' toggle is turned on.

## Task 1.5: Confirm object-level logging and query the access logs by using Athena

Create an S3 bucket named athena-results-9875

The screenshot shows the AWS S3 console interface. The navigation pane includes 'Buckets', 'Storage Lens', and 'CloudShell'. The main area displays 'General purpose buckets (6) Info' for the US East (N. Virginia) region. A table lists six buckets: 'athena-results-9875' (selected, created on September 24, 2024, 19:54:58 UTC+03:00), 'aws-config-02b4b454419cb2f78' (created on September 24, 2024, 19:06:43 UTC+03:00), 'cloudwatch-logs-02b4b454419cb2f78' (created on September 24, 2024, 19:06:43 UTC+03:00), 'data-bycket02b4b454419cb2f78' (created on September 24, 2024, 19:15:19 UTC+03:00), 's3-inventory-02b4b454419cb2f78' (created on September 24, 2024, 19:06:43 UTC+03:00), and 's3-objects-access-log-02b4b454419cb2f78' (created on September 24, 2024, 19:06:43 UTC+03:00).

## Create an Athena table from the access logs

## Estimate cost of S3 and Athena

The screenshot shows the AWS Pricing Calculator interface. On the left, there's a summary table with 'Upfront cost' at 0.00 USD and 'Monthly cost' at 1.75 USD, totaling 21.00 USD over 12 months. This total includes the upfront cost. To the right, there's a 'Getting Started with AWS' section featuring 'Get started for free' and 'Contact Sales' buttons. Below this is a 'My Estimate' section where users can manage their service configurations. The table in 'My Estimate' lists two services: 'Amazon Simple S...' and 'Amazon Athena'. The 'Amazon Simple S...' entry has an 'Upfront cost' of 0.00 USD and a 'Monthly cost' of 0.70 USD, located in the 'US East (N. Virginia)' region with 'S3 Standard stor...' config. The 'Amazon Athena' entry has an 'Upfront cost' of 0.00 USD and a 'Monthly cost' of 1.05 USD, also in the 'US East (N. Virginia)' region with 'Total number of s...' config.

Service Name	Status	Upfront cost	Monthly cost	Description	Region	Config Summary
Amazon Simple S...	-	0.00 USD	0.70 USD	-	US East (N. Virginia)	S3 Standard stor...
Amazon Athena	-	0.00 USD	1.05 USD	-	US East (N. Virginia)	Total number of s...

The screenshot shows an Excel spreadsheet titled "My Estimate (2) - Excel". The spreadsheet contains several sections of data:

- Estimate summary:** Includes a table with columns for Upfront and Monthly costs, totaling 21 USD.
- Detailed Estimate:** A table showing AWS services and their costs across different regions (US East, US West, EU West). It includes rows for S3 Standard, Data Transfer, and Amazon Athena.
- Acknowledgement:** A note stating that the estimate does not include taxes and actual fees depend on usage.

The Excel ribbon at the top shows tabs like File, Home, Insert, Page Layout, Formulas, Data, Review, View, and Help. The status bar at the bottom indicates the file is "Ready" and shows system information like temperature (25°C), battery level (84%), and date (10/10/2024).

## Phase 2: Securing VPCs

### Task 2.2: Create a VPC flow log

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with options like EC2 Global View, Virtual private cloud, Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, and NAT gateways. The main area displays the "Your VPCs" section with three VPCs listed:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
vpc-0fd6a728dfe7b25bd	vpc-0fd6a728dfe7b25bd	Available	172.31.0.0/16	-
LabVPC	vpc-0081f2f9a8e611e04	Available	10.1.0.0/16	-
NetworkFirewallVPC	vpc-0122add4c0a6990a0	Available	10.1.0.0/16	-

Below this, the "Flow logs" section shows one flow log named "LabVPCFlowLogs" with a flow log ID of "fl-00534e7f27d65ad13". The flow log is associated with the "cloud-watch-logs" destination type. The status bar at the bottom indicates the user is "voclabs/user3354961=Hanaa\_Mohamed\_Fayez @ 5957-6275-7945".

The screenshot shows the AWS CloudWatch Log Groups interface. The left sidebar is titled "CloudWatch" and includes sections for Favorites and recents, Dashboards, Alarms, Logs (Log groups, Log Anomalies, Live Tail, Logs Insights, Contributor Insights), Metrics, X-Ray traces, Events, Application Signals, and Network monitoring. The main content area shows the log group "LabVPCFlowLogs" with the log stream "eni-083a2acdc166702ed-all". The "Log events" table lists several log entries, each with a timestamp and a detailed message. The search bar at the top contains "54.145.129.57". The bottom status bar indicates the date as 9/24/2024 and the time as 8:58 PM.

## Task 2.3: Access the WebServer instance from the internet and review VPC flow logs in CloudWatch

### Task 2.4: Configure route table and security group settings

The screenshot shows the AWS EC2 Security Group configuration page for the "WebServerSecurityGroup". The left sidebar includes sections for Capacity Reservations, Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), and Load Balancing (Load Balancers). The main content area displays the security group details: Owner (595762757945), Inbound rules count (3 Permission entries), and Outbound rules count (1 Permission entry). Below this, the "Inbound rules (3)" table is shown, listing three rules: one for SSH (TCP port 22, source 156.192.196.46/32), one for HTTP (TCP port 80, source 0.0.0.0/0), and one for Custom TCP (TCP port 8080, source 0.0.0.0/0).

## Task 2.5: Secure the WebServerSubnet with a network ACL

The screenshot shows the AWS VPC Network ACL configuration page. The left sidebar shows the VPC dashboard and a list of subnets under the Virtual private cloud section. The main area displays the Network ACL settings for subnet `subnet-090fde0410ed7c3d`. The **Network ACL: acl-05491d0a0b936e2e6** is selected. The **Inbound rules (4)** table lists four rules:

Rule number	Type	Protocol	Port range	Source	Allow/Deny
99	SSH (22)	TCP (6)	22	0.0.0.0/0	Allow
100	All traffic	All	All	0.0.0.0/0	Deny
101	HTTP (80)	TCP (6)	80	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

The **Outbound rules (2)** table lists two rules:

Rule number	Type	Protocol	Port range	Destination	Allow/Deny
100	All traffic	All	All	0.0.0.0/0	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Access the webpage

The screenshot shows a browser window with the address bar set to `35.172.123.230`. The page content displays the message `Hello world from WebServer!`. The browser status bar indicates the connection is `Not secure`.

## Task 2.7: Create a network firewall

Create a firewall named NetworkFirewall

The screenshot shows the AWS VPC Firewall dashboard. A green success message box displays two items: "You've successfully created firewall NetworkFirewall" and "You've successfully created firewall policy FirewallPolicy". Below the message box, the "Firewalls" section lists one firewall named "NetworkFirewall" with a status of "Provisioning" and "Pending". The navigation bar at the top includes tabs for Lab Instructions, Instances, AWS Systems Manager, Route Tables, CloudWatch Metrics, Network Firewall, Subnets, New Tab, and a plus sign for creating new resources.

## Task 2.8: Create route tables

Create a new route table named IGW-Ingress-Route-Table in the *NetworkFirewallVPC*.

The screenshot shows the AWS VPC Route Tables details page for the route table "rtb-06ca659d33fe5fbb8 / IGW-Ingress-Route-Table". The "Details" tab is selected, showing the route table ID, VPC, and associations. The "Routes" tab is selected, displaying two routes: one for destination 10.1.0.0/16 targeting "local" (Status: Active, Propagated: No) and another for destination 10.1.3.0/28 targeting "vpce-0e2811f5cd0590145" (Status: Active, Propagated: No). The navigation bar at the top includes tabs for Lab Instructions, VpcDetails | VPC Console, RouteTableDetails | VPC Console, Instance details | EC2 | us-east-1, and a plus sign for creating new resources.

**VPC dashboard**

**Route tables**

**rtb-06ca659d33fe5fbb8 / IGW-Ingress-Route-Table**

**Details**

Route table ID rtb-06ca659d33fe5fbb8	Main No	Explicit subnet associations -	Edge associations igw-0a9d5a7164051c437 / NetworkFirewallIG
VPC vpc-0122add4c0a6990a0   NetworkFirewallVPC	Owner ID 595762757945		

**Associated internet gateways (1)**

ID	State	VPC	Owner
igw-0a9d5a7164051c437 / NetworkFirewallVPC	Attached	vpc-0122add4c0a6990a0	595762757945

Create another route table in *NetworkFirewallVPC* for the *FirewallSubnet*.

**VPC dashboard**

**Route tables**

**rtb-0c0e1ef937089586e / Firewall-Route-Table**

**Details**

Route table ID rtb-0c0e1ef937089586e	Main No	Explicit subnet associations subnet-095dabb41ea000178 / WebServer2Subnet	Edge associations -
VPC vpc-0122add4c0a6990a0   NetworkFirewallVPC	Owner ID 595762757945		

**Routes (2)**

Destination	Target	Status	Propagated
0.0.0.0/0	igw-0a9d5a7164051c437	Active	No
10.1.0.0/16	local	Active	No

The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with options like EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables), and NetworkFirewallVPC. The main area displays a success message: "You have successfully updated subnet associations for rtb-0c0e1ef937089586e / Firewall-Route-Table". Below this, the "Details" tab is selected, showing route table ID (rtb-0c0e1ef937089586e), Main (No), Owner ID (595762757945), and Explicit subnet associations (subnet-0f783980e291c4439 / FirewallSubnet). The "Subnet associations" tab is active, showing one entry: FirewallSubnet associated with subnet-0f783980e291c4439 and IPv4 CIDR 10.1.1.0/28. The "Routes" tab shows two routes: 0.0.0.0/0 targetting vpce-0e2811f5cd0590145 (Status: Active, Propagated: No) and 10.1.0.0/16 targetting local (Status: Active, Propagated: No).

Create another route table in *NetworkFirewallVPC* for the *Webserver2Subnet*.

The screenshot shows the AWS VPC dashboard. The sidebar includes EC2 Global View, Virtual private cloud (Your VPCs, Subnets, Route tables), and NetworkFirewallVPC. A success message is displayed: "You have successfully updated subnet associations for rtb-0c56e717d7305a441 / WebServer2-Route-Table". The "Details" tab is selected, showing route table ID (rtb-0c56e717d7305a441), Main (No), Owner ID (595762757945), and Explicit subnet associations (subnet-095dabb41ea000178 / WebServer2Subnet). The "Routes" tab is active, showing two routes: 0.0.0.0/0 targetting vpce-0e2811f5cd0590145 (Status: Active, Propagated: No) and 10.1.0.0/16 targetting local (Status: Active, Propagated: No).

The screenshot shows the AWS VPC dashboard. A green success message at the top states: "You have successfully updated subnet associations for rtb-0c56e717d7305a441 / WebServer2-Route-Table. rtb-0c56e717d7305a441 / WebServer2-Route-Table". The main pane displays the details of a route table, including its ID (rtb-0c56e717d7305a441), VPC (vpc-0122add4c0a6990a0), and subnet associations. The "Subnet associations" tab is selected, showing one explicit association to "WebServer2Subnet" (subnet-095dabb41ea000178) with an IPv4 CIDR of 10.1.3.0/28. There are also tabs for "Routes", "Edge associations", "Route propagation", and "Tags".

## Task 2.9: Configure logging for the network firewall

Create a CloudWatch log group named NetworkFirewallVPCLogs

The screenshot shows the CloudWatch Log Groups page. The left sidebar includes sections for Favorites and recents, Alarms, Logs (with Log groups selected), Metrics, X-Ray traces, Events, Application Signals, and Network monitoring. The main pane shows a list of log groups, with "NetworkFirewallVPCLogs" selected. The table columns include Log group, Log class, Anomaly d..., Da..., Se..., and Retent. The "NetworkFirewallVPCLogs" row has "Configure" links for Log class, Anomaly detection, Data sampling, and Retention. Other log groups listed are "/aws/lambda/c133601a5382972l7710711t1-AdjustA...", "/aws/lambda/c133601a5382972l7710711t1-AdjustB...", and "LabVPCFlowLogs".

The screenshot shows the AWS VPC Firewall dashboard. At the top, there are tabs for 'Lab Instructions: Securing a...', 'Firewall details | VPC Manag...', 'RouteTableDetails | VPC Con...', and 'CloudWatch | us-east-1'. The main content area has a 'Services' bar with 'VPC dashboard' selected. A search bar and a 'Delete protection' section (disabled) are visible. Below this is a 'Logging' section where Network Firewall generates logs for stateful rule groups. It includes fields for Log type (Flow, Alert), Alert log destination (CloudWatch log group - NetworkFirewallVPCLogs), Flow log destination (CloudWatch log group - NetworkFirewallVPCLogs), and TLS log destination (Not configured). There are also sections for 'Customer managed key' (Key type: AWS owned key) and 'Firewall tags (0)'. On the left sidebar, under 'Virtual private cloud', are links for Your PCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, and NAT gateways. A 'Filter by VPC' dropdown is also present.

## Task 2.10: Configure the firewall policy and test access

Configure the stateful rule group with the name NetworkFirewallVPCRuleGroup

The screenshot shows the AWS VPC dashboard. A green success message at the top states: "You've successfully created rule group NetworkFirewallVPCRuleGroup". Below this, under "Stateful rule evaluation order and default actions", it shows "Rule order Strict order" and "Default actions Drop established". In the "Stateful rule groups (1/1)" section, there is one entry: Priority 1, Name NetworkFirewallVPCRuleGroup, Capacity 100, Is managed? No, and Run in alert mode? Not available. At the bottom, two sections show capacity units consumed: 0/30,000 for stateless rule groups and 100/30,000 for stateful rule groups.

The screenshot shows a web browser window with the URL 3.95.93.75. The page content is "Hello world from WebServer2!". The browser taskbar shows multiple tabs, including Lab Instructions, Instances, RouteTab, Cloud9Insta, vpcs | VPC, SubnetDe, and New Tab.



## Cost estimate to secure a VPC with a network firewall

The screenshot shows the AWS Pricing Calculator interface. At the top, it displays the total cost: **9,480.84 USD**, which includes upfront cost. Below this, there is a table titled "My Estimate" listing various AWS services and their costs. The table includes columns for Service Name, Status, Upfront cost, Monthly cost, Description, Region, and Config Summary. The services listed are Amazon EC2, Amazon Virtual P..., AWS Data Transfer, and AWS Network Fir... The bottom of the calculator shows copyright information and a privacy link.

Service Name	Status	Upfront cost	Monthly cost	Description	Region	Config Summary
Amazon EC2	-	0.00 USD	3.87 USD		US East (N. Virginia)	Tenancy (Shared I...)
Amazon Virtual P...	-	0.00 USD	348.20 USD		US East (N. Virginia)	Working days per ...
AWS Data Transfer	-	0.00 USD	20.48 USD		US East (N. Virginia)	DT Inbound: Inter...
AWS Network Fir...	-	0.00 USD	417.52 USD		US East (N. Virginia)	Number of AWS ...

The screenshot shows an Excel spreadsheet titled "My Estimate (3) - Excel". The spreadsheet contains two main sections: "Estimate summary" and "Detailed Estimate". The "Estimate summary" section shows the total cost of 9480.84 USD. The "Detailed Estimate" section provides a breakdown of the cost by service, including Amazon EC2, VPN Connection, IPAM, AWS Data Transfer, and AWS Network Firewall. The bottom of the spreadsheet includes an acknowledgement and a note about AWS Pricing Calculator limitations.

Estimate summary						
Upfront	Monthly	Total 12 months cost	Currency			
0	790.07	9480.84	USD			
* Includes upfront cost						

Detailed Estimate						
Group	Region	Description	Service	Upfront	Monthly	First 12 m
My Estimate	US East (N. Virginia)	Amazon EC2		0	3.866	46.39 USD
My Estimate	US East (N. Virginia)	VPN Connection		0	348	4176 USD
My Estimate	US East (N. Virginia)	IPAM		0	0.2	2.4 USD
My Estimate	US East (N. Virginia)	AWS Data Transfer		0	20.48	245.76 USD
My Estimate	US East (N. Virginia)	AWS Network Firewall		0	417.52	5010.24 USD

Notes at the bottom:

- Acknowledgement: AWS Pricing Calculator provides only an estimate of your AWS fees and doesn't include any taxes that might apply. Your actual fees depend on a variety of factors, including your actual usage of AWS services.
- \* AWS Pricing Calculator provides only an estimate of your AWS fees and doesn't include any taxes that might apply. Your actual fees depend on a variety of factors, including your actual usage of AWS services.

## Phase 3: Securing AWS resources by using AWS KMS

### Task 3.1: Create a customer managed key and configure key rotation

Create an AWS KMS with the alias (name) of MyKMSKey

The screenshot shows the AWS KMS console interface. On the left, a navigation pane lists "Key Management Service (KMS)" under "Customer managed keys". The main area displays a table titled "Customer managed keys (1/1)" with one item: "MyKMSKey" (Key ID: 30dd6dd5-d799-4386-9c62-195ad2ed7cc3). The table includes columns for Aliases, Key ID, Status, Key type, Key spec, and Key usage.

**Customer managed keys (1/1)**

Aliases	Key ID	Status	Key type	Key spec	Key usage
MyKMSKey	30dd6dd5-d799-4386-9c62-195ad2ed7cc3	Enabled	Symmetric	SYMMETRIC_D...	Encrypt and decr...

**Key policy | Key ID: 30dd6dd5-d799-4386-9c62-195ad2ed7cc3**

The screenshot shows the "General configuration" tab for the key. It displays the following details:

Alias	Status	Creation date
MyKMSKey	Enabled	Sep 27, 2024 19:14 GMT+3
ARN	Description	Regionality
arn:aws:kms:us-east-1:595762757945:key/30dd6dd5-d799-4386-9c62-195ad2ed7cc3	-	Single Region

**Key policy**

Switch to policy view

At the bottom of the browser window, the URL is https://us-east-1.console.aws.amazon.com/kms/keys/30dd6dd5-d799-4386-9c62-195ad2ed7cc3/keyPolicy?.

Configure AWS KMS key rotation on the new key so that it is automatically rotated every year

The screenshot shows the AWS KMS console with the 'Key Management Service (KMS)' service selected. A key named 'MyKMSKey' is being viewed. The 'Key rotation' tab is active, showing the following configuration:

Setting	Value
Status	Enabled
Rotation period	365
Date of last automatic rotation	-
Next rotation date	Sep 28, 2025

At the bottom of the page, there is an 'Edit' button.

## Task 3.2: Update the AWS KMS key policy and analyze an IAM policy

Modify the key policy for *MyKMSKey* so that it allows the *sofia* IAM user to use the key

The screenshot shows the AWS KMS console with the 'Key Management Service (KMS)' service selected. A key named 'MyKMSKey' is being viewed. The 'Key policy' tab is active, displaying the following JSON policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::595762757945:role/voclabs",
                    "arn:aws:iam::595762757945:user/sofia"
                ]
            },
            "Action": [
                "kms:UntagResource",
                "kms:RotateKeyOnDemand"
            ],
            "Resource": "*"
        }
    ]
}
```

### Task 3.3: Use AWS KMS to encrypt data in Amazon S3

```
<Error>
<Code>AccessDenied</Code>
<Message>User: arn:aws:iam::595762757945:user/paulo is not authorized to perform: kms:Decrypt on resource: arn:aws:kms:us-east-1:595762757945:key/30dd6dd5-d799-4386-9c62-195ad2ed7cc3 because no identity-based policy allows the kms:Decrypt action.</Message>
<RequestId>2HZFT01HVCXKJQCJ</RequestId>
<HostId>7yXM1scVdqvhbYlo4vrnA/kh3AkmmU0TbXcuvBU1+unoPQ4V61lUq6ejMEPMETZHKEewL0zy6g=</HostId>
</Error>
```

### Task 3.4: Use AWS KMS to encrypt the root volume of an EC2 instance

Instances (1/4) **Info** Last updated 1 minute ago

Name	Instance ID	Instance state	Instance type	Status check
aws-cloud9-Cloud9Instance-2c2235c94db342...	i-0238f45add8990ce	Running	t2.micro	2/2 checks passed
WebServer	i-0df07c73da9ab827f1	Running	t2.micro	2/2 checks passed
WebServer2	i-0d5b3de5b87ea3e7f	Running	t2.micro	2/2 checks passed
<b>EncryptedInstance</b>	<b>i-0ce794106a9270f92</b>	<b>Running</b>	<b>t2.micro</b>	<b>Initializing</b>

**i-0ce794106a9270f92 (EncryptedInstance)**

Root device name	Root device type	EBS optimization
/dev/xvda	EBS	disabled

**Block devices**

Volume size (GiB)	Attachment status	Attachment time	Encrypted	KMS key ID	Delete on term.
3	Attached	2024/09/28 02:03 GMT+3	Yes	30dd6dd5-d799-4386-9c62-195ad2ed7...	Yes

## Task 3.5: Use AWS KMS envelope encryption to encrypt data in place

The screenshot shows a CloudShell terminal window with the following session history:

```
i-0d5b3de3b87ea3e7f (WebServer2)
PublicIPs: 3.95.93.75 PrivateIPs: 10.1.3.4

[ec2-user@webserver2 ~]$ mv unencrypted.txt unencrypted2.txt
mv: target 'unencrypted2.txt' is not a directory
[ec2-user@webserver2 ~]$ rm unencrypted2.txt
[ec2-user@webserver2 ~]$ rm unencrypted.txt
[ec2-user@webserver2 ~]$ cat > data_unencrypted.txt
[ec2-user@webserver2 ~]$ echo "Let's encrypt these file contents. Sensitive data here." > data_unencrypted.txt
[ec2-user@webserver2 ~]$ cat data_unencrypted.txt
[ec2-user@webserver2 ~]$
```

The terminal then shows the command to encrypt the file:

```
[ec2-user@webserver2 ~]$ aws kms encrypt --key-id arn:aws:kms:us-east-1:123456789012:key/12345678-1234-1234-1234-123456789012 --plaintext file://data_unencrypted.txt --ciphertext-blob file://data_encrypted.txt
```

Finally, the encrypted file is listed:

```
[ec2-user@webserver2 ~]$ ls -l
total 12
-rw-r--r-- 1 ec2-user ec2-user 12 9月 28 10:28 data_encrypted.txt
-rw-r--r-- 1 ec2-user ec2-user 12 9月 28 10:28 data_unencrypted.txt
```

## Task 3.6: Use AWS KMS to encrypt a Secrets Manager secret

The screenshot shows the AWS Secrets Manager console with the following message:

You successfully stored the secret mysecret. To show it in the list, choose Refresh.  
Use the sample code to update your applications to retrieve this secret.

The secrets list table shows one item:

Secret name	Description	Last retrieved (UTC)
mysecret	-	-

The CloudShell taskbar at the bottom shows the following information:

CloudShell Feedback Type here to search 24°C 2:28 AM 9/28/2024 © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

```

[ec2-user@webserver2 ~]$ aws secretsmanager get-secret-value
usage: aws [options] <command> <subcommand> [<subcommand> ...] [parameters]
To see help text, you can run:
  aws help
  aws <command> help
  aws <command> <subcommand> help

aws: error: the following arguments are required: --secret-id

[ec2-user@webserver2 ~]$ aws secretsmanager get-secret-value --secret-id mysecret
{
    "ARN": "arn:aws:secretsmanager:us-east-1:595762757945:secret:mysecret-UwQgnD",
    "Name": "mysecret",
    "VersionId": "5cd29ee5-75d1-4017-adb8-d2a1a21ae91f",
    "SecretString": "{\"secret\":\"my secret data\"}",
    "VersionStages": [
        "AWSCURRENT"
    ],
    "CreatedDate": "2024-09-27T23:33:18.573000+00:00"
}
[ec2-user@webserver2 ~]$ 

```

i-0d5b3de3b87ea3e7f (WebServer2)  
PublicIPs: 3.95.93.75 PrivateIPs: 10.1.3.4

## Estimate Cost for using AWS KMS

**My Estimate**

Upfront cost	Monthly cost	Total 12 months cost
0.00 USD	1.00 USD	12.00 USD Includes upfront cost

**My Estimate**

Service Name	Status	Upfront cost	Monthly cost	Description	Region	Config Summary
AWS Key Manage...	-	0.00 USD	1.00 USD		US East (N. Virginia)	Number of custo...

The screenshot shows an Excel spreadsheet titled "My Estimate (4) - Excel". The spreadsheet contains several rows of data:

- Row 1: "Estimate summary"
- Row 2: "Upfront" and "Monthly" costs.
- Row 3: Total cost of "12 USD".
- Row 4: A note stating "\* Includes upfront cost".
- Row 7: "Detailed Estimate".
- Row 8: Headers for "Group", "Region", "Service", "Upfront", "Monthly", "First 12 m", "Currency", and "Status".
- Row 9: Data for "My Estimate US East (N. Virginia)" showing AWS Key Management Service with values 0, 1.003, 12.04 USD, and Status "Configuration summary".

The status bar at the bottom indicates "Accessibility: Unavailable".

## Phase 4: Monitoring and logging

### Task 4.1: Use CloudTrail to record Amazon S3 API calls

Create trail (data-bucket-reads-writes)

The screenshot shows the AWS CloudTrail "Trails" page. A green banner at the top says "Trail successfully created". The table below lists the trail details:

Name	Home region	Multi-region trail	Insights	Organization trail	S3 bucket	Log file prefix	CloudWatch Logs log group	Status
data-bucket-reads-writes	US East (N. Virginia)	Yes	Disabled	No	cloudtrail-logs-0c2aa8f53f0f94f04	-	-	Logging

The status column shows "Logging" with a green checkmark. The status bar at the bottom indicates "CloudShell Feedback" and "Type here to search".

Upload the *customer-data.csv* file to *data-bucket*

The screenshot shows the AWS S3 console with a green success message: "Upload succeeded". The summary table indicates 1 file uploaded successfully (8.4 KB) and 0 files failed. The "Files and folders" tab shows a single file named "customer-da...". The Windows taskbar at the bottom includes icons for CloudShell, Feedback, and various Microsoft applications.

Use the CloudTrail console to create an Athena table that describes the format of the data in the *cloudtrail-logs* S3 bucket.

The screenshot shows the AWS Athena console with a query editor. The query is:

```
1 SELECT eventtime, useridentity.principalid, requestparameters, eventname
2 FROM cloudtrail_logs.cloudtrail_logs_0c2aa8f53f0f94f04
3 WHERE
4   eventname in ('PutObject') AND
5   requestparameters LIKE '%customer-data.csv%'
6 limit 10;
```

The results section shows the query ran in 70 ms with a run time of 673 ms and scanned 89.46 KB of data. The Windows taskbar at the bottom includes icons for CloudShell, Feedback, and various Microsoft applications.

The screenshot shows the AWS Athena Query Editor interface. On the left, the sidebar displays a table named "clouptrail\_logs\_clouptrail\_logs\_0c2aa8f53f0f94f04". The main area shows the SQL query "SELECT eventtime, useridentity.principalid, requestparameters, eventname, sourceipaddress, useragent FROM clouptrail\_logs\_clouptrail\_logs\_0c2aa8f53f0f94f04 WHERE eventname in ('GetObject') AND requestparameters LIKE '%customer-data.csv%' limit 10". The results section shows two rows of data:

#	eventtime	principalid	requestparameters
1	2024-10-08T18:15:35Z	AROAUIAATNQLGCYGOZGAU:user3354961=Hanaa_Mohamed_Fayez	{"X-Amz-Date": "2024-10-08T18:15:35Z", "X-Amz-Security-Token": "..."}
2	2024-10-08T18:19:53Z	AROAUIAATNQLGCYGOZGAU:user3354961=Hanaa_Mohamed_Fayez	{"X-Amz-Date": "2024-10-08T18:19:53Z", "X-Amz-Security-Token": "..."}

The screenshot shows the AWS Athena Query Editor interface. On the left, the sidebar displays a table named "clouptrail\_logs\_clouptrail\_logs\_0c2aa8f53f0f94f04". The main area shows the SQL query:

```
1 SELECT eventtime, useridentity.principalid, requestparameters, eventname, sourceipaddress, useragent
2 FROM clouptrail_logs_clouptrail_logs_0c2aa8f53f0f94f04
3 WHERE
4     eventname in ('GetObject') AND
5     requestparameters LIKE '%customer-data.csv%'
6 limit 10
```

The results section shows two rows of data:

#	eventtime	principalid	requestparameters
1	2024-10-08T18:15:35Z	AROAUIAATNQLGCYGOZGAU:user3354961=Hanaa_Mohamed_Fayez	{"X-Amz-Date": "2024-10-08T18:15:35Z", "X-Amz-Security-Token": "..."}
2	2024-10-08T18:19:53Z	AROAUIAATNQLGCYGOZGAU:user3354961=Hanaa_Mohamed_Fayez	{"X-Amz-Date": "2024-10-08T18:19:53Z", "X-Amz-Security-Token": "..."}

The screenshot shows the AWS Athena Query Editor interface. On the left, the sidebar displays 'Tables (1)' containing 'clouptrail\_logs' and 'Views (0)'. The main area shows a SQL query in progress, with the status bar indicating 'Completed' and execution times. The results table contains two rows of data, each with columns for eventtime, principalid, and requestparam.

#	eventtime	principalid	requestparam
1	2024-10-08T19:00:07Z	AROAUIAATNQLGCYGOZGAU:user3354961=Hanaa_Mohamed_Fayez	{"X-Amz-Date": "2024-10-08T19:00:07Z", "X-Amz-SecurityToken": "AQAB...", "X-Amz-Algorithm": "AWS4-HMAC-SHA256", "X-Amz-Credential": "AWS4-HMAC-SHA256/20241008/us-east-1/lambda/aws4_request", "X-Amz-SignedHeaders": "host", "X-Amz-Expires": "3600", "X-Amz-Signature": "..."}
2	2024-10-08T19:00:08Z	AROAUIAATNQLGCYGOZGAU:user3354961=Hanaa_Mohamed_Fayez	{"X-Amz-Date": "2024-10-08T19:00:08Z", "X-Amz-SecurityToken": "AQAB...", "X-Amz-Algorithm": "AWS4-HMAC-SHA256", "X-Amz-Credential": "AWS4-HMAC-SHA256/20241008/us-east-1/lambda/aws4_request", "X-Amz-SignedHeaders": "host", "X-Amz-Expires": "3600", "X-Amz-Signature": "..."}

## Task 4.2: Use CloudWatch Logs to monitor secure logs

Create a CloudWatch log group named EncryptedInstanceSecureLogs with all default settings.

The screenshot shows the AWS CloudWatch Log Groups page. A success message indicates that the log group 'EncryptedInstanceSecureLogs' has been created. The main table lists three log groups, with 'EncryptedInstanceSecureLogs' being the most recent addition, indicated by a checkmark in the first column.

Log group	Log class	Anomaly detection	Date range	Severity	Retention
/aws/lambda/c131326a3331476l7889375t1-AdjustA...	Standard	Configure	-	-	Never
/aws/lambda/c131326a3331476l7889375t1-AdjustB...	Standard	Configure	-	-	Never
EncryptedInstanceSecureLogs	Standard	Configure	-	-	Never

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "EC2 Instance Connect" and displays a terminal session. The terminal window has a blue header bar with the text "Keyboard shortcut" and "Close permanently". The main area of the terminal shows a log of events from October 8, 2024, at 19:15:35. The log includes messages about the listener starting, the EC2 tagger retrieving tags, and various sudo sessions being opened by the ec2-user root account. The terminal ends with a prompt for a command.

```
2024-10-08T19:15:35Z I! Statd listener listening on: [::]:8125
2024-10-08T19:15:35Z I! {"caller":"ec2tagger/ec2tagger.go:80","msg":"ec2tagger: Initial retrieval of tags succeeded","kind":"processor","name":"ec2tagger","pipeline":"metrics/host"}
2024-10-08T19:15:36Z I! {"caller":"ec2tagger/ec2tagger.go:391","msg":"ec2tagger: EC2 tagger has started, finished initial retrieval of tags and Volumes","kind":"processor","name":"ec2tagger","pipeline":"metrics/host"}
2024-10-08T19:15:36Z I! First time setting retention for log group EncryptedInstanceSecureLogs, update map to avoid setting twice
2024-10-08T19:15:36Z I! [outputs.cloudwatchlogs] Configured middleware on AWS client
2024-10-08T19:15:36Z I! [logagent] piping log from EncryptedInstanceSecureLogs/EncryptedInstanceSecureLogs-i-072defbe799089f37(/var/log/secure) to cloudwatchlogs with retention 180
2024-10-08T19:15:41Z W! [outputs.cloudwatchlogs] Retried 0 time, going to sleep 174.165293ms before retrying.
[ec2-user@ip-10-1-3-14 ~]$ sudo tail -f /var/log/secure
Oct 8 19:15:28 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
Oct 8 19:15:32 ip-10-1-3-14 sudo: pam_unix(sudo:session): session Closed for user root
Oct 8 19:15:51 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/sbin/service#040amazon-cloudwatch-agent#040 status
Oct 8 19:15:51 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
Oct 8 19:15:51 ip-10-1-3-14 sudo: pam_unix(sudo:session): session closed for user root
Oct 8 19:16:12 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/bin/cat#040/opt/aws/amazon-cloudwatch-agent/logs/amazon-cloudwatch-agent.log
Oct 8 19:16:12 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
Oct 8 19:16:12 ip-10-1-3-14 sudo: pam_unix(sudo:session): session closed for user root
Oct 8 19:16:34 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/bin/tail#040-f#040/var/log/secure
Oct 8 19:16:34 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
```

i-072defbe799089f37 (EncryptedInstance)  
PublicIPs: 3.230.118.156 PrivateIPs: 10.1.3.14

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 25°C 10:17 PM 10/8/2024 ENG

The screenshot shows a web browser window with multiple tabs open. The active tab is titled "EC2 Instance Connect" and displays a terminal session. The terminal window has a blue header bar with the text "Keyboard shortcut" and "Close permanently". The main area of the terminal shows a log of events from October 8, 2024, at 19:21:00. The log includes messages about accepting a public key from a specific IP address and port, and then attempting to connect to the instance using Cloud9Instance. The terminal ends with a prompt for a command.

```
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: Accepted publickey for ec2-user from 98.82.104.186 port 43406 ssh2: RSA SHA256:ANhz279oUKdBkvb92r+mrKI1jY6AlubE7x5pjM9hd+A
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: pam_unix(sshd:session): session opened for user ec2-user by (uid=0)

^C
[ec2-user@ip-10-1-3-14 ~]$ sudo tail -f /var/log/secure
Oct 8 19:16:12 ip-10-1-3-14 sudo: pam_unix(sudo:session): session closed for user root
Oct 8 19:16:34 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/bin/tail#040-f#040/var/log/secure
Oct 8 19:16:34 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
Oct 8 19:21:00 ip-10-1-3-14 ec2-instance-connect[4155]: Querying EC2 Instance Connect keys for matching fingerprint: SHA256:ANhz279oUKdBkvb92r+mrKI1jY6AlubE7x5pjM9hd+A
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: error: AuthorizedKeysCommand /opt/aws/bin/eic_runAuthorizedKeys ec2-user SHA256:ANhz279oUKdBkvb92r+mrKI1jY6AlubE7x5pjM9hd+A failed, status 255
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: Accepted publickey for ec2-user from 98.82.104.186 port 43406 ssh2: RSA SHA256:ANhz279oUKdBkvb92r+mrKI1jY6AlubE7x5pjM9hd+A
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: pam_unix(sshd:session): session opened for user ec2-user by (uid=0)
Oct 8 19:23:14 ip-10-1-3-14 sudo: pam_unix(sudo:session): session closed for user root
Oct 8 19:23:18 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/bin/tail#040-f#040/var/log/secure
Oct 8 19:23:18 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
```

i-072defbe799089f37 (EncryptedInstance)  
PublicIPs: 3.230.118.156 PrivateIPs: 10.1.3.14

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 25°C 10:23 PM 10/8/2024 ENG

The screenshot shows a browser window with multiple tabs open, including 'Lab Instructions: Secu...', 'Query editor | Athena', 'Instances | EC2 | us-e...', 'Cloud9Instance - AW...', 'EC2 Instance Connect...', and 'us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?addressFamily=ipv4&connType=standard&instanceId=i-072defbe799089f37&osUser=ec2-user...'. The main content area displays a terminal window with the following log output:

```
r:KiljY6AluB7x5qjM9hd+A failed, status 255
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: Accepted publickey for ec2-user from 98.82.104.186 port 43406 ssh2: RSA SHA256:ANhz279oUKdBkvb92r+mrKI
ljk6AluB7x5qjM9hd+A
Oct 8 19:21:00 ip-10-1-3-14 sshd[3999]: pam_unix(sshd:session): session opened for user ec2-user by (uid=0)
Oct 8 19:23:14 ip-10-1-3-14 sudo: pam_unix(sudo:session): session closed for user root
Oct 8 19:23:18 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/bin/tail#040-f#040/var/log/secure
Oct 8 19:23:18 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
Oct 8 19:23:18 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
Oct 8 19:24:26 ip-10-1-3-14 sshd[4189]: Received disconnect from 98.82.104.186 port 43406:11: disconnected by user
Oct 8 19:24:26 ip-10-1-3-14 sshd[4189]: Disconnected from 98.82.104.186 port 43406:11: disconnected by user
Oct 8 19:24:26 ip-10-1-3-14 sshd[3999]: pam_unix(sshd:session): session closed for user ec2-user
```
... (more log entries) ...
Oct 8 19:27:43 ip-10-1-3-14 sudo: pam_unix(sudo:session): session closed for user root
Oct 8 19:27:46 ip-10-1-3-14 sudo: ec2-user : TTY=pts/0 ; FWD=/home/ec2-user ; USER=root ; COMMAND=/bin/tail#040-f#040/var/log/secure
Oct 8 19:27:46 ip-10-1-3-14 sudo: pam_unix(sudo:session): session opened for user root by ec2-user(uid=0)
```

Below the terminal, the message 'i-072defbe799089f37 (EncryptedInstance)' is displayed, along with Public IPs: 3.230.118.156 and Private IPs: 10.1.3.14.

Create some security logs by successfully connecting and then failing to connect to the *EncryptedInstance* over SSH from your AWS Cloud9 IDE.

The screenshot shows a browser window with multiple tabs open, including 'Lab Instructions: Secu...', 'Instances | EC2 | us-e...', 'Cloud9Instance - AW...', 'AWS Academy Lab Pr...', 'translate - Google Se...', 'AWS Academy Gradu...', and 'us-east-1.console.aws.amazon.com/cloud9/ide/0808172c92b54f40b9e86161773a07f?region=us-east-1'. The main content area displays a terminal window with the following log output:

```
ec2-user@ip-10-1-3-14:~$ ssh -i labuser.pem ec2-user@EncryptedInstance-public-IP
ssh: Could not resolve hostname EncryptedInstance-public-IP: Name or service not known
voclabs:~$ ssh -i labuser.pem ec2-user@EncryptedInstance-public-IP
ssh: Could not resolve hostname EncryptedInstance-public-IP: Name or service not known
voclabs:~$ ssh -i labuser.pem ec2-user@44.202.211.223
The authenticity of host '44.202.211.223 (44.202.211.223)' can't be established.
EDSA key fingerprint is SHA256:PjzdgvWNO4122G+VIXJlnTlxq4i5CsgA7erwZdfBc.
EDSA key fingerprint is MD5:dfc4:1d:4e:a4:99:f9:8a:d9:02:9d:83:fa:f9:4b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '44.202.211.223' (EDSA) to the list of known hosts.
Last login: Tue Oct 8 22:11:41 2024 from ec2-98-82-104-186.compute-1.amazonaws.com
,
  #_
  \###_ Amazon Linux 2
  ~ \####_
  ~ \###_ AL2 End of Life is 2025-06-30.
  ~ \#/ 
  ~ \V- '-->
  ~ / A newer version of Amazon Linux is available!
  ~ / Amazon Linux 2023, GA and supported until 2028-03-15.
  ~ / https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-10-1-3-14 ~]$
```

Below the terminal, the message 'CodeWhisperer AWS: profile.default' is displayed, along with a Windows taskbar showing various icons and system status.

```
voclabs:~/environment $ ssh -i labuser.pem ubuntu@34.200.212.209
The authenticity of host '34.200.212.209 (34.200.212.209)' can't be established.
ECDSA key fingerprint is SHA256:PjzdgvMNO4122G+IXJnTuxp4i5CLCsPa/ernZdfBc.
ECDSA key fingerprint is MD5:df:c4:1d:4e:a4:99:f9:8a:d9:02:d9:83:fa:7a:f9:4b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '34.200.212.209' (ECDSA) to the list of known hosts.
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
voclabs:~/environment $ ssh -i labuser.pem ubuntu@34.200.212.209
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
voclabs:~/environment $ ssh -i labuser.pem ubuntu@34.200.212.209
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
voclabs:~/environment $ ssh -i labuser.pem ubuntu@34.200.212.209
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
voclabs:~/environment $ ssh -i labuser.pem ubuntu@34.200.212.209
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
voclabs:~/environment $ ssh -i labuser.pem ubuntu@34.200.212.209
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
voclabs:~/environment $
```

Verify that the secure logs are being written to the CloudWatch log group

| Timestamp                     | Message                                                                                                           |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------|
| 2024-10-08T22:15:36.058+03:00 | Oct 8 19:13:26 ip-10-1-3-14 ec2-instance-connect[3462]: Querying EC2 Instance Connect keys for matching instance. |
| 2024-10-08T22:15:36.058+03:00 | Oct 8 19:13:26 ip-10-1-3-14 ec2-instance-connect[3658]: Querying EC2 Instance Connect keys for matching instance. |
| 2024-10-08T22:21:00.566+03:00 | Oct 8 19:21:00 ip-10-1-3-14 ec2-instance-connect[4155]: Querying EC2 Instance Connect keys for matching instance. |

## Task 4.3: Create a CloudWatch alarm to send notifications for security incidents

Go to the *EncryptedInstanceSecureLogs* CloudWatch log group

The screenshot shows the AWS CloudWatch Alarms interface. On the left, a sidebar lists various CloudWatch services: Favorites and recent dashboards, Alarms (selected), In alarm, All alarms (highlighted in blue), Billing, Logs (Log groups, Log Anomalies, Live Tail, Logs Insights, Contributor Insights), and Metrics (All metrics, Explorer). The main pane displays a table of alarms. One alarm is listed:

| Name                                                 | State             | Last state update (Local) | Conditions                                       |
|------------------------------------------------------|-------------------|---------------------------|--------------------------------------------------|
| Not valid users exceeding limit on EncryptedInstance | Insufficient data | 2024-10-09 01:03:49       | NotValidUsers >= 5 for 1 datapoints within 1 day |

Create a CloudWatch alarm from the metric filter that you just created

The screenshot shows the AWS CloudWatch Metrics filters interface. On the left, a sidebar lists various CloudWatch services: Favorites and recent dashboards, Alarms, Logs (Log groups selected), Log Anomalies, Live Tail, Logs Insights, Contributor Insights, Metrics, X-Ray traces, Events, Application Signals, and Network monitoring. The main pane shows a metric filter named "Not valid users".

| Filter pattern | Metric                 | Metric value | Default value |
|----------------|------------------------|--------------|---------------|
| "Invalid user" | secure / NotValidUsers | 1            | 0             |

In the CloudWatch console, in the latest log stream for the *EncryptedInstanceSecureLogs* log group, filter the log events for Invalid user

The screenshot shows the AWS CloudWatch Log Events interface. The left sidebar navigation includes 'CloudWatch', 'Favorites and recents', 'Alarms' (with 1 alarm), 'Logs' (selected), 'Log groups' (selected), 'Metrics', and 'Explorer'. The main content area displays a table of log events with columns 'Timestamp' and 'Message'. A search bar at the top contains the query 'invalid user'. The log entries show multiple instances of 'invalid user' being logged by 'sshd' from various IP addresses and times.

In the Alarms area of the CloudWatch console, confirm that the *Invalid users* alarm has a state of *In alarm*

The screenshot shows the AWS CloudWatch Alarms interface. The left sidebar navigation includes 'CloudWatch', 'Favorites and recents', 'Alarms' (selected), 'Logs', and 'Metrics'. The main content area shows a single alarm named 'Not valid users exceeding limit on EncryptedInstance'. The status bar indicates 'Not validUsers >= 5 for 1 datapoints within 1 day'. Below this, a chart titled 'Count' shows the number of invalid users over time, with a red bar indicating the current count of 39. The timeline shows data points from October 3rd to 9th, with the most recent data point at 10/09 showing an 'In alarm' status.

The screenshot shows the AWS CloudWatch Metrics Dashboard. On the left, the navigation pane includes sections for Alarms, Logs, Metrics, and CloudShell. The main area displays 'Alarms by AWS service' and 'Recent alarms'. A red box highlights a recent alarm titled 'Not valid users exceeding limit on EncryptedInstance' in the US East (N. Virginia) region. The alarm details show a threshold of 5.0, with 5 data points exceeding it between 02:00 and 02:00. The status is shown as 'NotValidUsers'.

Go to the inbox for the email address

The screenshot shows the Yahoo Mail inbox. The left sidebar lists categories like Inbox (1.9k), Unread, Starred, Drafts, Sent, Archive, Spam, and Deleted items. The inbox contains an email from 'Amazon Web Services' (amazonaws.com). The subject is 'ALARM: "Not valid users exceeding limit on EncryptedInstance" in US East (N. Virginia)'. The email body provides details about the alarm threshold being crossed and includes a link to view the alarm in the AWS Management Console. The right side of the screen features two promotional banners for 'yahoo/finance'.

## Task 4.4: Configure AWS Config to assess security settings and remediate the configuration of AWS resources

Create a new S3 bucket named compliance-bucket-unique-ID

The screenshot shows the AWS S3 buckets page in the us-east-1 region. A green success message at the top states: "Successfully created bucket 'compliance-bucket-0c2aa8f53f0f94f04'". Below this, a table lists seven buckets. The newly created bucket, "compliance-bucket-0c2aa8f53f0f94f04", is highlighted with a blue border. The table columns include Name, AWS Region, IAM Access Analyzer, and Creation date.

| Name                                            | AWS Region                      | IAM Access Analyzer                         | Creation date                         |
|-------------------------------------------------|---------------------------------|---------------------------------------------|---------------------------------------|
| aws-athena-query-results-292059048982-us-east-1 | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a> | October 8, 2024, 21:41:17 (UTC+03:00) |
| aws-config-0c2aa8f53f0f94f04                    | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a> | October 8, 2024, 20:56:23 (UTC+03:00) |
| cloudtrail-logs-0c2aa8f53f0f94f04               | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a> | October 8, 2024, 20:56:22 (UTC+03:00) |
| <b>compliance-bucket-0c2aa8f53f0f94f04</b>      | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a> | October 9, 2024, 01:22:52 (UTC+03:00) |
| databucket02b4b454419cb2f78                     | US East (N. Virginia) us-east-1 | <a href="#">View analyzer for us-east-1</a> | October 8, 2024, 21:14:31 (UTC+03:00) |
| temp-0c2aa8f53f0f94f04                          | US East (N. Virginia) us-east-1 |                                             | October 8, 2024, 20:56:23             |

Enable object ownership on the s3-objects-access-log bucket

The screenshot shows the AWS S3 bucket permissions page for "s3-objects-access-log-0c2aa8f53f0f94f04". A green success message at the top states: "Successfully edited Object Ownership.". The page displays the "Permissions overview" section, which includes information about access findings and the "Block public access (bucket settings)" section, which is currently disabled.

The screenshot shows the 'Permissions overview' section of the AWS S3 console. It includes an 'Access finding' summary, a 'Block public access (bucket settings)' section where 'Block all public access' is set to 'On', and a 'Bucket policy' section. The browser status bar at the bottom indicates the user's email and session ID.

Navigate to the AWS Config console, and choose Get started

The screenshot shows the AWS Config console dashboard. The left sidebar lists navigation options like Dashboard, Conformance packs, Rules, Resources, Aggregators, Advanced queries, Settings, and What's new. The main area displays 'Compliance status' with 1 Noncompliant rule(s) and 7 Noncompliant resource(s). Below this is a table for 'Noncompliant rules by noncompliant resource count', showing a single entry for 's3-bucket-logging-enabled'. A 'Resource inventory (0)' section is also present. The browser status bar at the bottom indicates the user's email and session ID.

In the AWS Config console, add an AWS managed rule

The screenshot shows the AWS Config Rules page. A green success message at the top states: "The rule: s3-bucket-logging-enabled has been added to your account." Below this, the "Rules" section is displayed with a table. The table has columns: Name, Remediation action, Type, and Enabled evaluation mode. One row is shown: "s3-bucket-logging-enabled", "Not set", "AWS managed", and "DETECTIVE". The left sidebar shows navigation options like Dashboard, Conformance packs, Rules, Resources, Aggregators, and more.

Configure manual remediation settings for the *s3-bucket-logging-enable* rule

The screenshot shows the AWS Config Rule details page for the "s3-bucket-logging-enabled" rule. A green success message at the top says: "Success! s3-bucket-logging-enabled has been updated." The page is titled "s3-bucket-logging-enabled" and contains two main sections: "Rule details" and "Parameters". In the "Rule details" section, there are three columns: "Description" (which includes a link to the config rule ARN), "Enabled evaluation mode" (set to "DETECTIVE"), and "Detective evaluation trigger type" (which lists "Oversized configuration changes" and "Configuration changes"). The "Parameters" section is currently empty. The left sidebar shows the same navigation options as the previous screenshot.

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules/details?configRuleName=s3-bucket-logging-enabled

AWS Config Services Search [Alt+S] GrantedPermission 1/1 Edit Delete

**AWS Config**

Dashboard Conformance packs **Rules** Resources Aggregators Compliance Dashboard Conformance packs Rules Inventory Dashboard Resources Authorizations Advanced queries [Preview](#) Settings What's new Documentation

**Remediation action**

|                    |                              |             |                              |
|--------------------|------------------------------|-------------|------------------------------|
| Remediation action | AWS-ConfigureS3BucketLogging | Description | Enables Logging on S3 Bucket |
|--------------------|------------------------------|-------------|------------------------------|

**Parameters**

| Key                                | Value                                                            | Description        |
|------------------------------------|------------------------------------------------------------------|--------------------|
| AutomationAssumeRole               | arn:aws:iam::292059048982:role/SSMAutomationRole                 | (Optional) The AR  |
| TargetPrefix                       | -                                                                | (Optional) Specifi |
| GranteeEmailAddress                | -                                                                | (Optional) Email a |
| GranteeType                        | CanonicalUser                                                    | (Optional) Type o  |
| BucketName                         | RESOURCE_ID                                                      | (Required) The na  |
| GranteeId                          | 5036ff219d0f224e919b0b6700d1f2860163c4aa83ed6cbdb14bcdb2bb939d1c | (Optional) The ca  |
| GranteeUri                         | -                                                                | (Optional) URI of  |
| TargetObjectKeyPartitionDataSource | -                                                                | (Optional) Specifi |

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 24°C 12:08 AM 10/10/2024

us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/rules/details?configRuleName=s3-bucket-logging-enabled

AWS Config Services Search [Alt+S] Remediate 1/1

**AWS Config**

Dashboard Conformance packs **Rules** Resources Aggregators Compliance Dashboard Conformance packs Rules Inventory Dashboard Resources Authorizations Advanced queries [Preview](#) Settings What's new Documentation

**TargetObjectKeyPrefix** - (Optional) Amazo

**Resources in scope**

| ID                                              | Type      | Status | Annotation |
|-------------------------------------------------|-----------|--------|------------|
| aws-athena-query-results-292059048982-us-eas... | S3 Bucket | -      | -          |
| aws-config-0c2aa8f53f0f94f04                    | S3 Bucket | -      | -          |
| cloudtrail-logs-0c2aa8f53f0f94f04               | S3 Bucket | -      | -          |
| compliance-bucket-0c2aa8f53f0f94f04             | S3 Bucket | -      | -          |
| databucket02b4b454419cb2f78                     | S3 Bucket | -      | -          |
| s3-inventory-0c2aa8f53f0f94f04                  | S3 Bucket | -      | -          |
| s3-objects-access-log-0c2aa8f53f0f94f04         | S3 Bucket | -      | -          |

https://us-east-1.console.aws.amazon.com/config/home?region=us-east-1#/resources/details?resourceId=databucket02b4b454419cb2f78&resourceType=AWS%3A%3AS3%3A%3ABucket © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 24°C 12:21 AM 10/10/2024

Invoke the AWS Config remediation action so that object logging is enabled on the *compliance-bucket*

The screenshot shows the AWS Config console with the URL [us-east-1.console.aws.amazon.com/config/home?region=us-east-1#rules/details?configRuleName=s3-bucket-logging-enabled](https://us-east-1.console.aws.amazon.com/config/home?region=us-east-1#rules/details?configRuleName=s3-bucket-logging-enabled). The left sidebar is collapsed. The main area displays a table titled "Resources in scope" with the following data:

| ID                                              | Type      | Status                  | Annotation |
|-------------------------------------------------|-----------|-------------------------|------------|
| aws-athena-query-results-292059048982-us-eas... | S3 Bucket | -                       | -          |
| aws-config-0c2aa8f53f0f94f04                    | S3 Bucket | -                       | -          |
| cloudtrail-logs-0c2aa8f53f0f94f04               | S3 Bucket | -                       | -          |
| <b>compliance-bucket-0c2aa8f53f0f94f04</b>      | S3 Bucket | Action execution failed | -          |
| databucket02b4b454419cb2f78                     | S3 Bucket | -                       | -          |
| s3-inventory-0c2aa8f53f0f94f04                  | S3 Bucket | -                       | -          |
| s3-objects-access-log-0c2aa8f53f0f94f04         | S3 Bucket | -                       | -          |

A modal window titled "Remediate" is open over the table, containing the text "(Optional) Amazon Lambda function to run when a resource becomes non-compliant". A "Remediate" button is visible at the bottom right of the modal.

## Estimate cost of CloudTrail, CloudWatch, and AWS Config

The screenshot shows the AWS Pricing Calculator with the URL [calculator.aws/#/estimate](https://calculator.aws/#/estimate). The top section displays the total estimated cost for 12 months: **59.16 USD**, including upfront cost. Below this, the "My Estimate" section shows the breakdown of services and their costs:

| Service Name      | Upfront cost | Monthly cost | Description         | Region                | Config Summary      |
|-------------------|--------------|--------------|---------------------|-----------------------|---------------------|
| AWS CloudTrail    | 0.00 USD     | 3.78 USD     | Management eve...   | US East (N. Virginia) | Management eve...   |
| Amazon CloudWa... | 0.00 USD     | 1.03 USD     | Number of Metric... | US East (N. Virginia) | Number of Metric... |
| AWS Config        | 0.00 USD     | 0.12 USD     | Number of Conti...  | US East (N. Virginia) | Number of Conti...  |

The bottom of the page includes standard links like Privacy, Site terms, and Cookie preferences, along with a copyright notice: © 2024, Amazon Web Services, Inc. or its affiliates. All rights reserved.

The screenshot shows a browser window with multiple tabs open, including 'Knowledge Check: Sec...', 'Instances | EC2 | us...', 'Cloud9Instance - AW...', 'AWS Academy Lab P...', 'translate - Google Se...', and 'AWS Academy Gradu...'. The main content is from the 'awsacademy.instructure.com' website, showing a knowledge check titled 'Knowledge Check: Securing and Monitoring Resources with AWS'. The sidebar on the left lists navigation options: Home, Modules, Discussions, Grades, Courses, Calendar, Inbox, History, and Help. The main area displays the results of a knowledge check:

**KEYBOARD NAVIGATION**

**Knowledge check results**

Your score: 100% (50 points)  
Required score: 70% (35 points)

**Result:** Congratulations! You have completed this knowledge check.

To continue, choose **Next** in the lower-right corner.

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

This badge was issued to [hanaa mohammed fayez](#) on October 07, 2024

[Verify](#) [Celebrate](#)

**AWS Academy Graduate - AWS Academy Cloud Security Builder**

Issued by [Amazon Web Services Training and Certification](#)

Earners of this badge have completed AWS Academy Lab Project - Cloud Security Builder.

[Learn more](#)

 A blue hexagonal badge with a white border. Inside, the AWS logo is at the top, followed by the word "ACADEMY" in a bold sans-serif font, and "Cloud Security Builder" below it.

**Skills**

Amazon KMS AWS Cloudwatch AWS KMS Data Security Elastic Block Storage

Type here to search

25°C 10/10/2024 9:43 PM