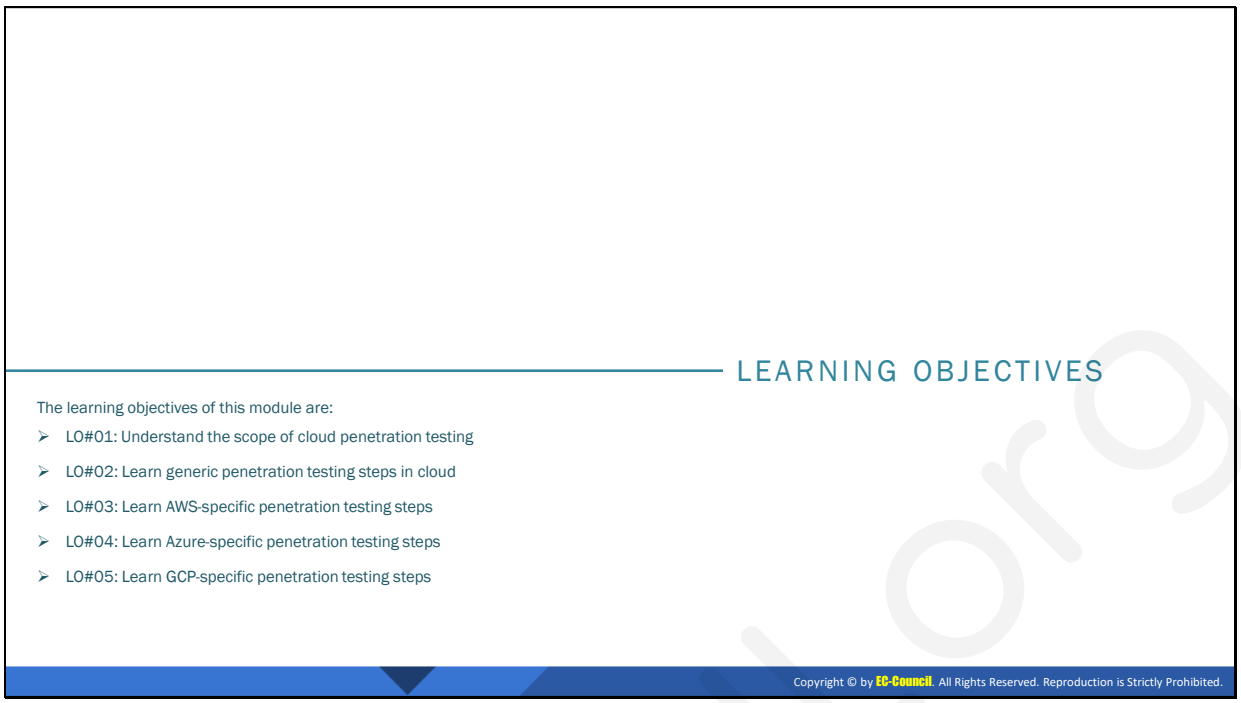
EC-Council

CCSE™
Certified Cloud Security Engineer

Penetration Testing in Cloud

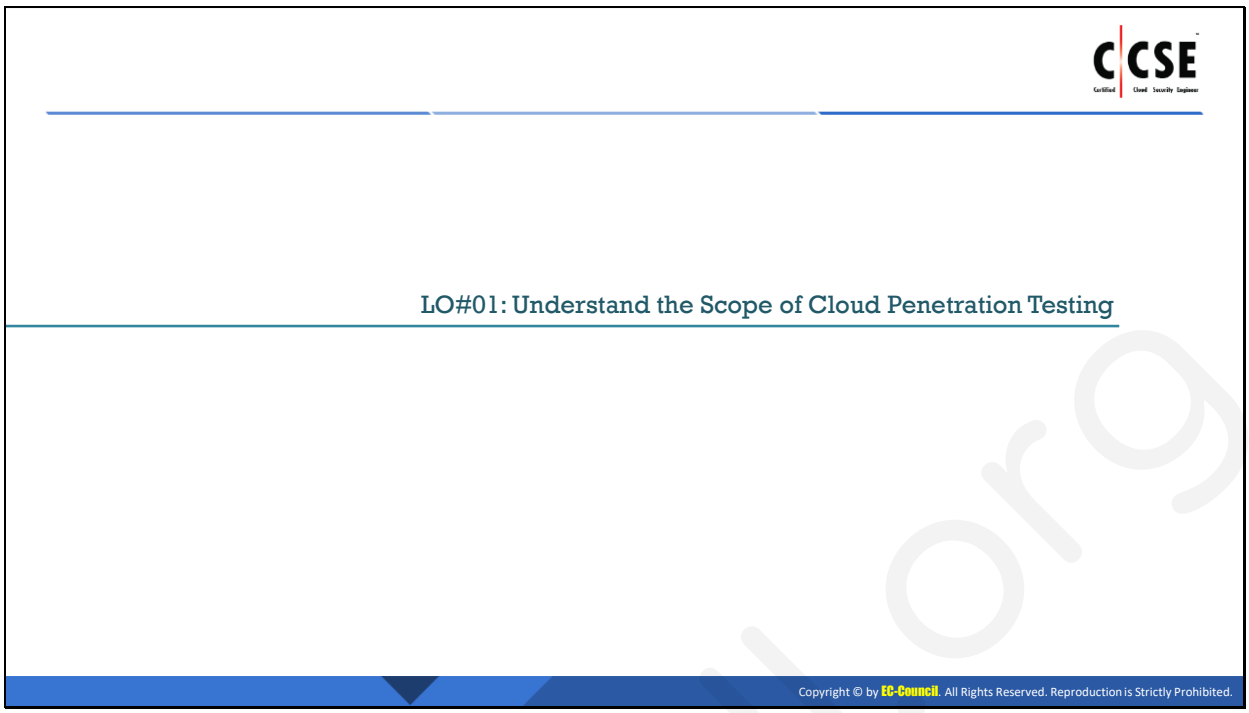
Module 06

This page is intentionally left blank.



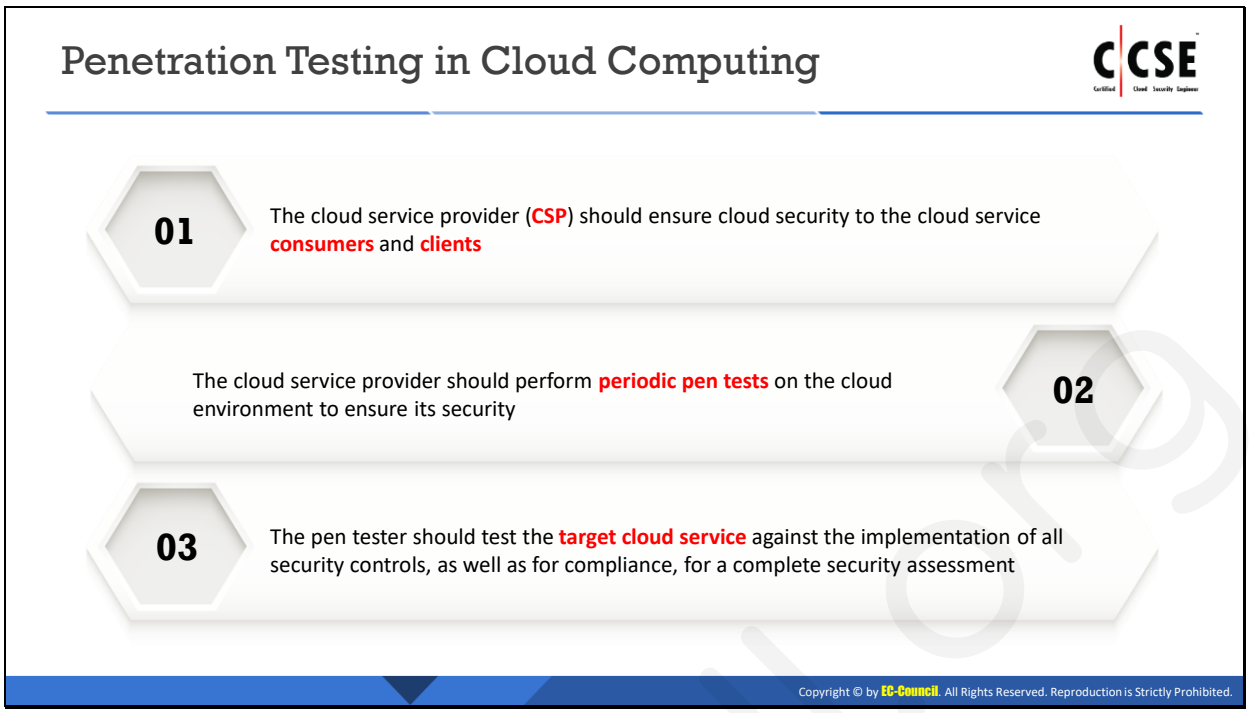
Learning Objectives

This module provides an insight about various concerns and security issues associated with cloud computing. It helps you in learning how to implement a comprehensive penetration testing methodology for assessing security of organization's cloud infrastructure. It also helps you to understand the importance of securing the company's data stored on cloud and learn the scope of cloud pen testing; and its methodology provides knowledge about the compliance and governance issues that companies face in implementing a cloud infrastructure and helps to detect them. It also explains the processes of verifying user authentication, data retention, and performing security analysis of cloud.



LO#01: Understand the Scope of Cloud Penetration Testing

The objective of this section is to understand the scope of cloud penetration testing. The cloud service provider (CSP) should ensure cloud security to the cloud service consumers and clients. Therefore, organizations should know the scope of cloud penetration testing to conduct regular pen tests of cloud.



Penetration Testing in Cloud Computing

Advancement in technology has improved benefits of cloud, such as scalability, ease of use, etc., which has led to multifold adoption of cloud across various companies. This has also resulted in increased risk factors and has put cloud on the radar of hackers as a premium target. Therefore, the hackers are trying to find vulnerabilities and security flaws in systems, app, networks, and servers that may provide access to cloud and its data.

The cloud service provider (CSP) should ensure cloud security to the cloud service consumers and clients. Therefore, we recommend both companies to conduct regular pen tests of cloud.

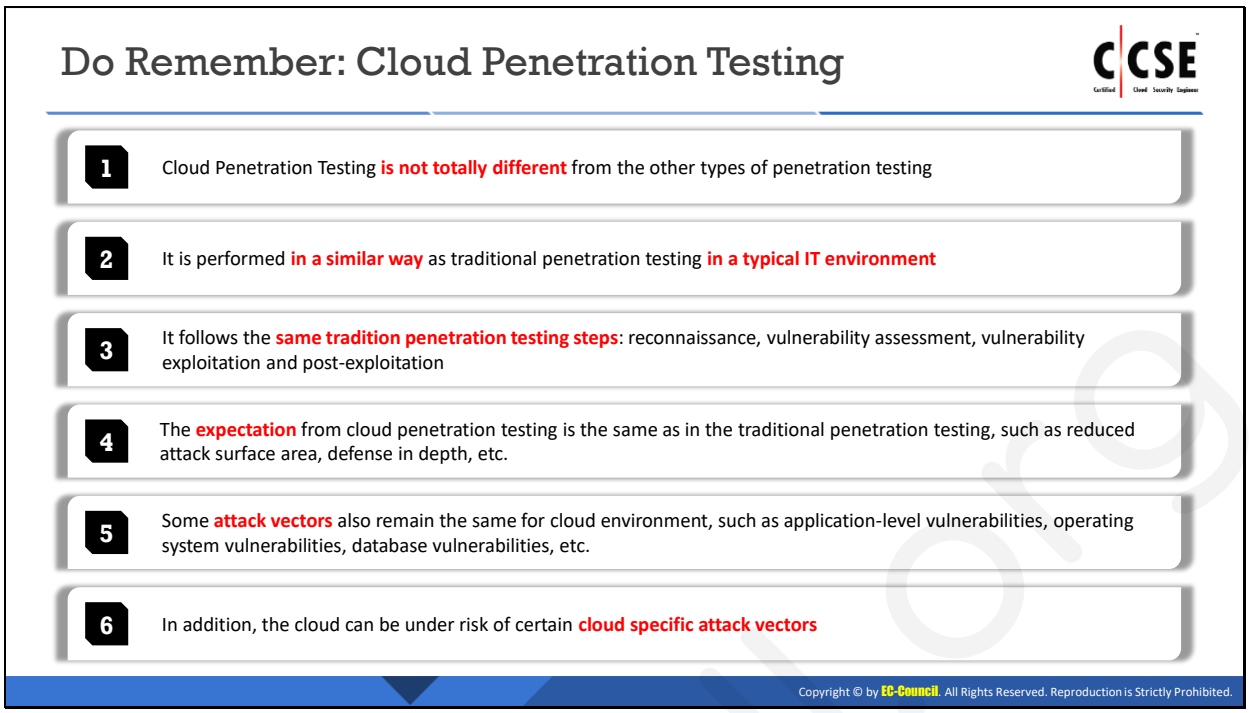
Cloud pen testing is the process of evaluating security posture of cloud. It involves detection of potential vulnerabilities resulting from hardware or software flaws, shared resources, system misconfigurations, operational weaknesses, and other sources.

Pen testing a cloud ensures confidentiality, integrity, and security of the data it hosts. It helps the companies to ensure that all their information assets are auditable, comply with industry regulations, and do not jeopardize their data and app.

The process includes in-depth evaluation of all the components such as apps, networks, servers, and databases that form cloud. The tests will analyze complete security of cloud and its resources to ensure security of hosted data, apps, and services.

Pen testing will help the companies in complying with the local and international standards to avoid legal issues, detecting malicious insiders, finding the weak security policies and configurations, and determine the weak network spots.

Black box pen testing (that is, testing cloud infrastructure without prior knowledge of cloud) is the most effective method of assessing cloud security. Cloud pen testing may be either manual, using industry standard techniques, or automated, which includes use of software apps, such as Core CloudInspect, CloudPassage Halo, Alert Logic, and SecludIT.

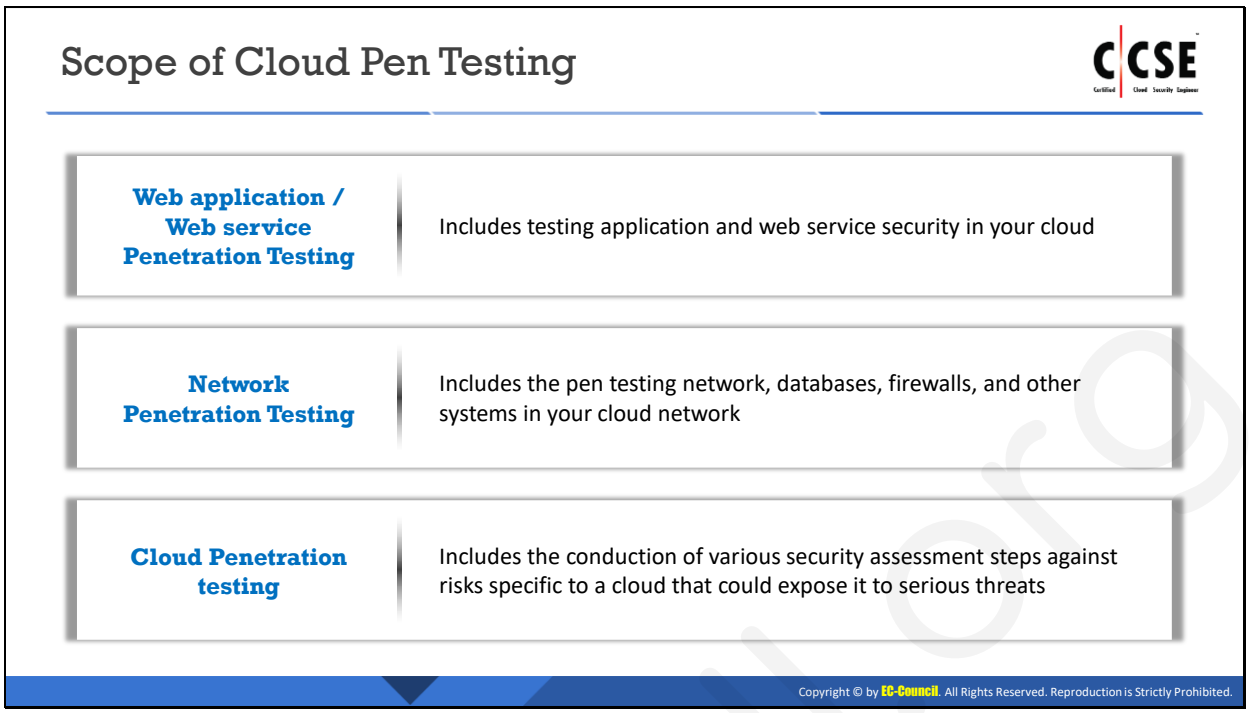


Do Remember: Cloud Penetration Testing

Cloud Penetration Testing is not totally different from the other types of penetration testing. It is performed in a similar way as traditional penetration testing in a typical IT environment. It follows the same tradition penetration testing steps: reconnaissance, vulnerability assessment, vulnerability exploitation and post-exploitation. The expectation from cloud penetration testing is the same as in the traditional penetration testing, such as reduced attack surface area, defense in depth, etc. Some attack vectors also remain the same for cloud environment, such as application-level vulnerabilities, operating system vulnerabilities, database vulnerabilities, etc. In addition, the cloud can be under risk of certain cloud specific attack vectors.

We recommend the testers to check the following points before performing cloud pen testing:

- Go through the Service Level Agreement (SLA) to check if CSP and client have developed and implemented proper security policies
- Ensure appropriate division of responsibilities between CSP and subscriber
- Check the SLA document and track the record of CSP as well as identify role and responsibility to maintain cloud resources
- Verify the usage policy of computer and internet to ensure that the CSP has implemented it as per the proper policy
- Observe cloud networks for unused ports and protocols and ensure that the CSP blocks these services
- Check if the CSP encrypts the data before storing it in cloud servers by default
- Find if cloud uses two factor authentication service and validate the OTP to ensure the network security
- Check the SSL certificates for cloud services in the URL and make sure certificates are purchased from repudiated Certificate Authority



Scope of Cloud Pen Testing

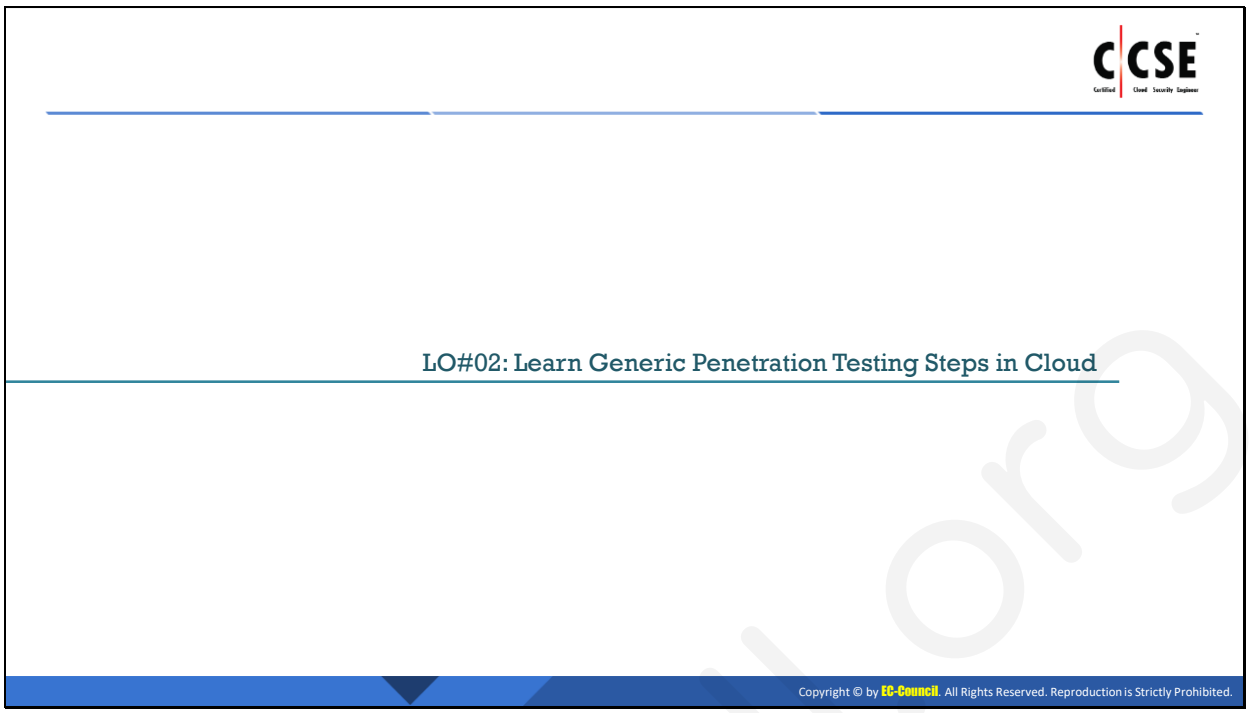
Determining the scope of cloud pen testing is important because cloud security is a responsibility shared between the CSP and the client and offers a multi-tenant environment. Cloud also includes dynamic resources, such as dynamic IP addresses, scalable storage, architecture, networks, client apps, etc.; the testers need to calculate the scope very cautiously to prevent accidental testing of resources not included in the scope.

Therefore, we recommend the testers to determine the scope based on the contracting party, cloud deployment model, offered service model, technologies deployed, and the SLA. The testers may ask the contracting party to provide the detailed of the test or evaluate the test based on the type of cloud service provided. They are advised to also read the SLAs signed between the CSP and clients using clouds to understand the limitations, service, and deployment model, as well as type of access provided.

The scope of cloud pen testing consists of three segments including:

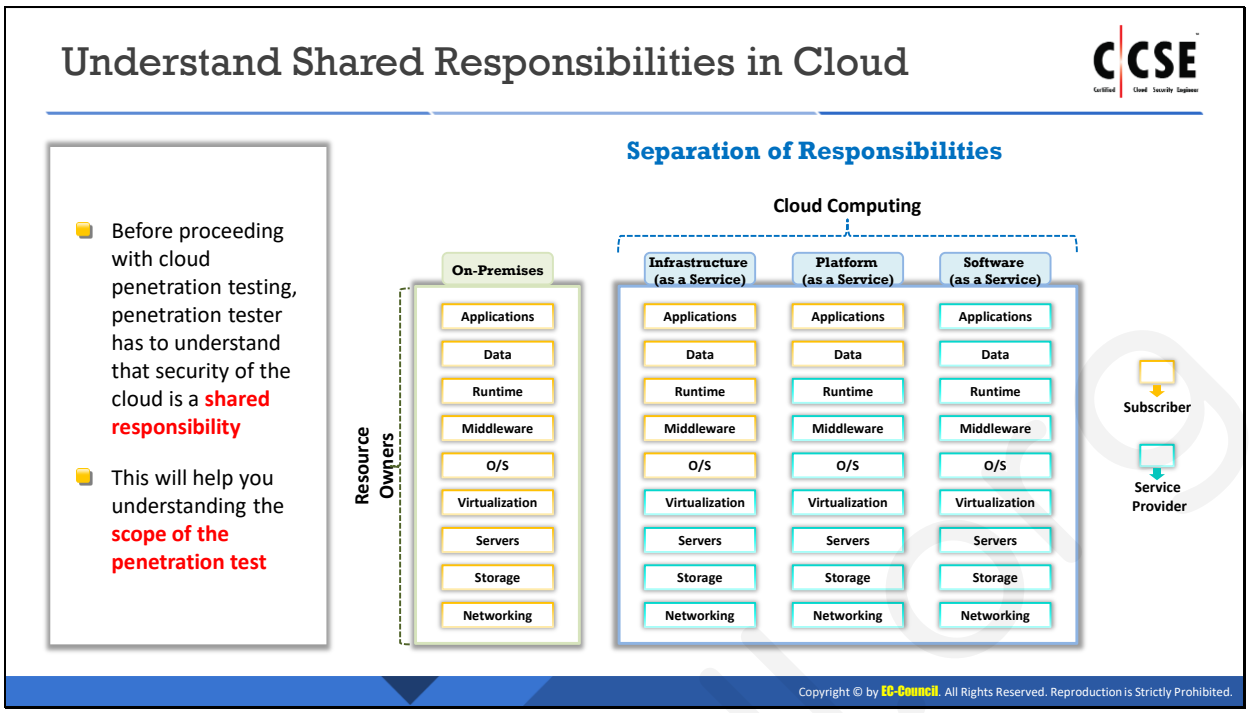
- **App/Web Service Penetration Testing:** App refers to the software program that allows clients and users to sign in onto cloud to access, store, and exchange data. The type of service determines the developer of the app and its deployment side. We recommend the testers to test the app and its services for any vulnerabilities and weaknesses.
- **Network Penetration Testing:** On the CSP side, cloud network consisting of the network connecting cloud storage media with the servers includes databases, firewalls, and other systems whereas the client network may include systems used to access cloud app and routers connecting the systems with the internet.

- **Cloud Penetration Testing:** Cloud pen test refers to evaluation of security across virtual machines, installed apps, and operating systems in a cloud. It includes conducting various security assessment steps against risks specific to a cloud that could expose it to serious threats.



LO#02: Learn Generic Penetration Testing Steps in Cloud

The objective of this section is to introduce Penetration Testing Process, Policies, and Limitations.



Understand Shared Responsibilities in Cloud

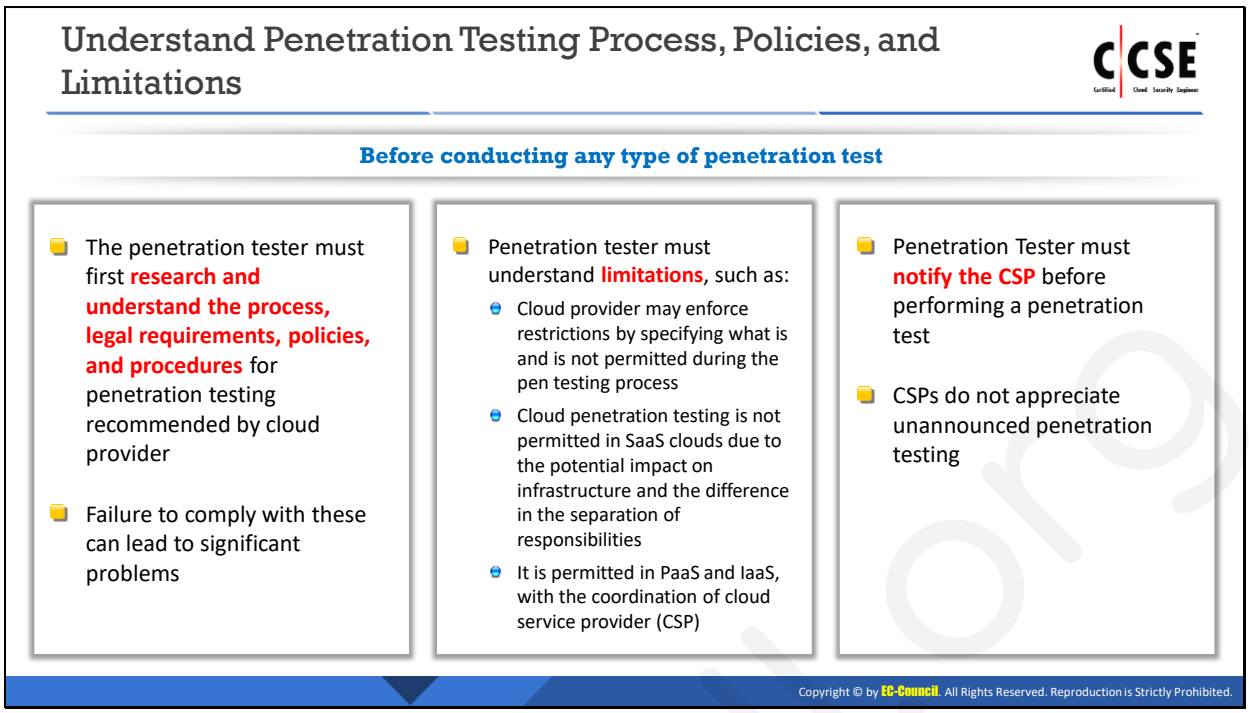
In a cloud environment, the shared responsibility presents various limitations based on the type of service cloud is offering. In such cases, we recommend the companies conducting the test to limit it to the resources they control and also ensure that the tests do not disrupt any services or result in failure of network or app leading to losses for other parties.

Cloud Penetration testing is not allowed in SaaS cloud due to the impact of the infrastructure and the difference in the level of responsibilities. It is allowed in PaaS and IaaS with the coordination of CSP.

The deployment model of cloud, such as public, private, and hybrid, may impact the limitations of pen testing. The public model has more limitations as various companies may be part of it. The private cloud model has only one company using cloud to store and offer its services. The hybrid model is the most complex and has more limitations as it comprises companies using cloud as private and public service.

Another limitation of cloud pen testing that we recommend is that the tester to not perform DDoS cyberattacks on the service, as it may impact the entire network and other crucial resources and result in unavailability of complete cloud.

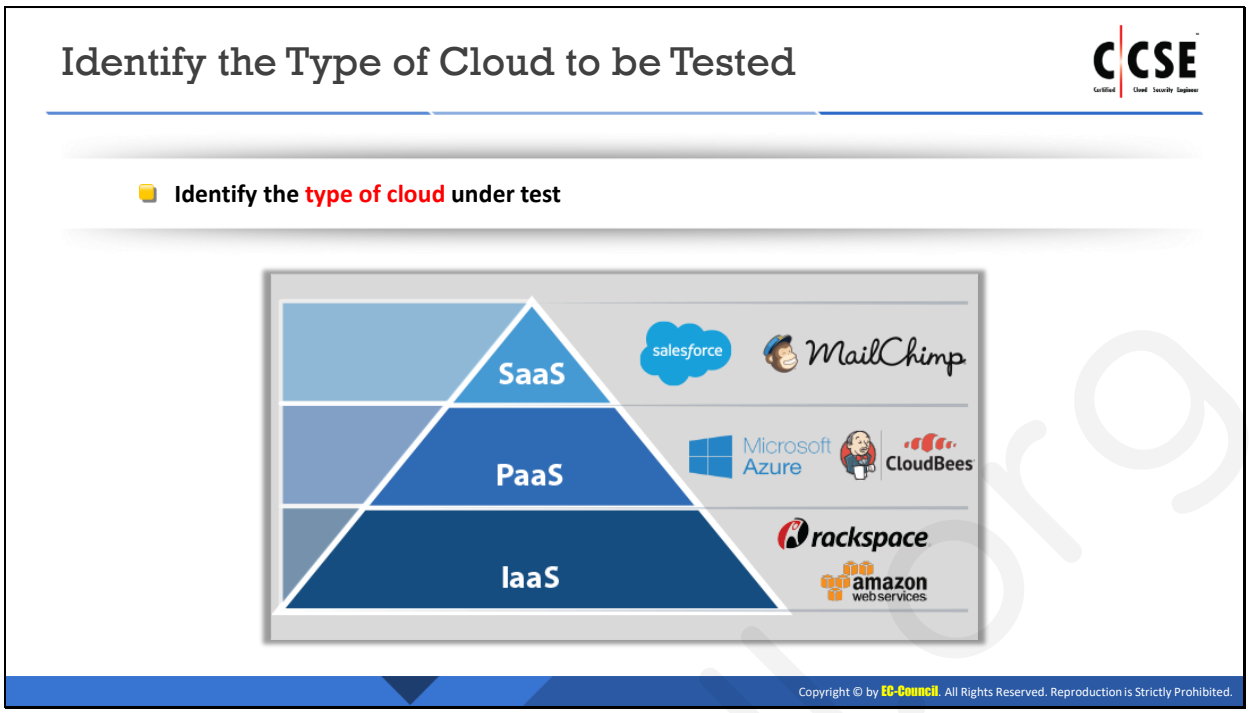
In a cloud, we recommend the testers to not use the pivot cyberattack, which uses a compromised system in a network as a base to cyberattack other systems. These cyberattacks may result in compromise of the resources belonging to other tenants.



Understand Penetration Testing Process, Policies, and Limitations

Before conducting any type of penetration test:

- The penetration tester must first research and understand the process, legal requirements, policies, and procedures for penetration testing recommended by cloud provider.
- Failure to comply with these can lead to significant problems.
- Penetration tester must understand limitations, such as:
  - Cloud provider may enforce restrictions by specifying what is and is not permitted during the pen testing process.
  - Cloud penetration testing is not permitted in SaaS clouds due to the potential impact on infrastructure and the difference in the separation of responsibilities.
  - It is permitted in PaaS and IaaS, with the coordination of cloud service provider (CSP).
- Penetration Tester must notify the CSP before performing a penetration test.
- CSPs do not appreciate unannounced penetration testing.



Identify the Type of Cloud to be Tested

We recommend the testers Identify the type of cloud under test (i.e.) identify if the company they are testing is a cloud provider or tenant. It is also possible that the company acts as a cloud provider as well as tenant. This will help in determining type of cloud services the testers target and define the scope of cloud pen test. Different types of clouds based on the service provided are as follows:

- **Infrastructure as a Service (IaaS):** In IaaS, cloud provider supplies hardware and network connectivity to the tenant and the tenant is responsible for the Virtual machine and everything that runs within it. For example, rackspace, amazon web services, etc.
- **Platform as a Service (PaaS):** In PaaS, cloud provider supplies all the components required to run the app and the tenant supplies the app they wish to deploy. For example, Microsoft Azure, CloudBees, etc.
- **Software as a Service (SaaS)**: In SaaS, cloud provider supplies the app and all the components required to run it. For example, salesforce, mailchimp, etc.

The cloud service model directly affects the scope of testing, as it determines the resources controlled by the target company and if the test is possible. Therefore, we recommend the testers to identify the type of cloud services offered prior to initiating the tests.

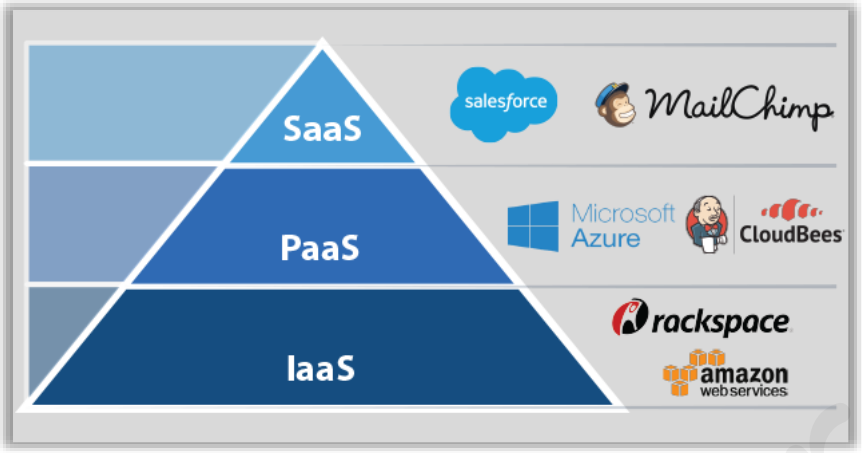
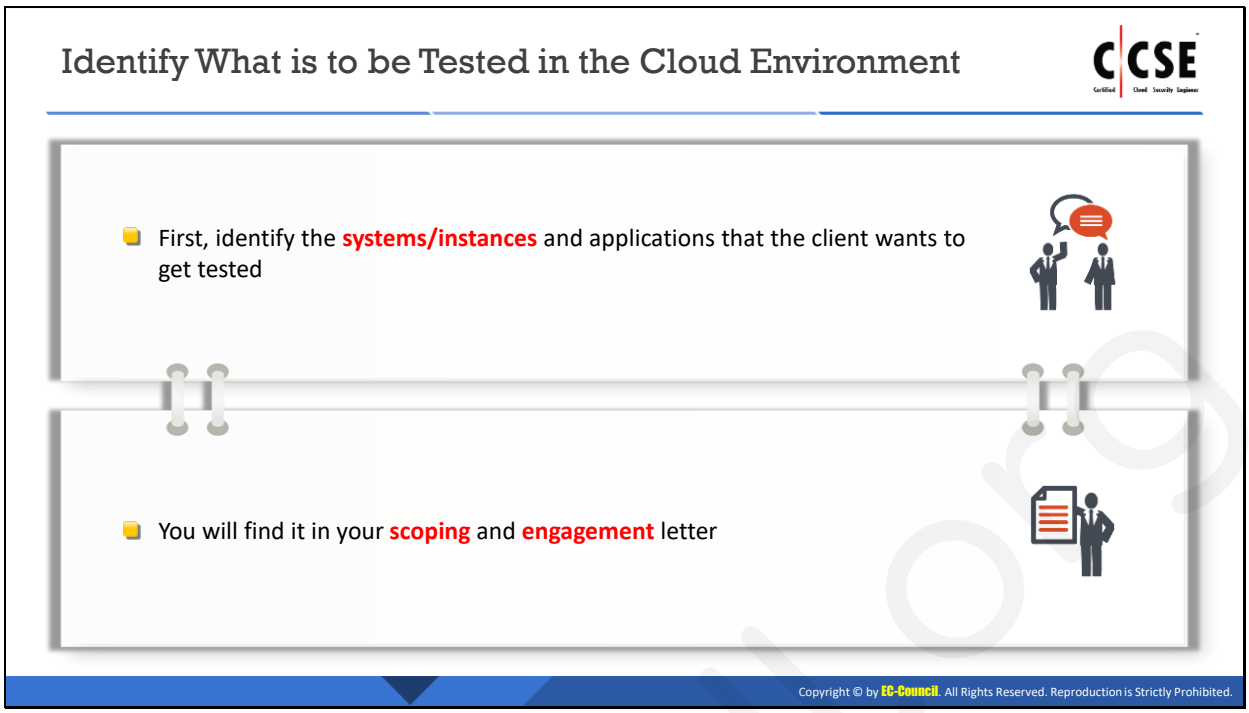
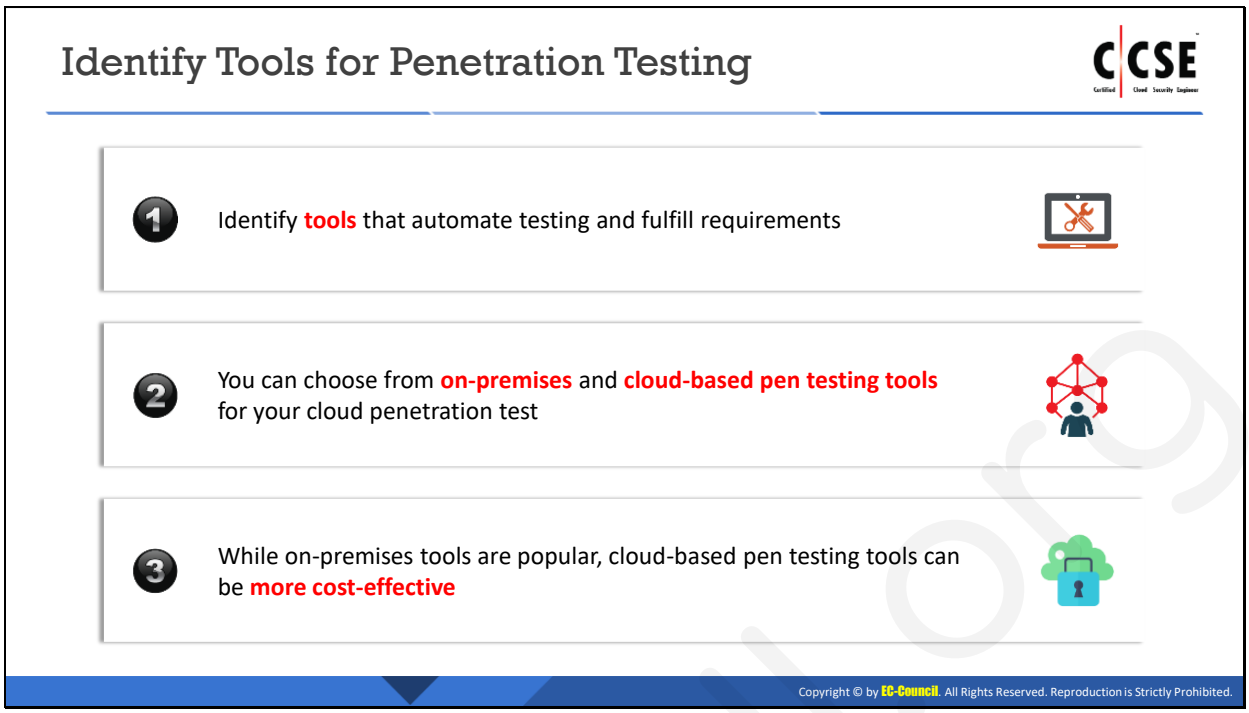


Figure 6.1: Cloud-Based Services



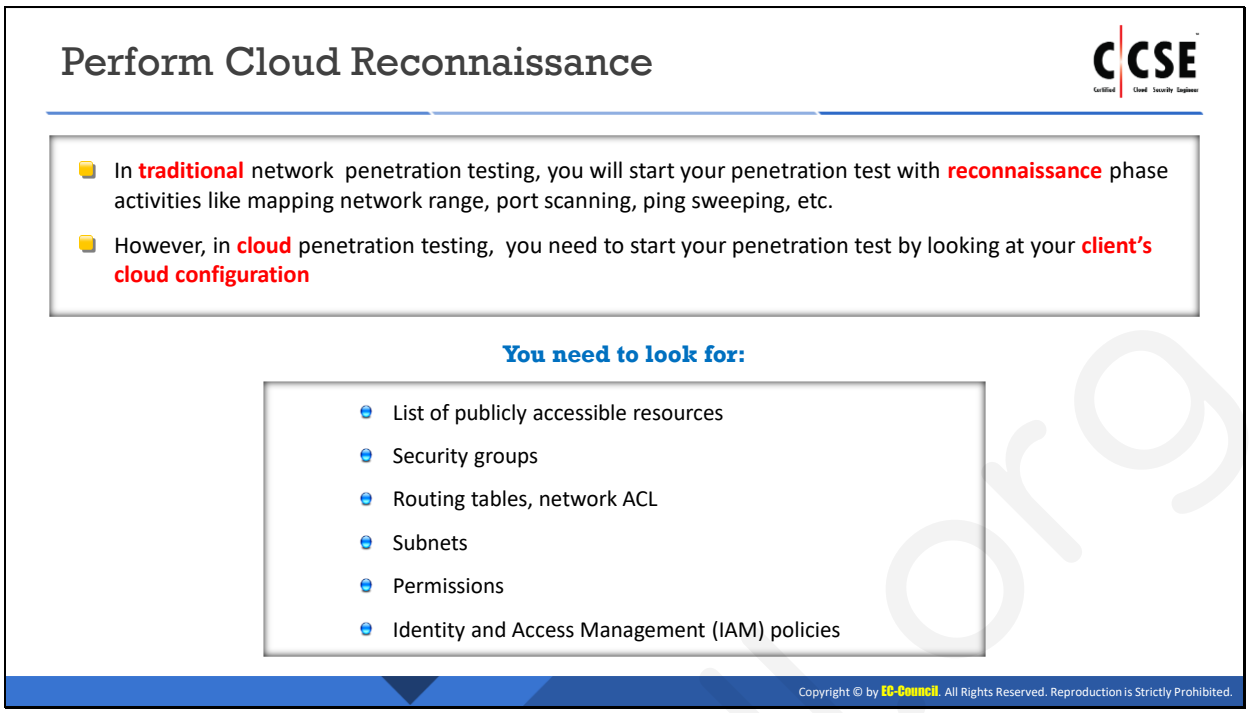
Identify What is to be Tested in the Cloud Environment

- First, identify the systems/instances and applications that the client wants to get tested
- You will find it in your scoping and engagement letter



Identify Tools for Penetration Testing

We recommend you to find the tools that may help you in performing different tests on cloud environment based on cloud type and tests the company wants to conduct and to find the tools that may completely automate the process and meet the pen goals of the company.



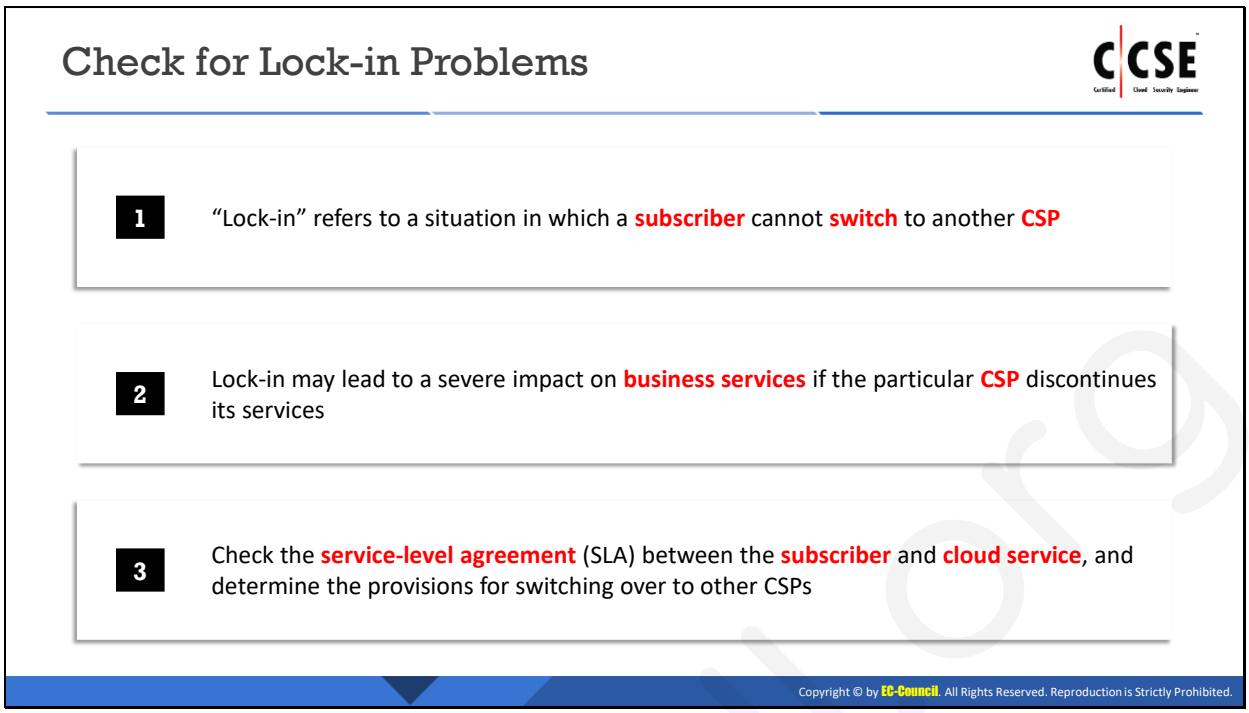
Perform Cloud Reconnaissance

In traditional network penetration testing, you will start your penetration test with reconnaissance phase activities like mapping network range, port scanning, ping sweeping, etc. However, in cloud penetration testing, you need to start your penetration test by looking at your client's cloud configuration. In performing pen testing as well as vulnerability scanning, information about the target will play an important role. Information refers to the details such as hardware, software, networks, databases, operating systems, and their versions to find the known vulnerabilities present on cloud.

Reconnaissance refers to the process of searching and gathering the information about a target to detect its weaknesses and flaws. The testers perform cloud reconnaissance to identify the security flaws and vulnerabilities and exploit them to penetrate cloud security or simulate cyberattacks.

As a pen tester, we recommend that you look for the following components and resources of cloud to perform successful test:

- List of publicly accessible resources
- Security groups
- Routing tables, network ACL
- Subnets
- Permissions
- Identity and Access Management (IAM) policies

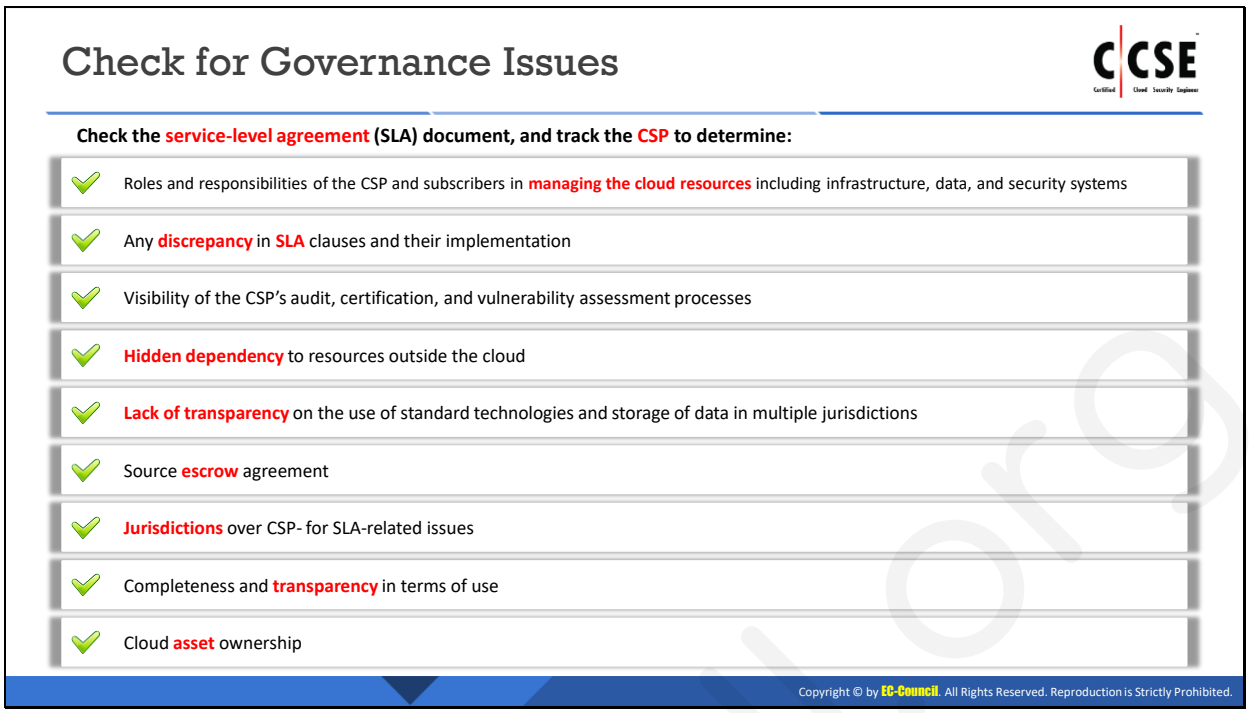


Check for Lock-in Problems

Vendor lock-in problem is the major drawback of cloud computing. Lock-in" refers to a situation in which a subscriber cannot switch to another CSP. It is the act of making a customer dependent on a vendor for products and services and unable to use another vendor without substantial switching costs.

Lock-in may lead to a severe impact on business services if the particular CSP discontinues its services. The lock-in problem occurs when a company wants to change its cloud providers and not able to move apps or data across different cloud services, due to difference in the operating systems and configurations of resources and services of cloud providers. The incompatible resources, settings, and configurations make it difficult for the clients to interoperate, shift, or manage data and services, and collaborate with other customers or vendors. Therefore, we recommend the client to have flexibility to change cloud providers as per the business requirement.

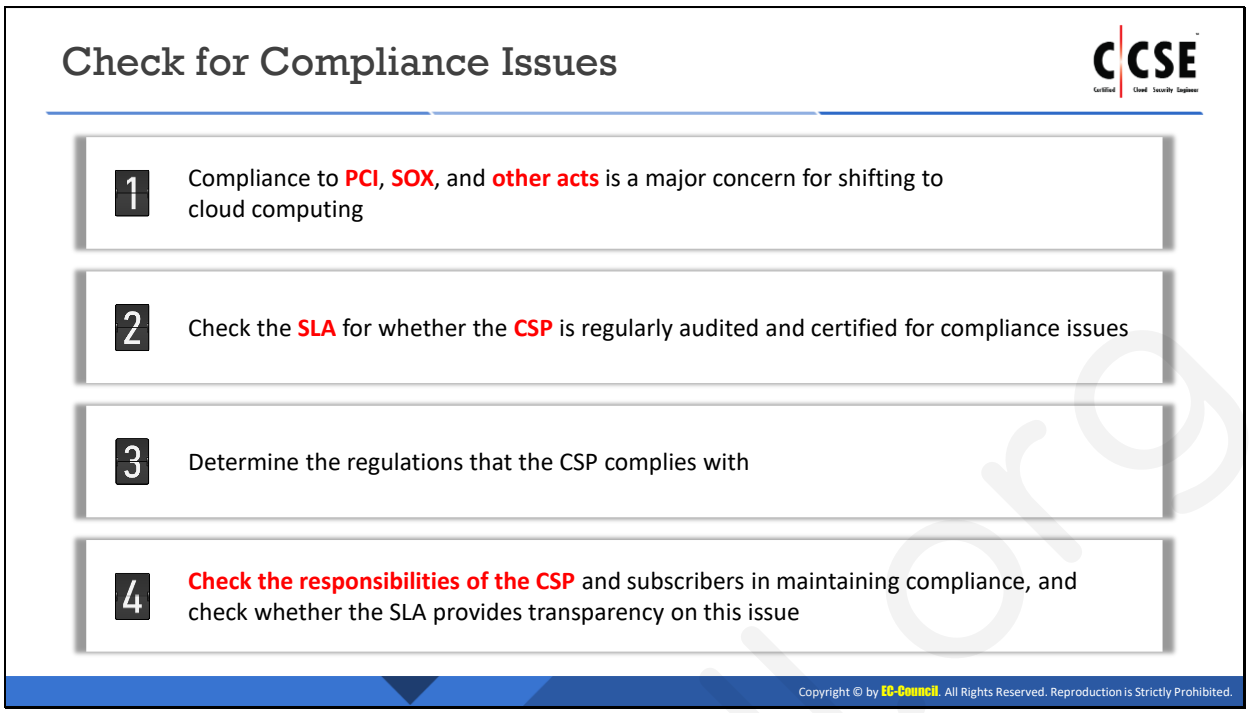
The testers may access and read the SLAs and other documents related to cloud service contract to understand the lock-in terms and conditions. Determine the terms and conditions the subscriber needs to follow to migrate to other clouds. Ensure that the client does not face any lock-in issues with cloud service.



Check for Governance Issues

As testers, we recommend you to check the SLA document and track record of CSP to find if it had any governance issues in the past and to check for the following issues and processes to find if the client may have governance issues with the CSP:

- Roles and responsibilities of the CSP and subscribers in managing cloud resources including infrastructure, data, and security systems
- Any discrepancy in SLA clauses and their implementation
- Visibility of the CSP's audit, certification, and vulnerability assessment processes
- Hidden dependency to resources outside cloud
- Lack of transparency on the use of standard technologies and storage of data in multiple jurisdictions
- Source escrow agreement
- Jurisdictions over CSP for SLA-related issues
- Completeness and transparency in terms of use
- Cloud asset ownership



Check for Compliance Issues

We recommend the testers check if cloud services abide to the regulations and standards of all the regions, where the client is operating or offering services. Check if the service is compliant with international standards for data protection belonging to the different industries served by the client. Some of the standards include PCI for banking, SOX for corporate disclosure, etc.

We recommend you read the SLA documents carefully to find if the CSP performs regular audits and tests of the resources used to offer cloud services. Determine the regulations that CSP complies and check if they meet the requirements of the client. Check the responsibilities of the CSP and subscribers in maintaining compliance, and check whether the SLA provides transparency on this issue.

Some of the compliance issues presented by using cloud services to store, access, and manage information belonging to various companies include:

- **Privacy Compliance:** We recommend that you check if the cloud provider is able to provide proper security to the sensitive client data stored on cloud. Cross examine the privacy of data offered with the local laws and SLAs signed between the subscriber and CSP.
- **Geographical Compliance:** Cloud computing is a globally accessible service and a CSP residing in one or more nations may offer services to companies residing in other nations. All the countries have different approach to data security and compliance. For example, a country might have provisions to access and verify data stored on databases stored in it in case of civil or legal issues while the client's country may have no such laws. Therefore, we recommend the testers verify if the cloud provider complies with

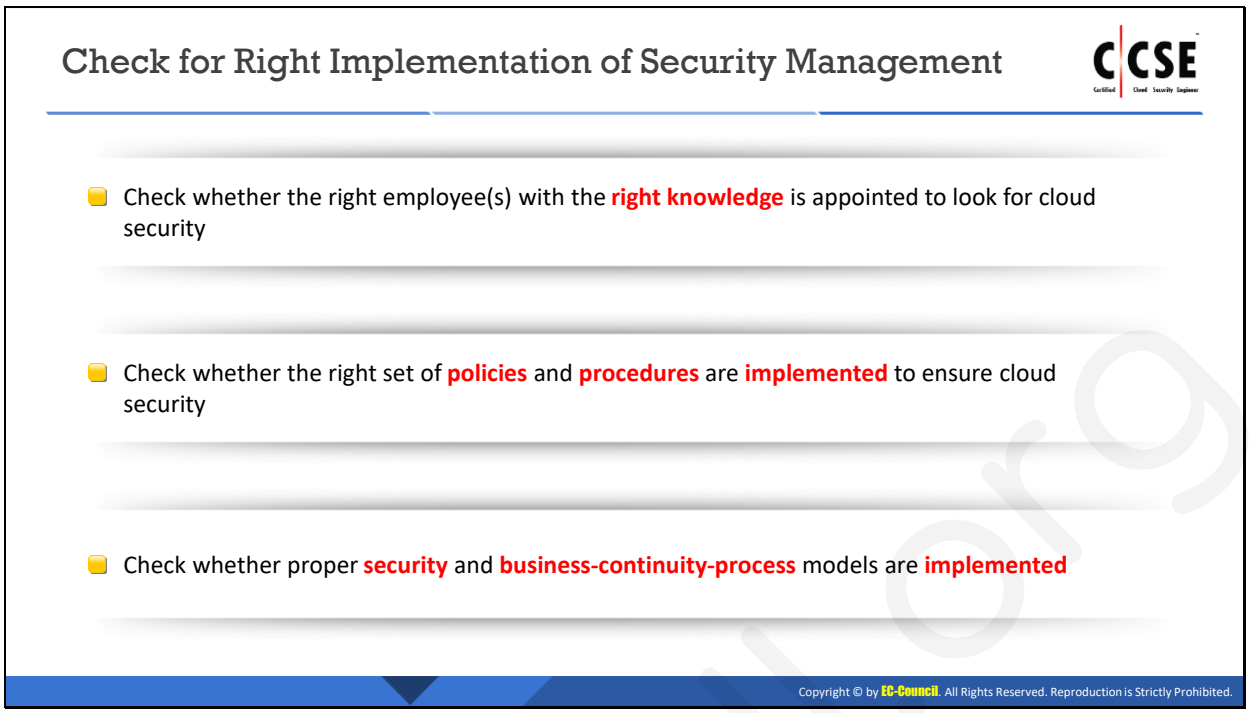
data security norms of all the nations they provide services from and to, as well as check if it fits the client's data security policies.

- **Industry Compliance:** Companies selecting cloud services may be from different industries and may have different data and apps to store or run on cloud environment. We recommend these companies to strictly operate under the defined industrial standards, regulations, and security laws. Therefore, we recommend the testers to understand the industrial regulations the client company to comply with and check if cloud services comply with them.

Factors of Compliance

The testers may evaluate the following factors to see if cloud services comply with all the regulations:

- **Accessibility:** Check for all the users and authorities having access to the data stored in client using logs to ensure that only authorized personnel may access it. Verify the processes that determine users and their access rights. Ensure that the cloud provided privileged access to limited and authorized personnel only, and only the client has complete authority to escalate privileges of any user.
- **Location:** Request the CSP to provide data about the location of all databases storing the client data and servers providing the access.
- **Platform Integrity and Security:** Determine the technology used to store, process, and transfer data across cloud. Ensure that the CSP uses the latest technology with all the updates installed to provide data integrity and security.
- **Alerting Systems:** Find if the CSP has installed any alerting systems on cloud that reports all the security issues and if there is any process of reporting a security incident to the client. The alerts are also required to include details about the severity of incident, impact on the client and actions required.
- **Auditing:** We recommend the testers to find the process of auditing and reporting followed by the CSP and to ensure that it meets the industrial standards and regulations. Check the regulations followed by the CSP and its compliance with various sectors of the industries.



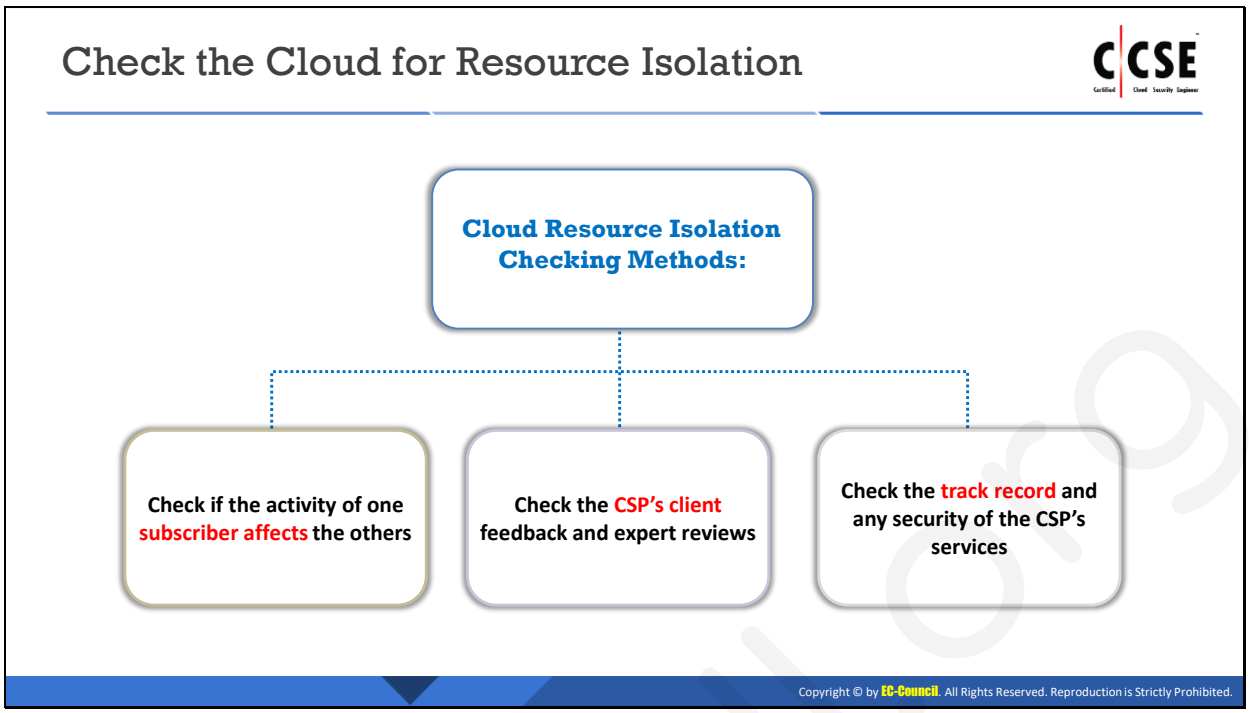
Check for Right Implementation of Security Management

The proper security management process assists the company in finding and fixing the various security vulnerabilities and risks. It includes regular audits, scans, and checks of all the hardware devices and software programs to find missing updates, vulnerabilities, and configuration flaws. This will help the company to mitigate the risks and reduce the impact of cyberattacks that exploit security shortcomings.

Check if the company has hired employees having fair knowledge of working with cloud based technologies. Verify the different security, authentication, and authorization processes these employees follow. Check if the employees working on ensuring security of cloud have right knowledge and if they use all the necessary tools.

We recommend the testers verify if the client company has implemented necessary policies and procedures regarding the use of cloud services. We recommend the company have proper policies regarding system settings, network, access, authorization, remote access, and proper physical access procedures across all it branches. It is advised to implement strict employee, device, email, and software policies to ensure secure cloud usage.

We recommend the companies check whether proper security and business-continuity-process models are implemented. Check whether companies have process for risk assessment, vulnerability assessment, risk capture, risk mitigation, etc. We advice you to also check if the company has implemented proper processes for handling security incidents by deploying proper backup and disaster recovery mechanisms. Check for idle resources the company has in reserve to implement business continuity process.

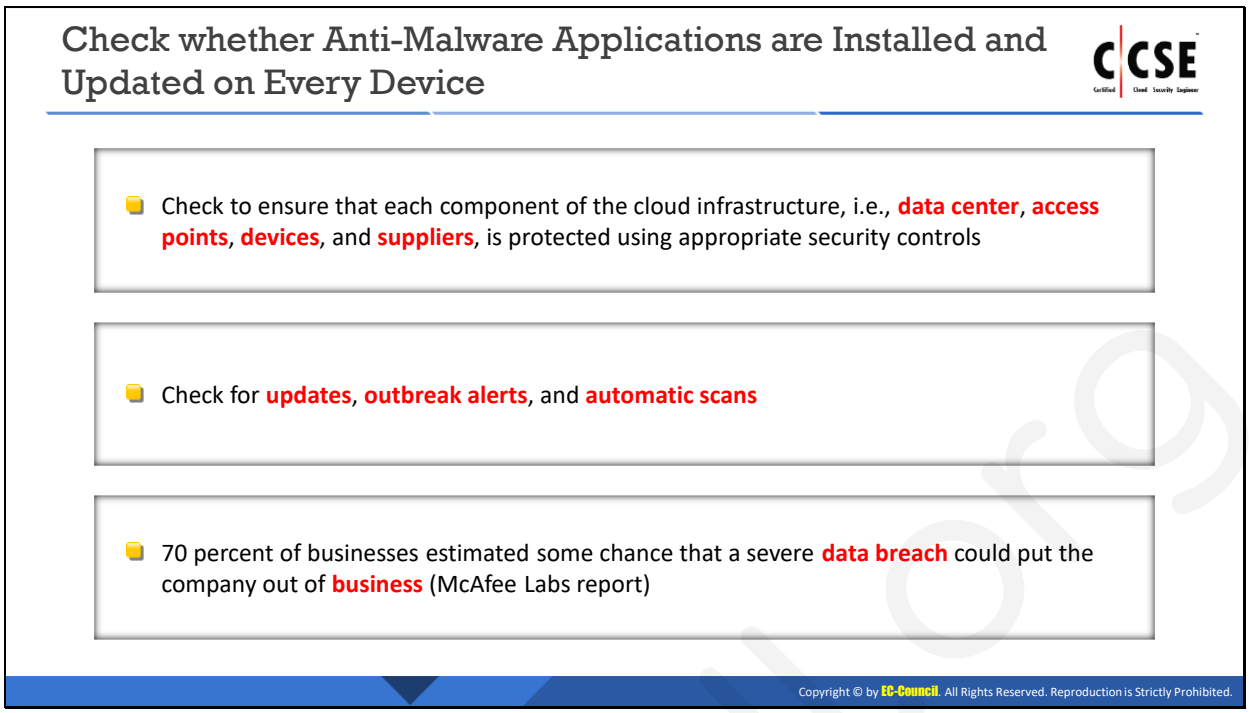


Check the Cloud for Resource Isolation

CSPs use multi-tenant and multi-deployment models to provide cloud services to the client. Improper isolation of data, app, and other client resources will impact resources of one client when other client is over using the resources or is under cyberattack. Therefore, we recommend the testers to always check if the CSP provides isolation of resources in cloud.

Penetrations may check resource isolation in a cloud using the following methods.

- **Check If Activity of One Subscriber Affects the Others:** We recommend you check if the performance of other cloud tenants is impacting the quality of service. Analyze the resource and bandwidth offered during normal hours and peak hours. Examine the logs and find service drops, slower connection speeds, inability to rescale the space, and other activities.
- **Check the CSP's Client Feedback and Expert Reviews:** Go through the CSP website, forums, and other social media campaigns to read the customer feedback, complaints, and expert reviews regarding the services provided as well as interruptions. Frequent complaints, bad feedback, and poor reviews display that the services are at fault.
- **Check the Track Record and Any Security of the CSP's Services:** Look at the past performance of cloud and verify if the CSP has any previous issues regarding the services and resource isolation activity. Check if the same issues exist on cloud by communicating with the users using same services.

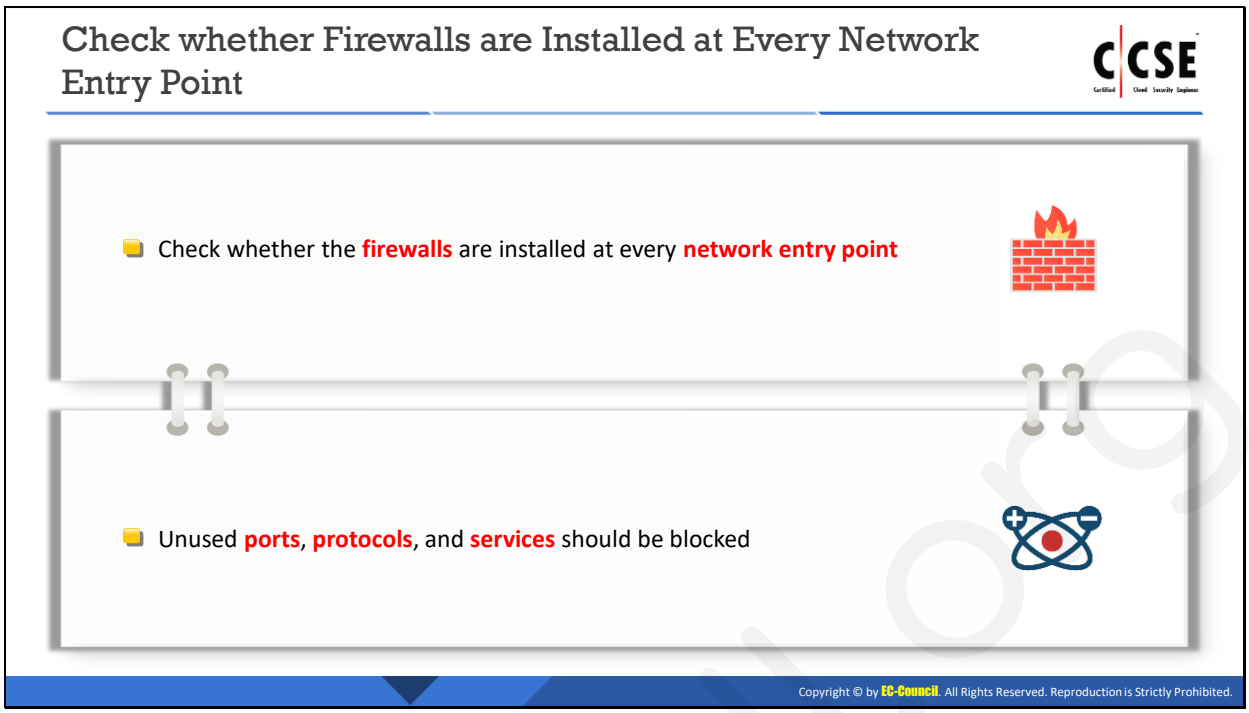


Check Whether Anti-Malware Applications are Installed and Updated on Every Device

We recommend the testers to ensure security of all the components used to access or build cloud services, such as system, app, server, database, routing devices, data center, access points, devices, and suppliers, etc. All these devices are also recommended to be enabled to handle and avert any malware cyberattacks targeted towards cloud. Therefore, ensure that all the components have antimalware apps along with facility to install regular updates automatically.

Each component may have a different malware component cyberattacking it, based on the operating system, functionality, and other specifications. 70 percent of businesses estimated some chance that a severe data breach could put the company out of business (McAfee Labs report). These components are recommended to contain proper antimalware apps that may detect customized risks and alert the administrator about the cyberattack.

You may use the audit reports to find all the components and devices at the client location as well as the antimalware solutions installed on them. You may also use manual methods of testing and verifying the apps installed on them.

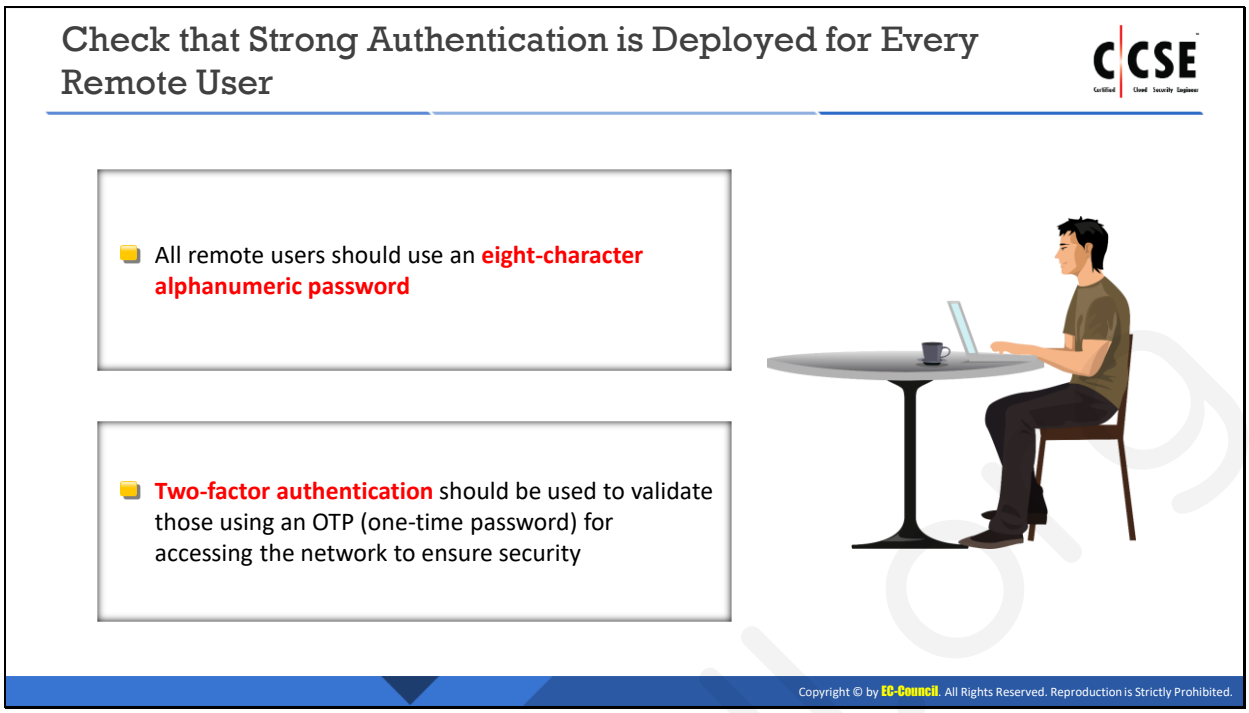


Check Whether Firewalls are Installed at Every Network Entry Point

Firewalls monitor incoming and outgoing network traffic and decide whether to allow or block specific traffic based on defined set of security rules. They also create logs of averted traffic along with other details. Therefore, we recommend the testers manually check if the CSP has installed firewalls at every network entry point to provided cloud services and verify their effectiveness in preventing transmission of malicious traffic.

We recommend you check the service provider's firewall implementation policy to find details about the type of firewall, update type and schedule, firewall rules, etc. Check the firewall logs and network configuration based on the firewall settings. We recommend the testers check the list of persons who may access the firewall, modify the configuration, and perform regular audits.

Finally, check the alert mechanism of the firewall and the process of managing the alerts and reports in the company. Ensure that the firewall is up to date and is capable of blocking all kinds of malicious traffic. Also ensure unused ports, protocols, and services are blocked.



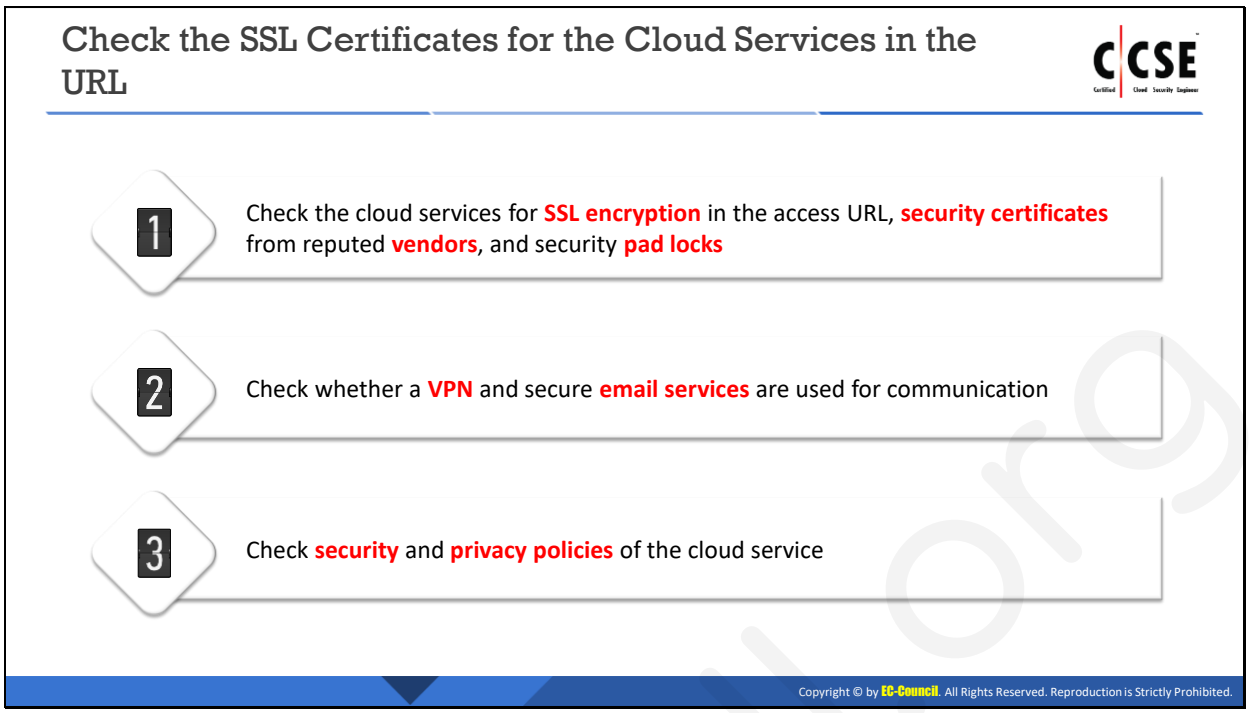
Check That Strong Authentication is Deployed for Every Remote User

In cloud computing, the major risk is the process of establishing a secure environment supported by strong authentication mechanism. In most of the cases, the client will be responsible for implementing a strong password policy for authentication of users trying to access cloud either from local or remote locations. The process is to ideally include a two-factor authentication process as well.

As a tester, we recommend you check if the client has implemented strong authentication policies for users accessing cloud services and if the company has a good password policy and if it has implemented it successfully.

We recommend that the strong authentication policy to include the following:

- The password to have at least eight characters that is alphanumeric in nature
- Include two-factor authentication to validate users by sending OTP (One Time Password) for accessing the network to ensure security
- The server to use private/public PKI key pairs to encrypt the transferred data
- Use encrypted communications only, such as SSH or VPNs
- Deploy MAC address or IP address filtering
- Deny telnet access to the unit, as it does not encrypt the communications channel



Check the SSL Certificates for the Cloud Services in the URL

To ensure the safe and secure file transfers to/from cloud servers, we recommend the tester to ensure that the CSP uses encryption on the transport layer of cloud. You may check the following to ensure that the CSP has proper encryption mechanism deployed:

- **Check Cloud Services for TLS Encryption**

  Transport Layer Security (TLS) encryption is essential for the transfer of files to/from cloud, as it establishes an encrypted link between cloud server and web browser of the user. Installing the SSL certificate on a cloud server activates the secure http protocol (https) and a padlock. The padlock and https depict that the connection established between cloud server and user's web browser is completely secure.

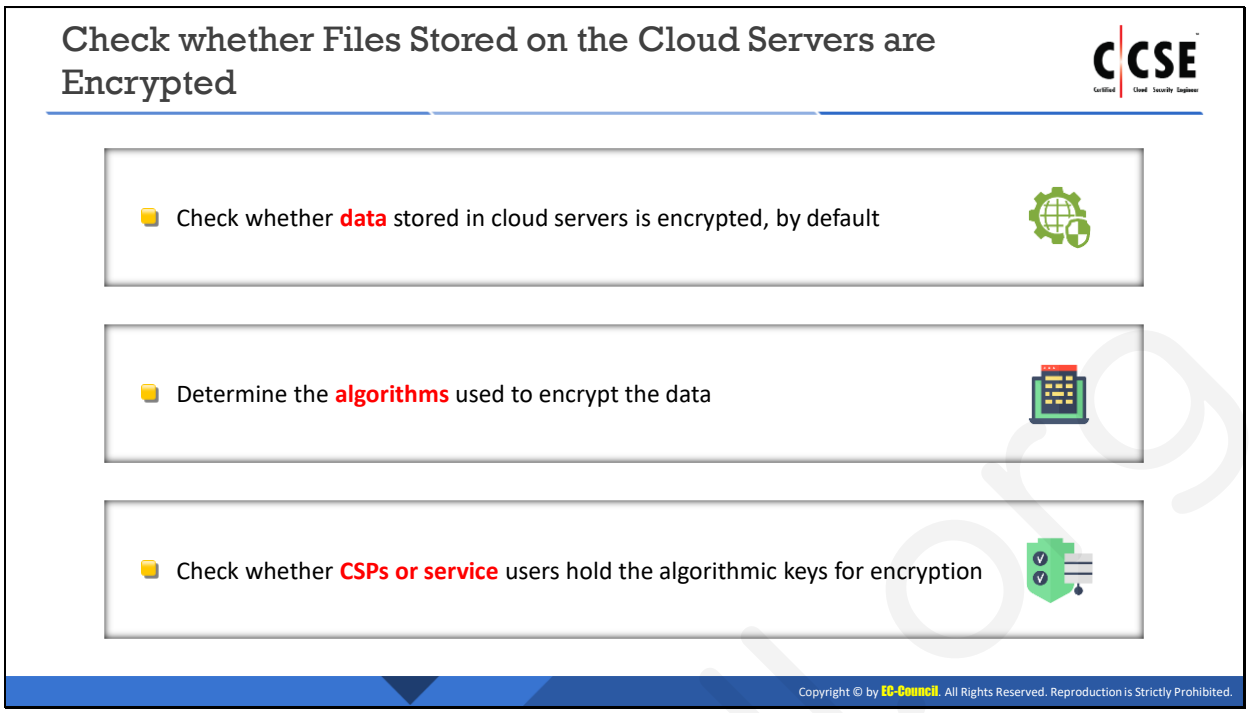
- **Check Whether Cloud Uses VPN and Secure Email Services for Communication**

  Virtual Private Network (VPN) uses various techniques such as tunneling for implementing a secure communication over the network. It encrypts the files and ensures that nobody on the public network could read the files, except the person having the right decryption key. Hence, we recommend the testers ensure the use of VPN services to secure file transfer in cloud environment.

  Various companies use virtual mail servers in the cloud to handle company mails and messages. We recommend the testers to ensure the security of these mail messages with use of proper encryption and security mechanisms.

- **Check Security and Privacy Policies of Cloud Service**

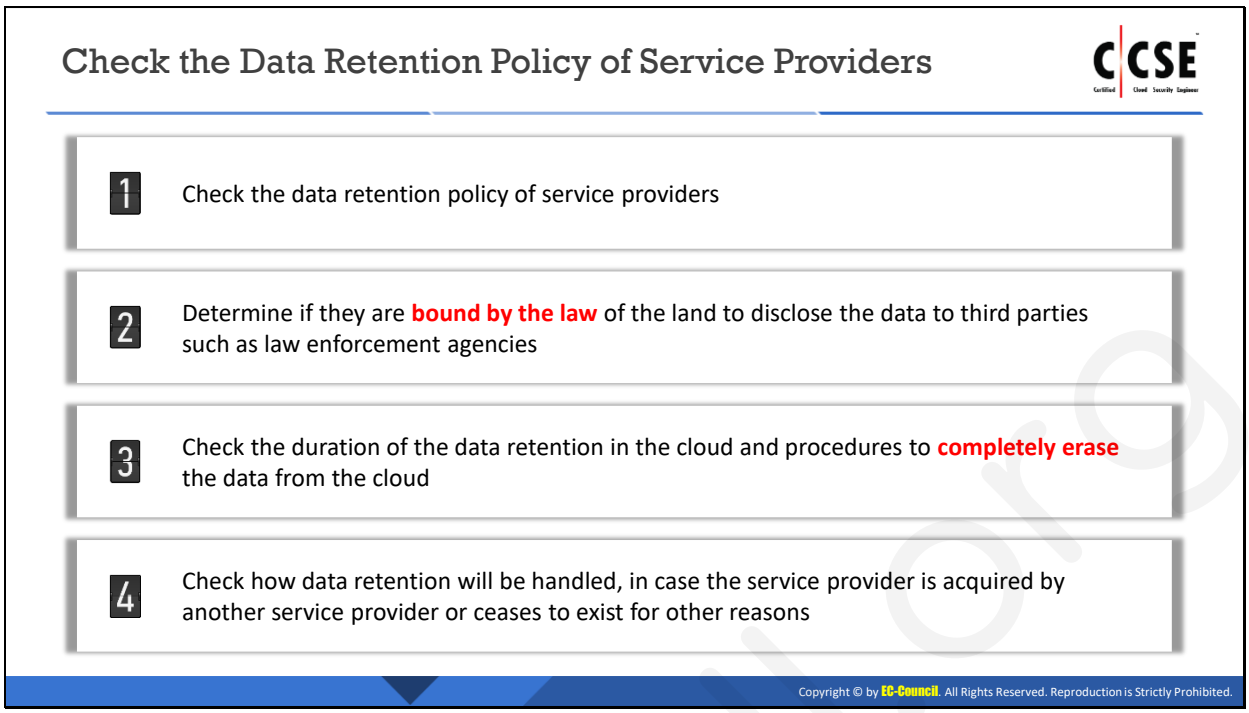
  Security and privacy policies of cloud service protect not only the integrity of systems and the data itself, but also maintain their customers' privacy.



Check Whether Files Stored on the Cloud Servers are Encrypted

We recommend CSPs to implement methods to encrypt not only the transferred files, but also the files stored on cloud by default. We recommend them to use strong algorithm to encrypt the data and use combination of public and private key pairs to encrypt the data. Encrypting the stored files will help in securing the data even if the hackers have gained unauthorized access to cloud.

We recommend you verify the algorithms that CSP uses to secure the data. Find the details of the authorities having access to the algorithm and its keys. It is important because, these keys may allow the users to decrypt the stored data.



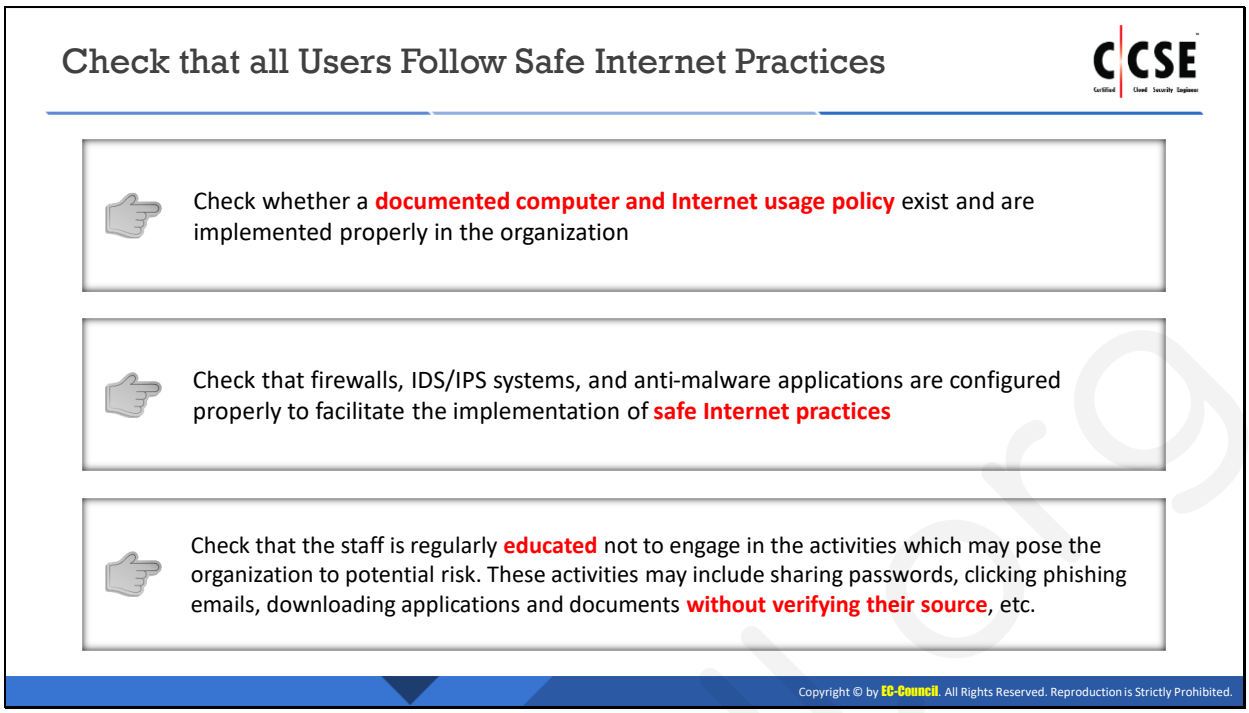
Check the Data Retention Policy of Service Providers

Data retention has become a big issue in financial services for regulatory and risk management reasons like the sudden market fall, sub-prime mortgage meltdown, financial standards caused by computer-generated algorithmic trades. We recommend you ensure that the cloud provider policies and data retention policies meet the company's needs and comply with internal corporate policy. For a cloud service provider, data retention assurance demonstration is easy when compared with data destruction. Make sure that the CSP perform regular backups and recovery tests to assure logical segregation and controls.

Find if the CSP has a data retention policy by reading the SLA document carefully. We recommend the testers evaluate the CSP's data retention policy and check its terms and conditions. Determine if the data retention policy of the CSP complies with the laws and regulations of the client country or region. Determine for sure that the process of data retention was followed and find the process of disclosing and the third parties to whom the CSP will disclose the data.

Verify if the duration of the data retention in cloud is minimal and does not result in losses to the client. Ensure that the CSP provides an option to completely erase the data from cloud.

Contact the CSP and check how it handles the data retention in case another company acquires the service provider or if the CSP ceases to exit or the business ceases to exist. Find the type of procedures they will follow to report changes to the client and the time interval and other provisions they will offer to the client to shift or extract the data or services present in cloud.



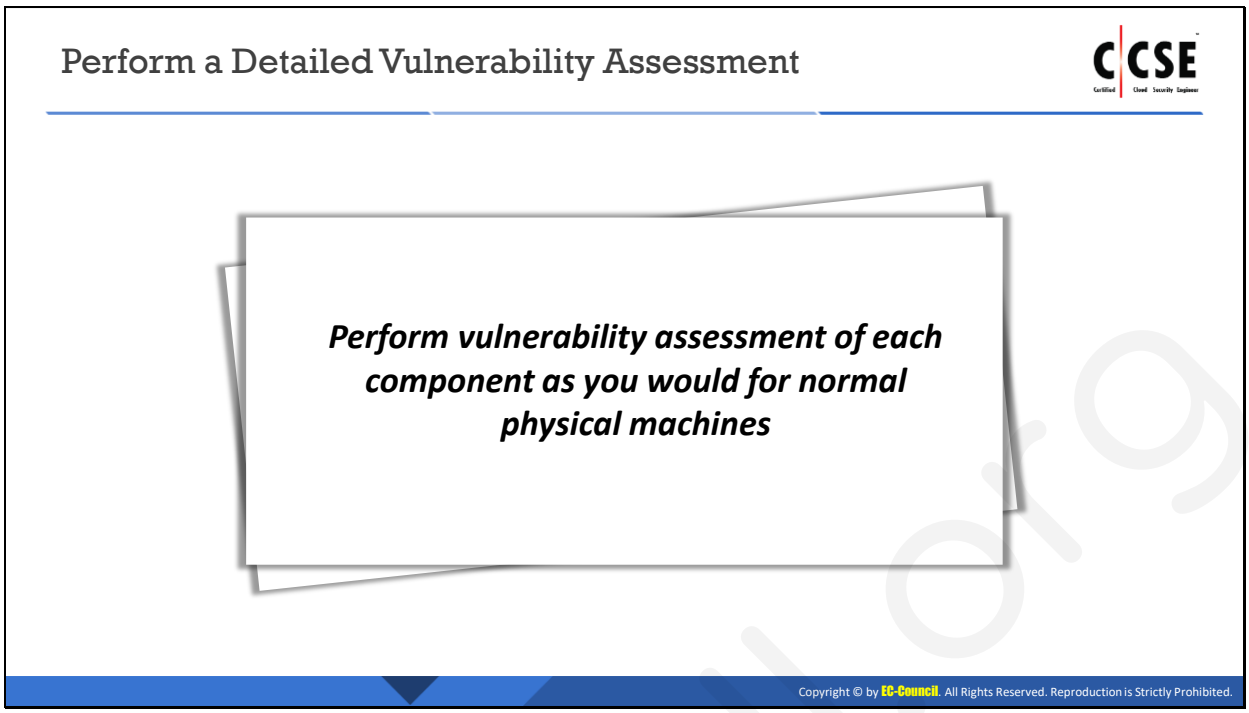
Check That All Users Follow Safe Internet Practices

All the companies implement internet usage policy that governs the employee internet usage from inside the company. This policy includes do's and don'ts for the employees to safely use the internet. We recommend the client companies using cloud services to also define and strictly implement such policies.

We recommend the testers check if the company has a documented computer and internet usage policy and effectiveness of implementing it across all the systems and devices. We recommend the testers check the logs for violation of the policies by the employees and ensure that the company has proper measures to tackle such events.

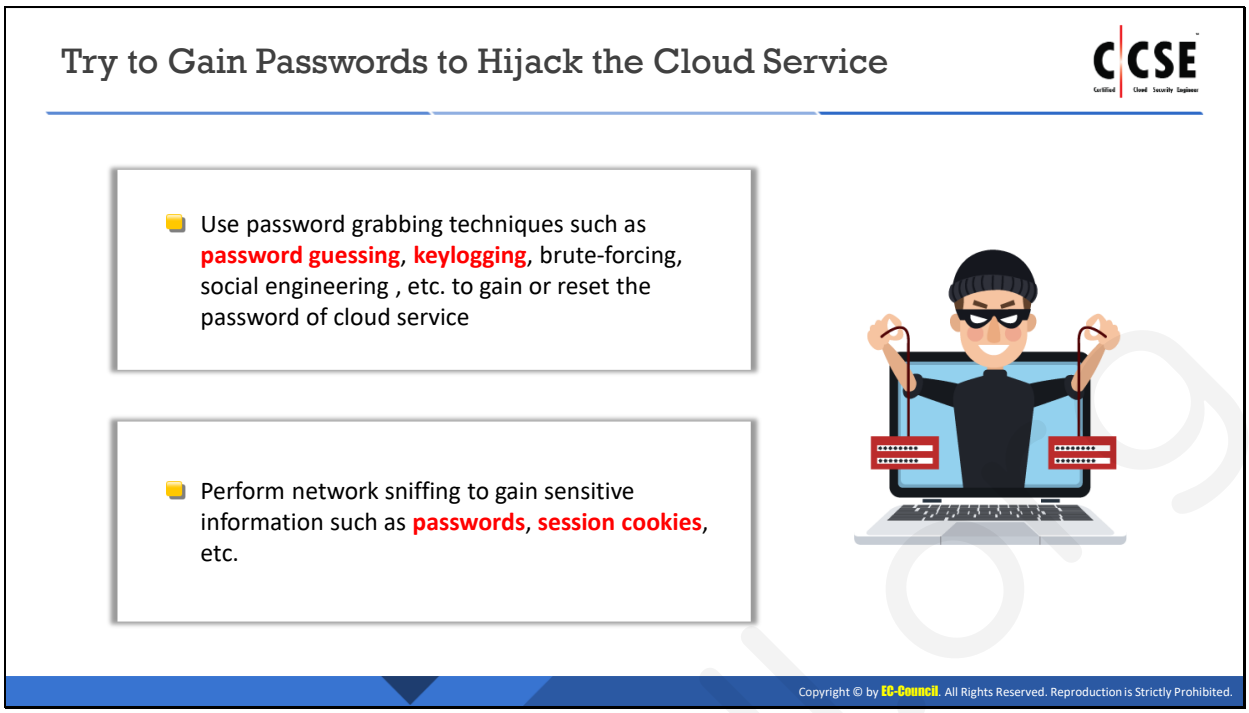
We recommend that the policy has to clearly state the implementation of security solutions, such as firewalls, IDS/IPS systems, and antimalware apps, across all the network devices. The policy should also define a process for alerting the security personnel in case of breach of the policy and actions they need to take.

Ensure that the company has hired educated staff, who also understand the importance of policies and implement them to maintain the security of data and other corporate assets. We recommend the policy define the process of framing security, creating passwords, responding to phishing emails, downloading apps and documents from the internet, etc. and the staff to have knowledge on the process of verifying source before downloading and verifying the information or the sender's email address and the information prior to sharing across the company.



Perform a Detailed Vulnerability Assessment

Vulnerability assessment refers to the process of scanning and discovering the security threats in cloud services. As a tester, we recommend you scan all cloud services for vulnerabilities using different methods and tools and analyze all the components of cloud for vulnerabilities. A tester should perform vulnerability assessment of each component as he would for normal physical machines.



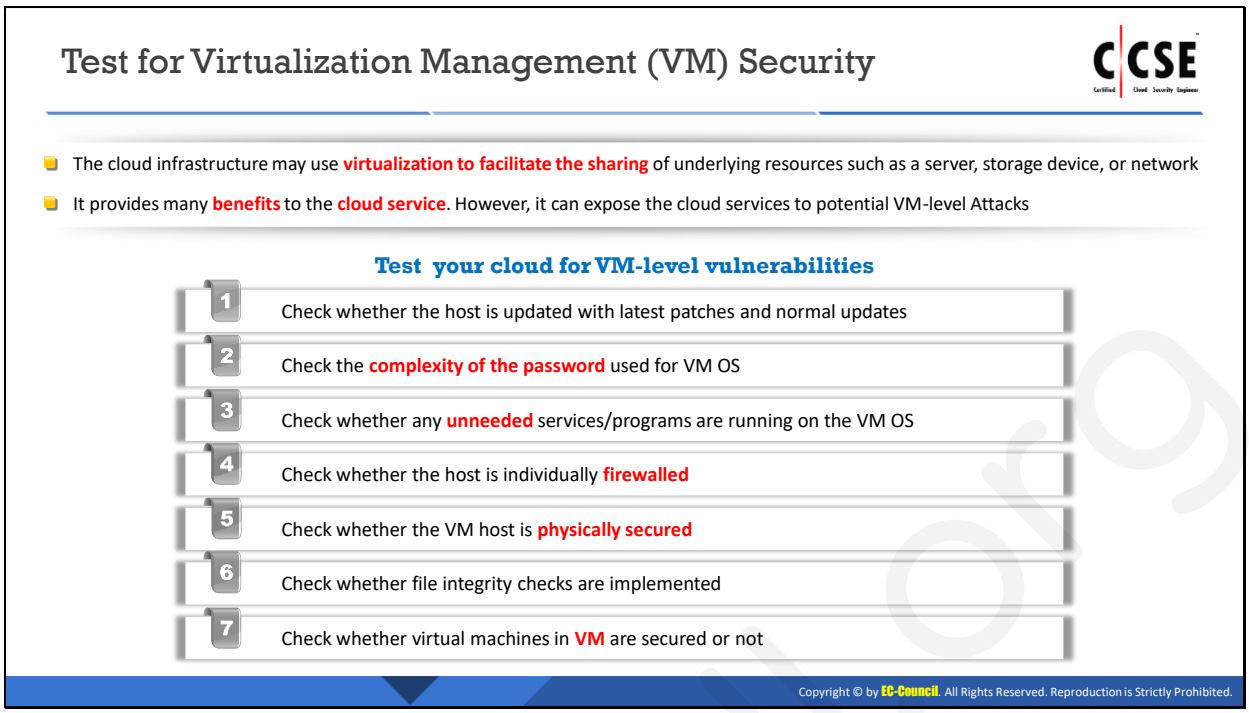
Try to Gain Passwords to Hijack the Cloud Service

In today's cyber world, hackers may find one or other way to penetrate cloud and gain access to the user's confidential data. To make sure that cloud is safe from various hackers' tactics, it is important to perform all possible tests on it.

Password grabbing, network sniffing, keylogging, Brute forcing, and social engineering are the most noted techniques that a hacker would use to gain your cloud password. To ensure that the password is safe and cloud is defensive against such password hacking techniques, we recommend you simulate similar password cracking cyberattacks against cloud resources handled by the client company.

We recommend you use password grabbing techniques, such as password guessing, keylogging, Brute forcing, social engineering, etc., to gain or reset the password of cloud service. Try these cyberattacks on cloud apps that the company controls. Other cyberattacks you are advised to perform include, network sniffing cyberattacks such as passwords, session cookies, etc., that may help to gain other sensitive information about cloud.

We recommend these tests to be inside the scope and not interrupt or impact any of the resources, which the company does not control.



Test for Virtualization Management (VM) Security

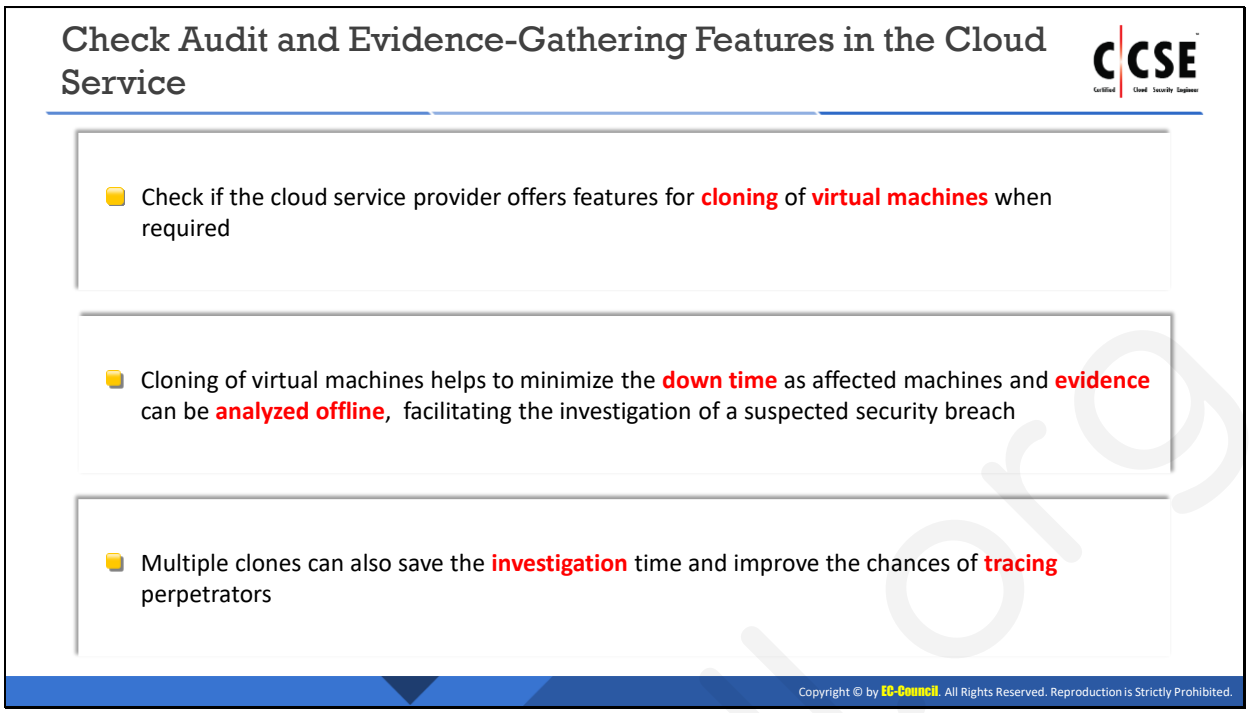
Cloud infrastructure uses the virtualization technique as it provides a blend of technologies as a single service, based on the user requirement. Virtual Management is the process of administrating and ensuring the functioning of the virtual platforms, infrastructure, storage devices, operating system, and every service that a virtual machine delivers.

There are various risks associated with virtualization in cloud. The most hazardous and noticeable risk is compromising the virtual machine hypervisor, which is the major source that provides all the virtualization services. It is also prone to cyberattacks due to network traffic that flows to/from the virtual machines.

We recommend you evaluate the security of the virtualized resources to find the vulnerabilities and flaws in their security configuration. This will help in eradicating the vulnerabilities and fixing the flaws.

Test the cloud for the following VM-level vulnerabilities:

- If CSP updates the host with latest patches and normal updates
- Complexity of the password used for VM OS
- Any unnecessary services/programs are running on the VM OS
- If the hosts have individual firewalls
- Check whether the VM host is physically secure
- If cloud has file integrity checks
- Check whether virtual machines in VM are secure

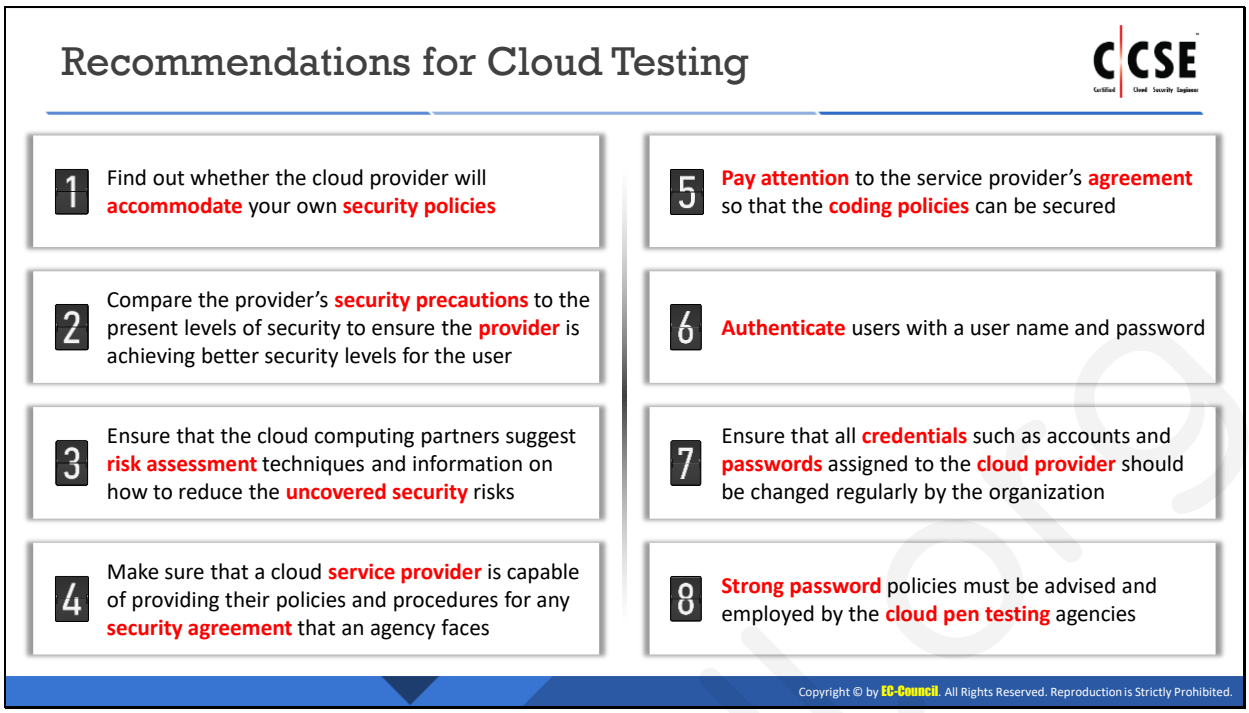


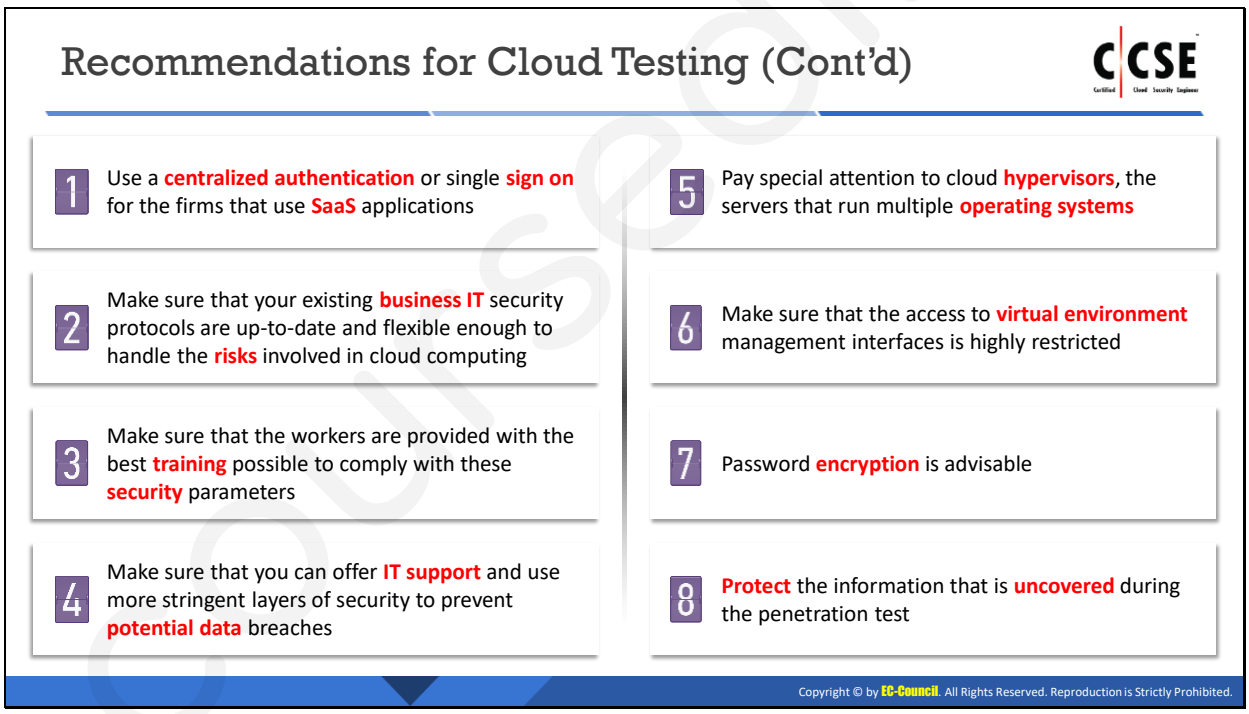
Check Audit and Evidence-Gathering Features in the Cloud Service

We recommend the testers check if the cloud provider includes provisions for performing audit and gathering evidence in case of a security incident. These will simplify the process of investigation and forensics to find the cyberattack path, method, and source. Check if cloud service provider offers features for cloning of virtual machines when required.

Cloning of virtual machines helps in minimizing the down-time, because they allow the investigators to clone the affected machines and use them to gather evidence for investigation without disrupting the services. The CSP has to provide enough storage to create multiple clones and analyze them in parallel for the sake of reducing investigation time as well as for improving chances of tracing perpetrators.

Check for the mention of such features across the SLA and also check the type of provisions the CSP would provide in case of a cyberattack and how it may help in the investigation process.



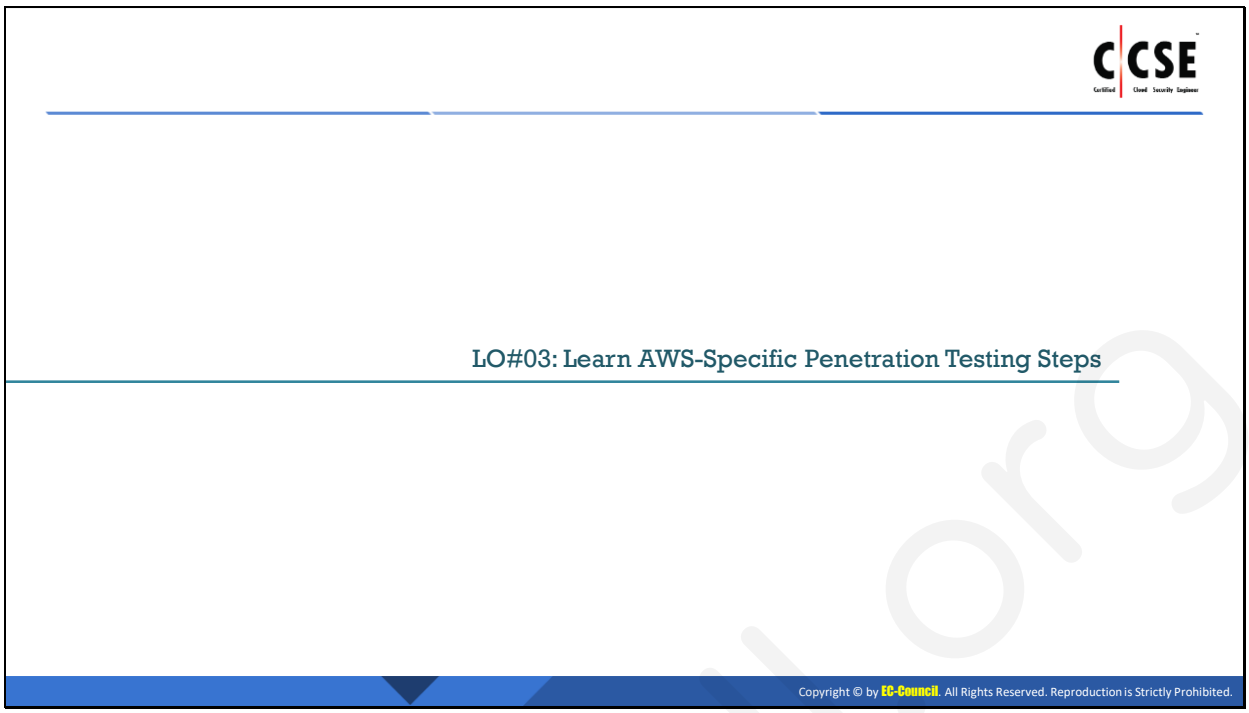


Recommendations for Cloud Testing

Some recommendations for the CSP to follow to secure cloud are as follows:

- Find out whether cloud provider will accommodate your own security policies
- Compare the provider's security precautions to the present levels of security to ensure the provider is achieving better security levels for the user

- Ensure that cloud computing partners suggest risk assessment techniques and information on how to reduce the uncovered security risks
- Make sure that a cloud service provider may provide their policies and procedures for any security agreement that an agency faces
- Pay attention to the service provider's agreement to secure the coding policies
- Authenticate users with a user name and password each
- Ensure that all credentials, such as accounts and passwords assigned to cloud provider is advised to be changed regularly by the company
- Strong password policies are advised as mandatory and need to be deployed by cloud pen testing agencies
- Use a centralized authentication or single sign on for the firms that use SaaS apps
- Make sure that your existing business IT security protocols are up-to-date and flexible enough to handle the risks involved in cloud computing
- Make sure that the workers are provided with the best training possible to comply with these security parameters
- Make sure that you may offer IT support and use more stringent layers of security to prevent potential data breaches
- Pay special attention to cloud hypervisors, the servers that run multiple operating systems
- Make sure that the access to virtual environment management interfaces is highly restricted
- Password encryption is advisable
- Protect the information that is uncovered during the test

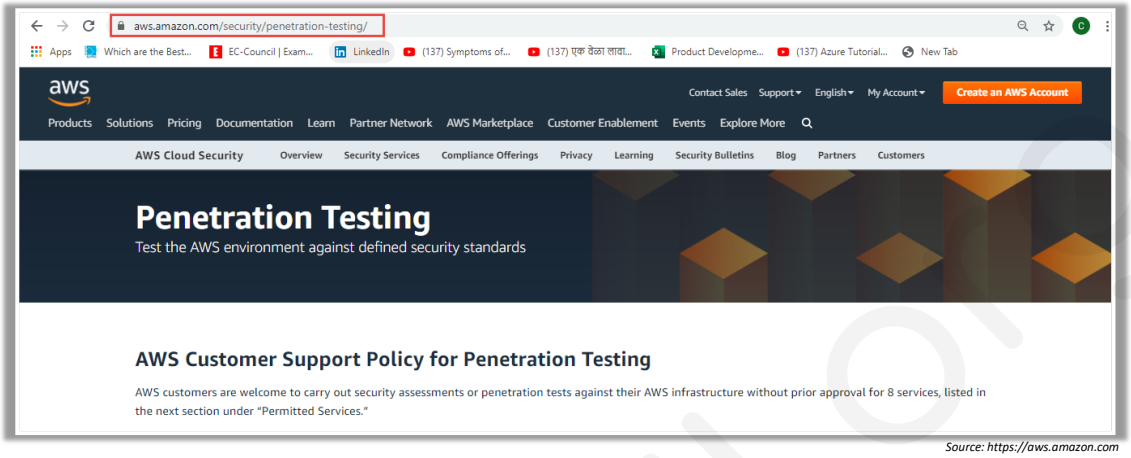


LO#03: Learn AWS-Specific Penetration Testing Steps

The objective of this section is to familiarize with and understand policies, permissions, procedures, terms, and conditions regarding AWS penetration testing.

Understand AWS Penetration Testing Policy and Procedures

- Visit AWS website to familiarize with and understand **policies, permissions, procedures, terms,** and **conditions** regarding AWS penetration testing



Source: https://aws.amazon.com

Understand AWS Penetration Testing Policy and Procedures (Cont'd)

Customer Service Policy for Penetration Testing

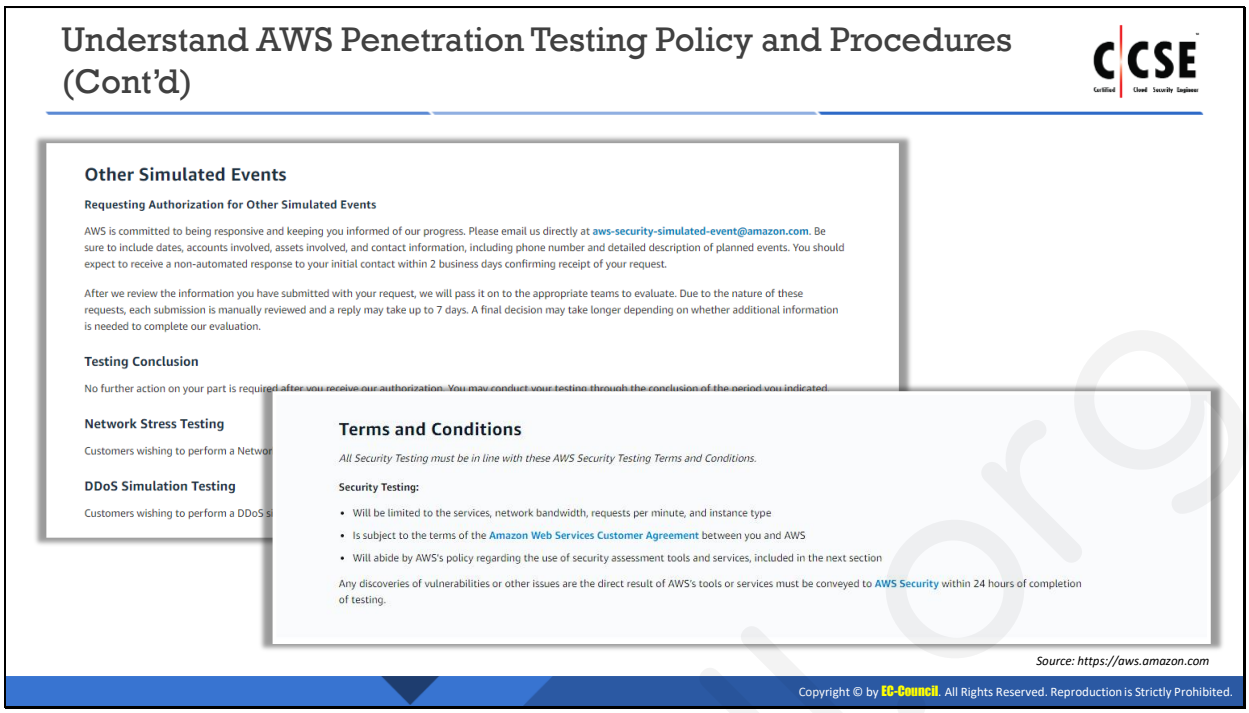
Permitted Services

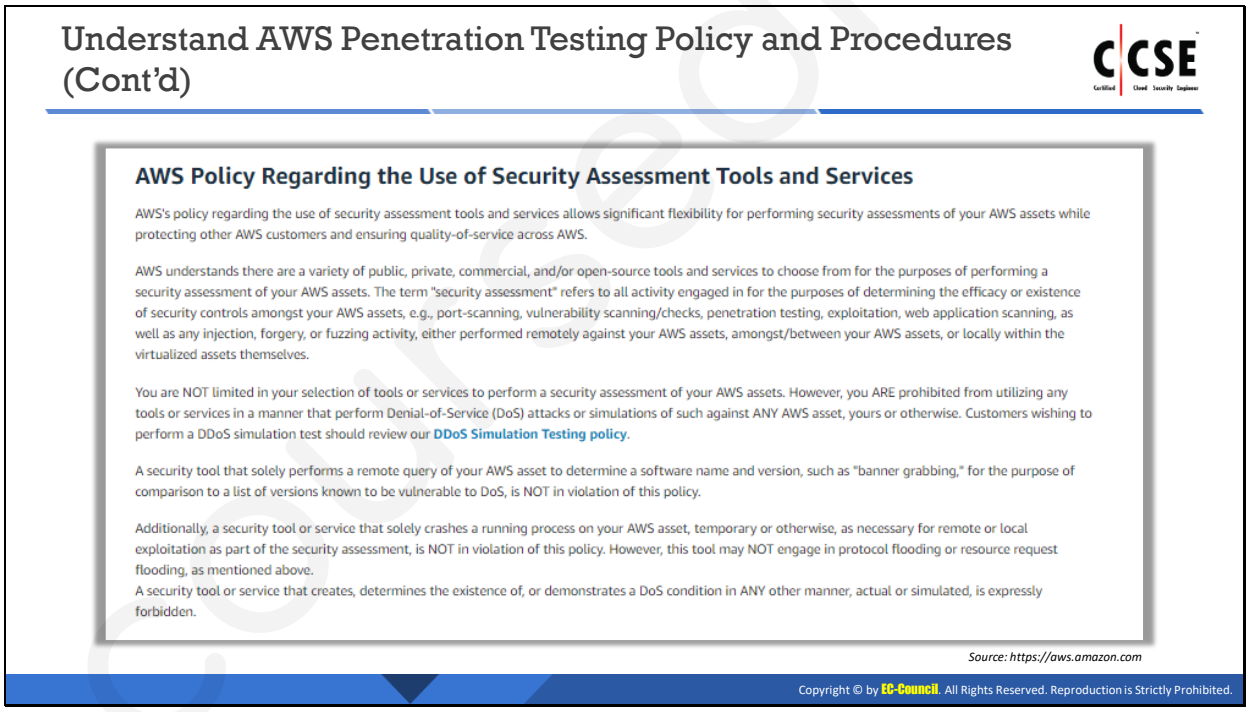
- Amazon EC2 instances, NAT Gateways, and Elastic Load Balancers
- Amazon RDS
- Amazon CloudFront
- Amazon Aurora
- Amazon API Gateways
- AWS Lambda and Lambda Edge functions
- Amazon Lightsail resources
- Amazon Elastic Beanstalk environments

Prohibited Activities

- DNS zone walking via Amazon Route 53 Hosted Zones
- Denial of Service (DoS), Distributed Denial of Service (DDoS), Simulated DoS, Simulated DDoS (These are subject to the DDoS Simulation Testing policy)
- Port flooding
- Protocol flooding
- Request flooding (login request flooding, API request flooding)

Source: https://aws.amazon.com





Understand AWS Penetration Testing Policy and Procedures

Visit AWS website to familiarize with and understand policies, permissions, procedures, terms, and conditions regarding AWS penetration testing.

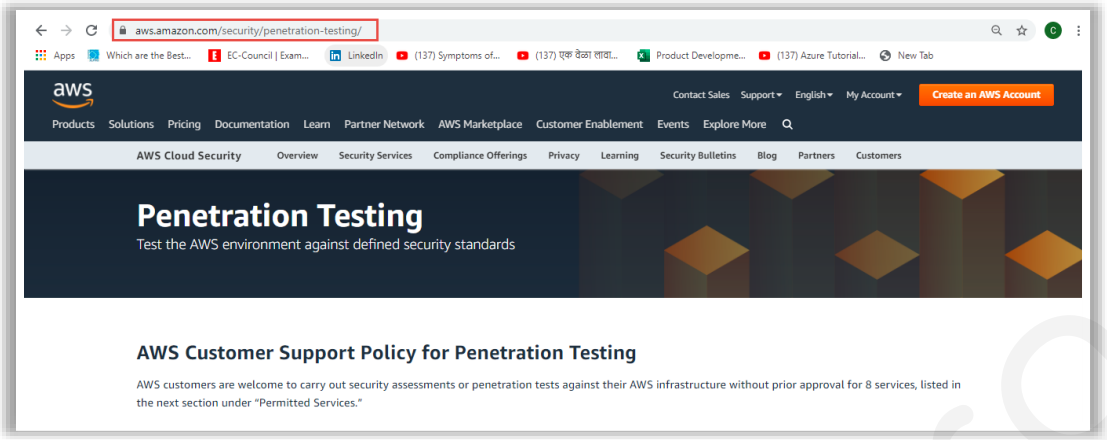


Figure 6.2: AWS Website Showing AWS Penetration Testing Guidelines

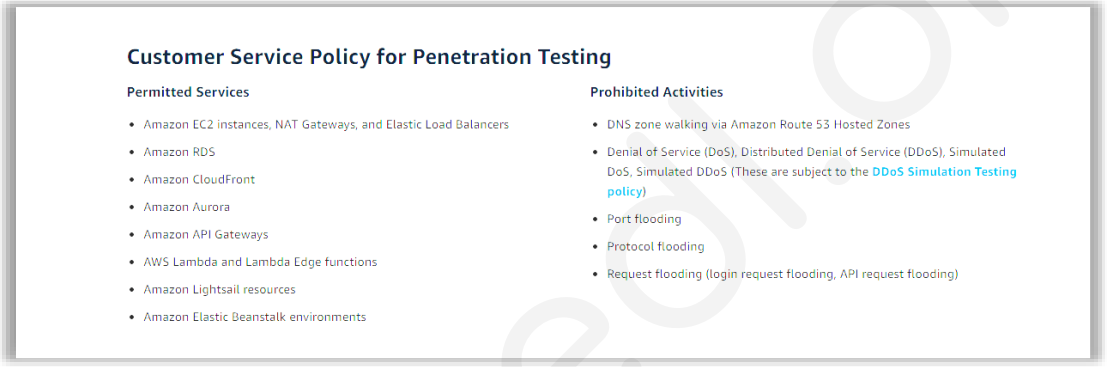


Figure 6.3: AWS Website Showing Customer Service Policy for Penetration Testing

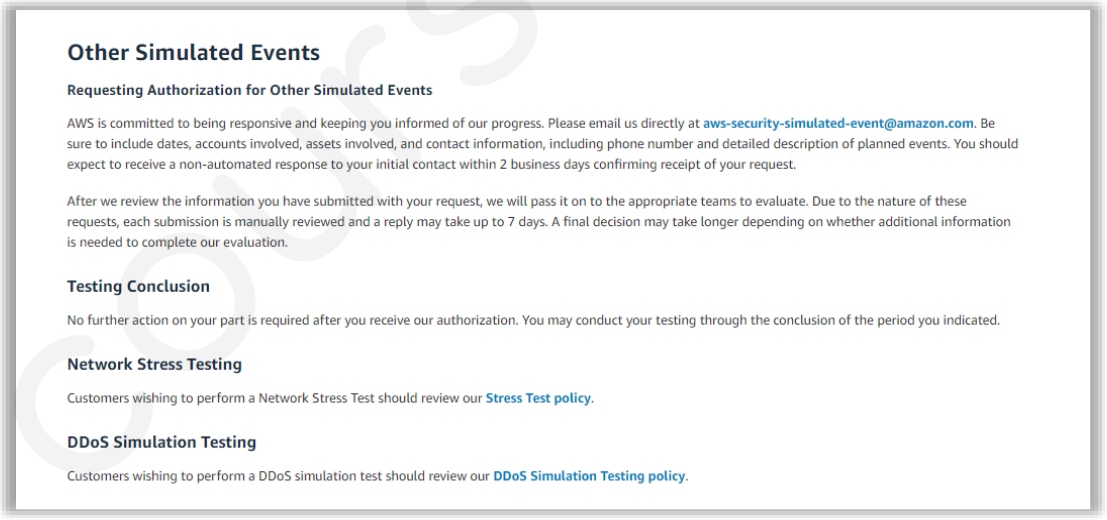
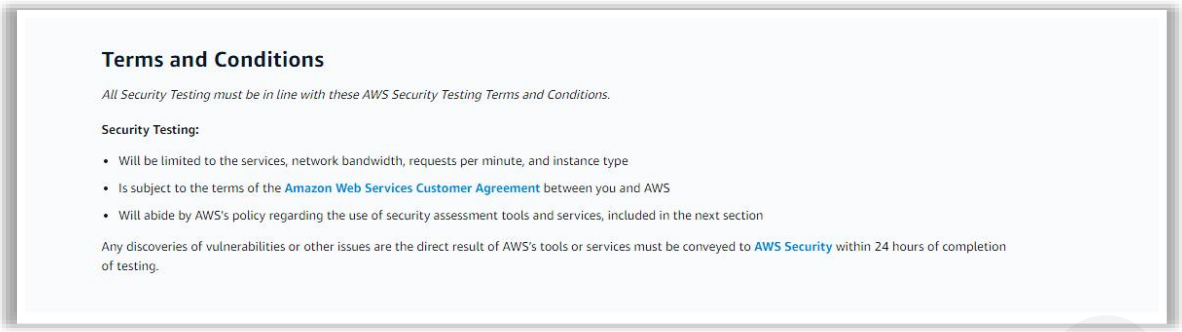


Figure 6.4: AWS Website Showing Customer Guidelines for Penetration Testing



Terms and Conditions

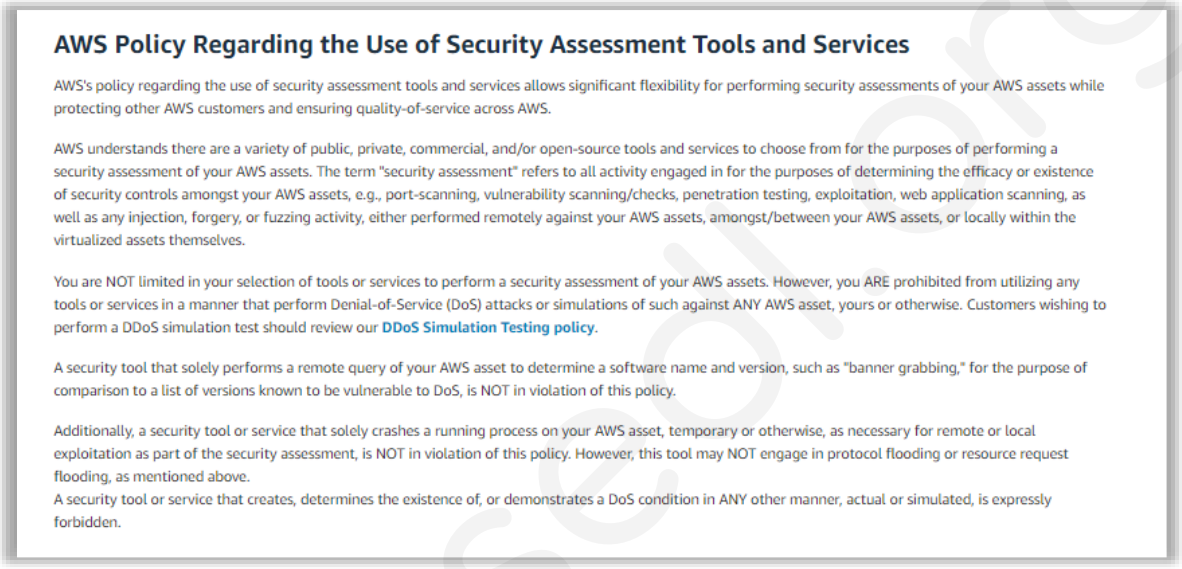
All Security Testing must be in line with these AWS Security Testing Terms and Conditions.

Security Testing:

- Will be limited to the services, network bandwidth, requests per minute, and instance type
- Is subject to the terms of the **Amazon Web Services Customer Agreement** between you and AWS
- Will abide by AWS's policy regarding the use of security assessment tools and services, included in the next section

Any discoveries of vulnerabilities or other issues are the direct result of AWS's tools or services must be conveyed to **AWS Security** within 24 hours of completion of testing.

Figure 6.5: AWS Website Showing Customer Terms and Conditions for Penetration Testing



AWS Policy Regarding the Use of Security Assessment Tools and Services

AWS's policy regarding the use of security assessment tools and services allows significant flexibility for performing security assessments of your AWS assets while protecting other AWS customers and ensuring quality-of-service across AWS.

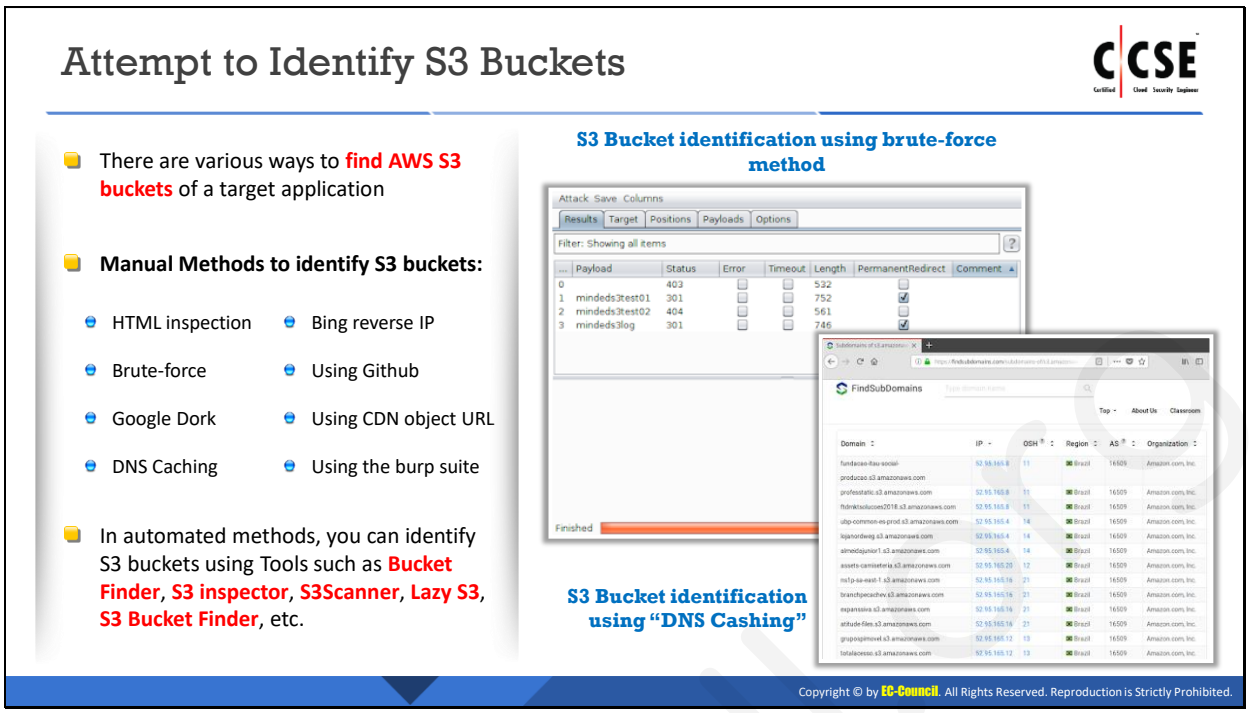
AWS understands there are a variety of public, private, commercial, and/or open-source tools and services to choose from for the purposes of performing a security assessment of your AWS assets. The term "security assessment" refers to all activity engaged in for the purposes of determining the efficacy or existence of security controls amongst your AWS assets, e.g., port-scanning, vulnerability scanning/checks, penetration testing, exploitation, web application scanning, as well as any injection, forgery, or fuzzing activity, either performed remotely against your AWS assets, amongst/between your AWS assets, or locally within the virtualized assets themselves.

You are NOT limited in your selection of tools or services to perform a security assessment of your AWS assets. However, you ARE prohibited from utilizing any tools or services in a manner that perform Denial-of-Service (DoS) attacks or simulations of such against ANY AWS asset, yours or otherwise. Customers wishing to perform a DDoS simulation test should review our **DDoS Simulation Testing policy**.

A security tool that solely performs a remote query of your AWS asset to determine a software name and version, such as "banner grabbing," for the purpose of comparison to a list of versions known to be vulnerable to DoS, is NOT in violation of this policy.

Additionally, a security tool or service that solely crashes a running process on your AWS asset, temporary or otherwise, as necessary for remote or local exploitation as part of the security assessment, is NOT in violation of this policy. However, this tool may NOT engage in protocol flooding or resource request flooding, as mentioned above.
A security tool or service that creates, determines the existence of, or demonstrates a DoS condition in ANY other manner, actual or simulated, is expressly forbidden.

Figure 6.6: AWS Website Showing Guidelines for Using Security Assessment Tools and Services



Attempt to Identify S3 Buckets

There are various ways to find AWS S3 buckets of a target application. Manual Methods to identify S3 buckets:

- HTML inspection
- Brute-force
- Google Dork
- DNS Caching
- Bing reverse IP
- Using Github
- Using CDN object URL
- Using the burp suite

In automated methods, you can identify S3 buckets using Tools such as Bucket Finder, S3 inspector, S3Scanner, Lazy S3, S3 Bucket Finder, etc.

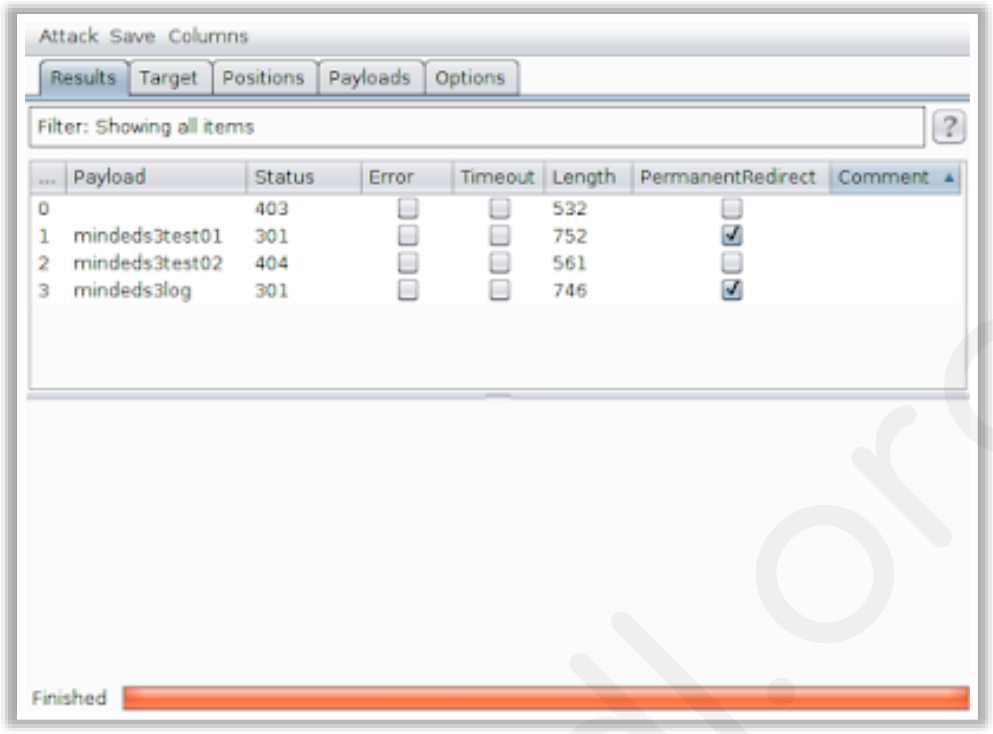


Figure 6.7: S3 Bucket Identification Using Brute-Force Method

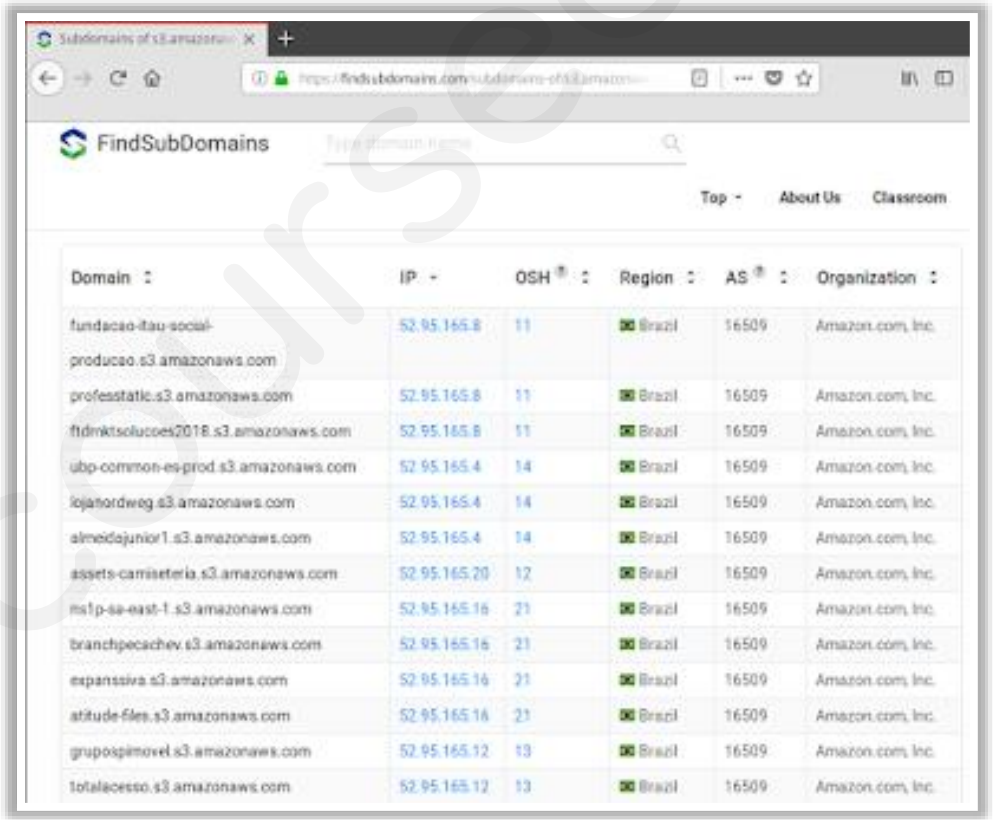
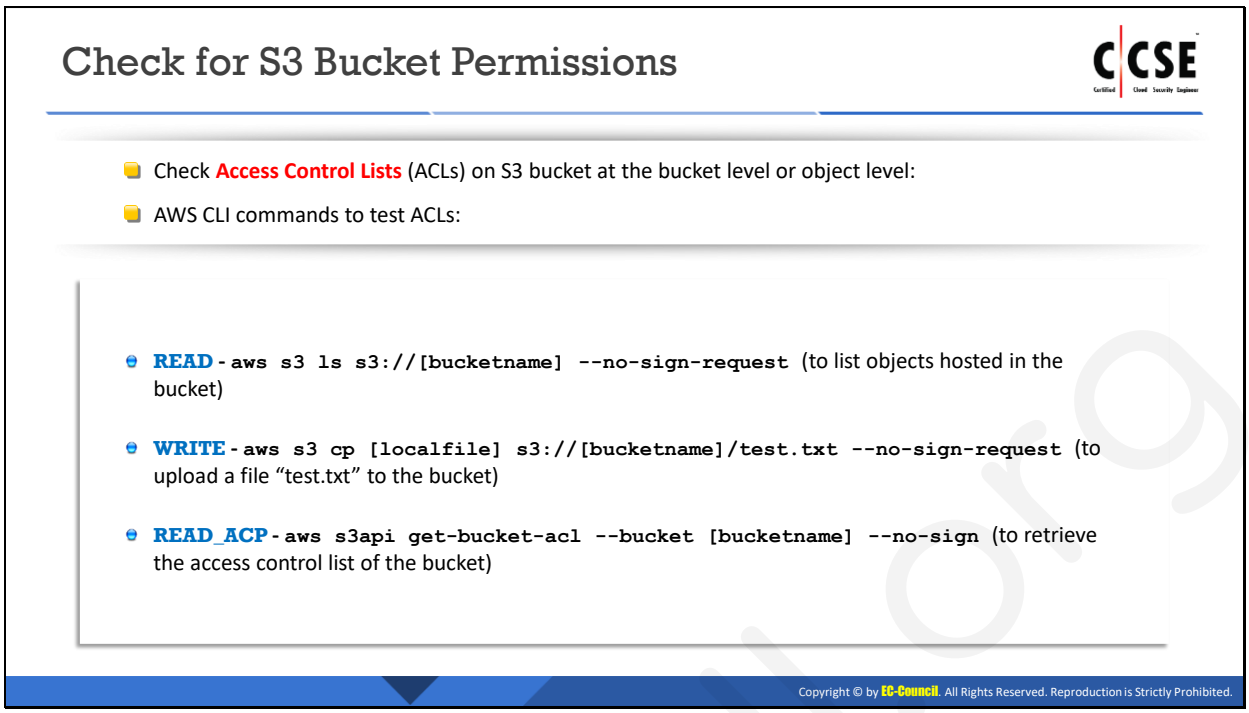


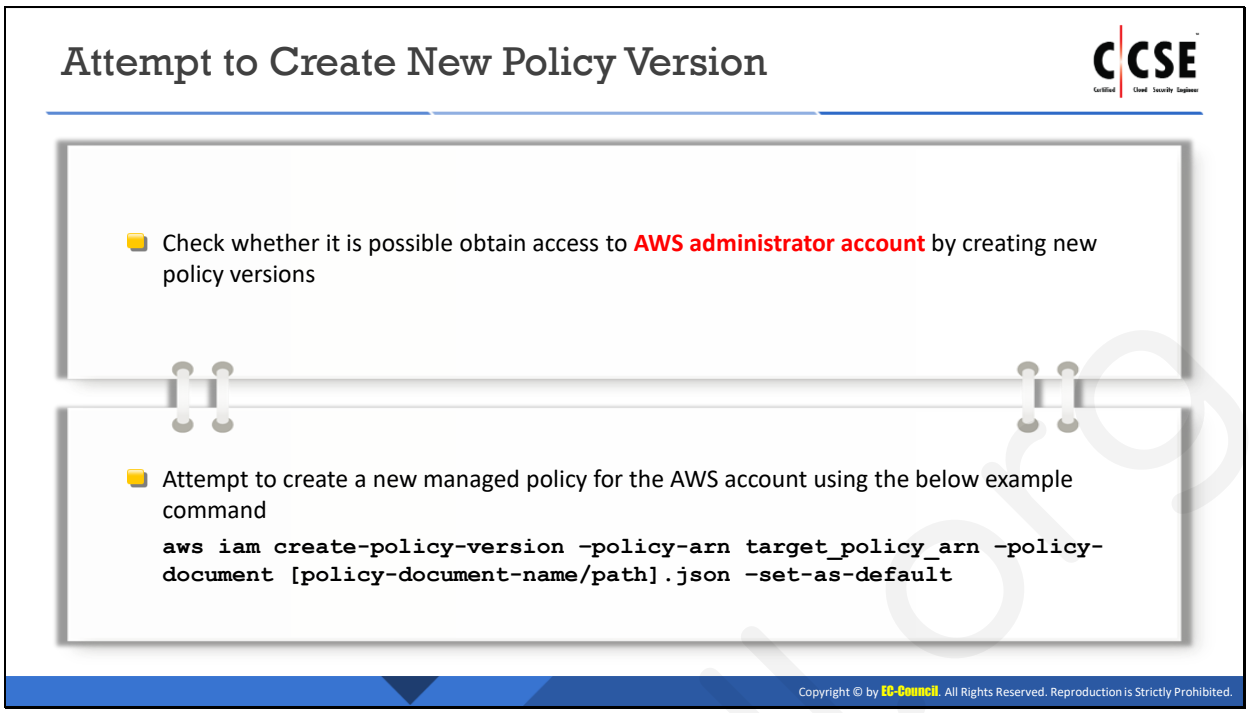
Figure 6.8: S3 Bucket Identification Using “DNS Caching”



Check for S3 Bucket Permissions

Check Access Control Lists (ACLs) on S3 bucket at the bucket level or object level. AWS CLI commands to test ACLs:

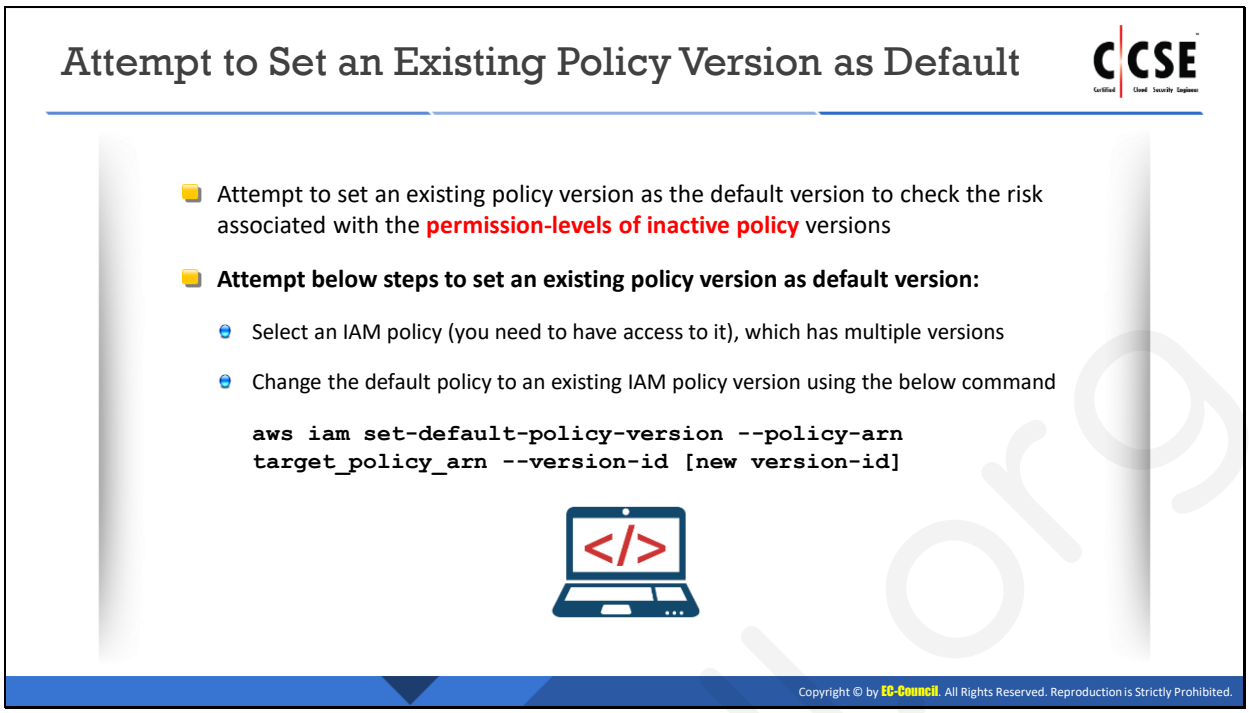
- **READ** - **`aws s3 ls s3://[bucketname] --no-sign-request`** (to list objects hosted in the bucket)
- **WRITE** - **`aws s3 cp localfile s3://[bucketname]/test.txt --no-sign-request`** (to upload a file "test.txt" to the bucket)
- **READ_ACP** - **`aws s3api get-bucket-acl --bucket [bucketname] --no-sign`** (to retrieve the access control list of the bucket)
- **WRITE_ACP** - **`aws s3api put-bucket-acl --bucket [bucketname] [ACLPERMISSIONS] --no-sign-request`** (to set the access control list of the bucket (WRITE_ACP) without actually changing it)



Attempt to Create New Policy Version

Check whether it is possible obtain access to AWS administrator account by creating new policy versions. Attempt to create a new managed policy for the AWS account using the below example command:

```
aws iam create-policy-version –policy-arn target_policy_arn –policy-document [policy-document-name/path].json –set-as-default
```

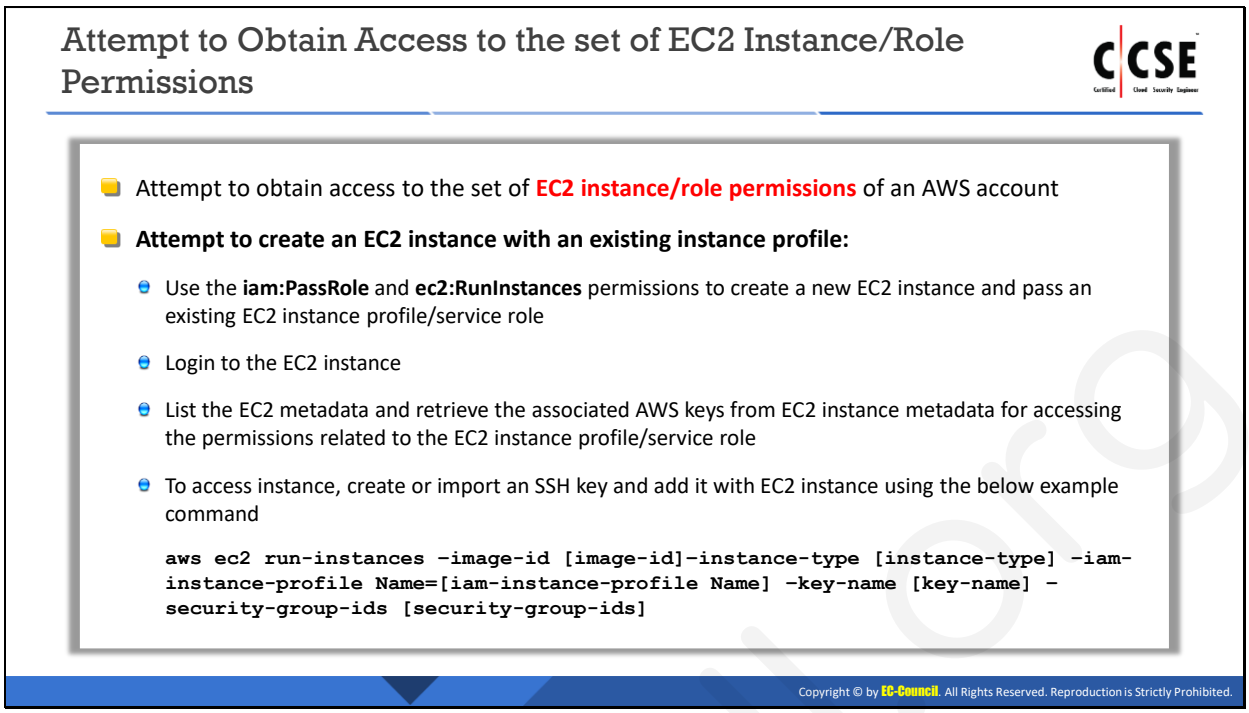


Attempt to Set an Existing Policy Version as Default

Attempt to set an existing policy version as the default version to check the risk associated with the permission-levels of inactive policy versions. Attempt below steps to set an existing policy version as default version:

- Select an IAM policy (you need to have access to it), which has multiple versions.
- Change the default policy to an existing IAM policy version using the below command:

```
aws iam set-default-policy-version --policy-arn target_policy_arn
--version-id [new version-id]
```

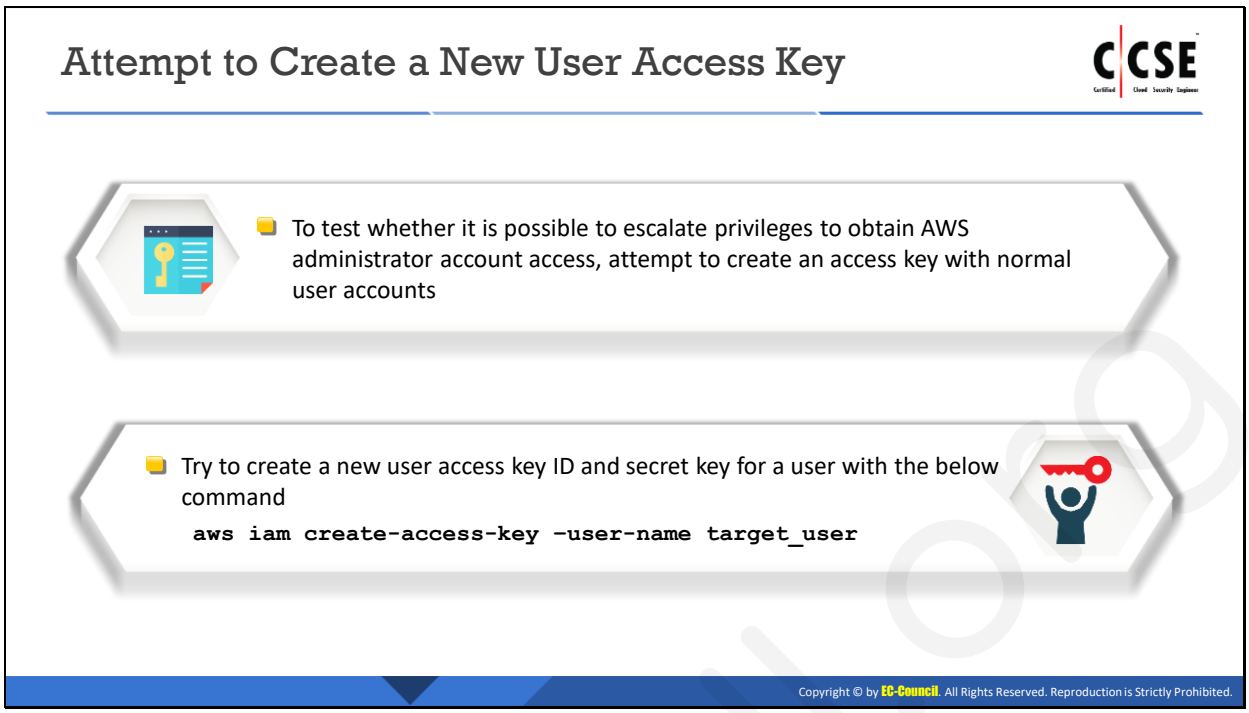


Attempt to Obtain Access to the Set of EC2 Instance/Role Permissions

Attempt to obtain access to the set of EC2 instance/role permissions of an AWS account. Attempt to create an EC2 instance with an existing instance profile:

- Use the **`iam:PassRole`** and **`ec2:RunInstances`** permissions to create a new EC2 instance and pass an existing EC2 instance profile/service role
- Login to the EC2 instance
- List the EC2 metadata and retrieve the associated AWS keys from EC2 instance metadata for accessing the permissions related to the EC2 instance profile/service role
- To access instance, create or import an SSH key and add it with EC2 instance using the below example command

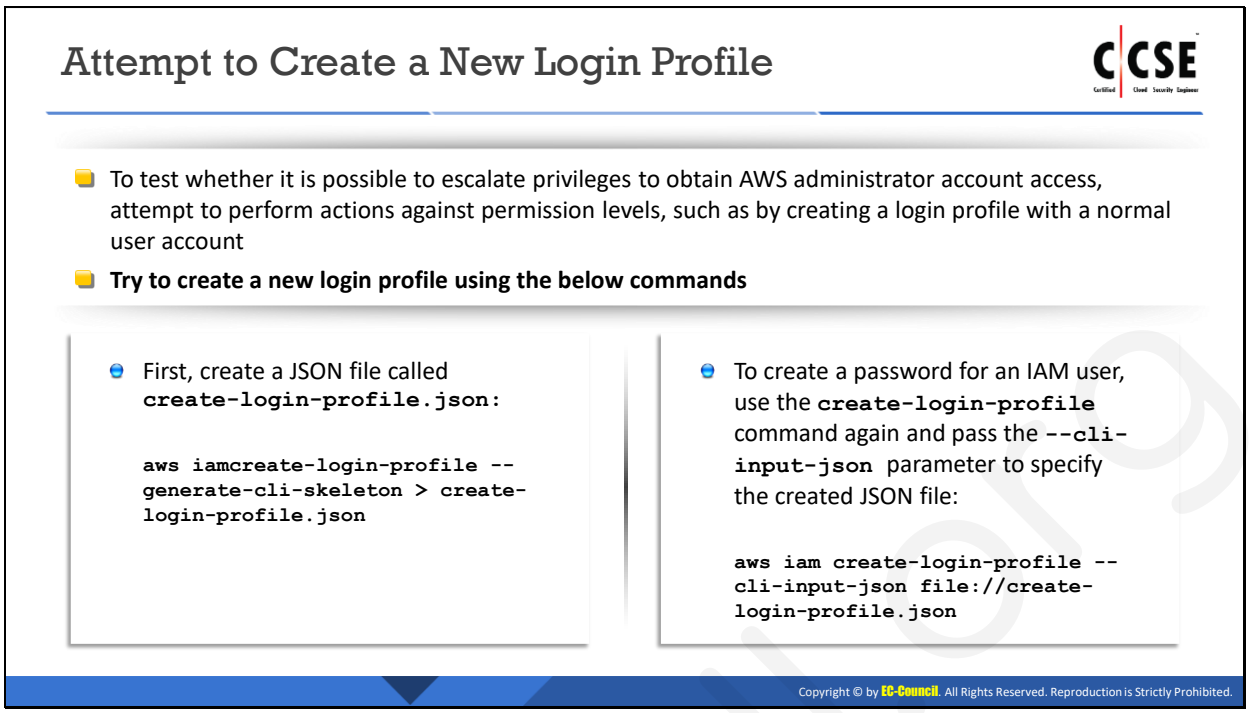
```
aws ec2 run-instances –image-id [image-id]–instance-type
[instance-type] –iam-instance-profile Name=[iam-instance-profile
Name] –key-name [key-name] –security-group-ids [security-group-
ids]
```



Attempt to Create a New User Access Key

To test whether it is possible to escalate privileges to obtain AWS administrator account access, attempt to create an access key with normal user accounts. Try to create a new user access key ID and secret key for a user with the below command.

```
aws iam create-access-key –user-name target_user
```



Attempt to Create a New Login Profile

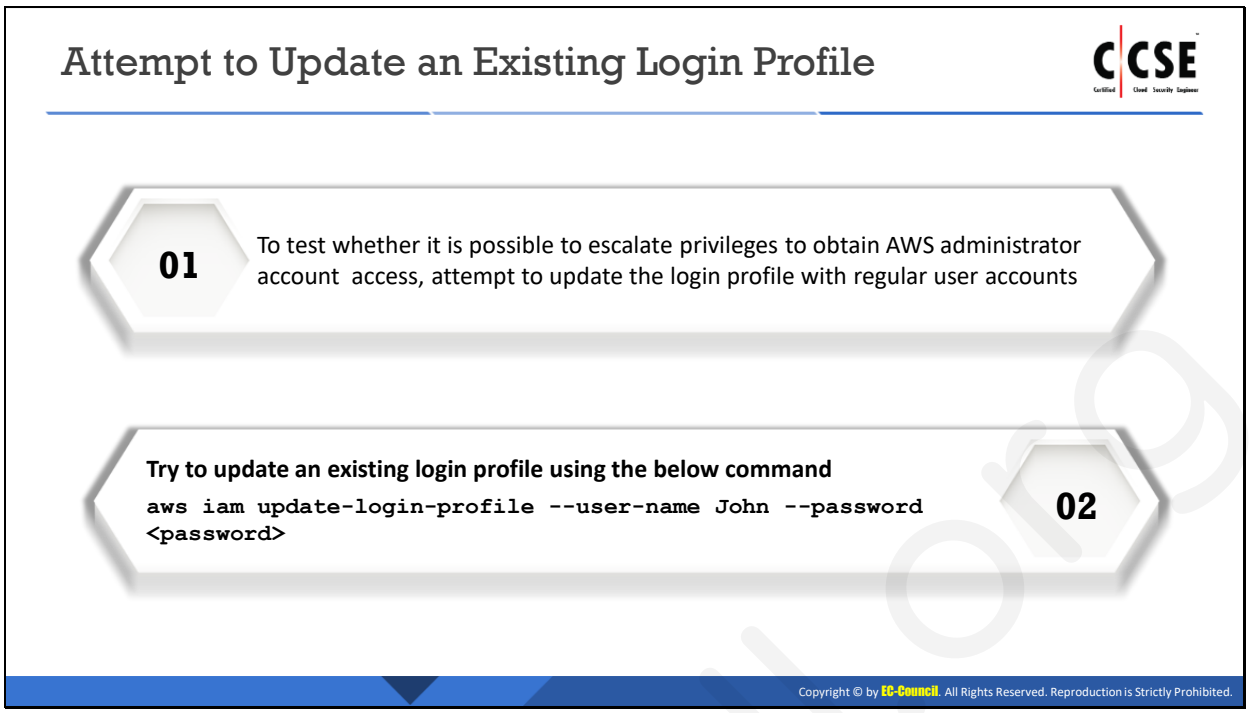
To test whether it is possible to escalate privileges to obtain AWS administrator account access, attempt to perform actions against permission levels, such as by creating a login profile with a normal user account. Try to create a new login profile using the below commands:

- First, create a JSON file called **`create-login-profile.json`**:

  `aws iamcreate-login-profile --generate-cli-skeleton > create-login-profile.json`

- To create a password for an IAM user, use the **`create-login-profile`** command again and pass the **`--cli-input-json`** parameter to specify the created JSON file:

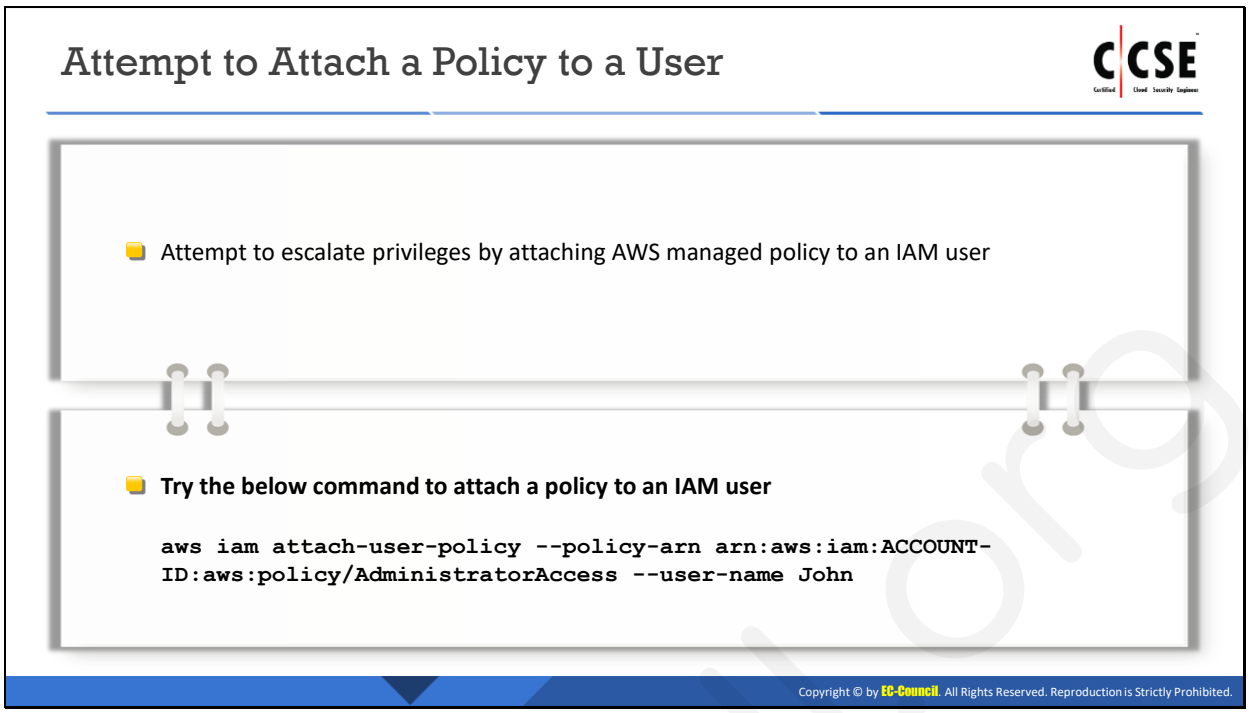
  `aws iam create-login-profile --cli-input-json file://create-login-profile.json`



Attempt to Update an Existing Login Profile

To test whether it is possible to escalate privileges to obtain AWS administrator account access, attempt to update the login profile with regular user accounts. Try to update an existing login profile using the below command:

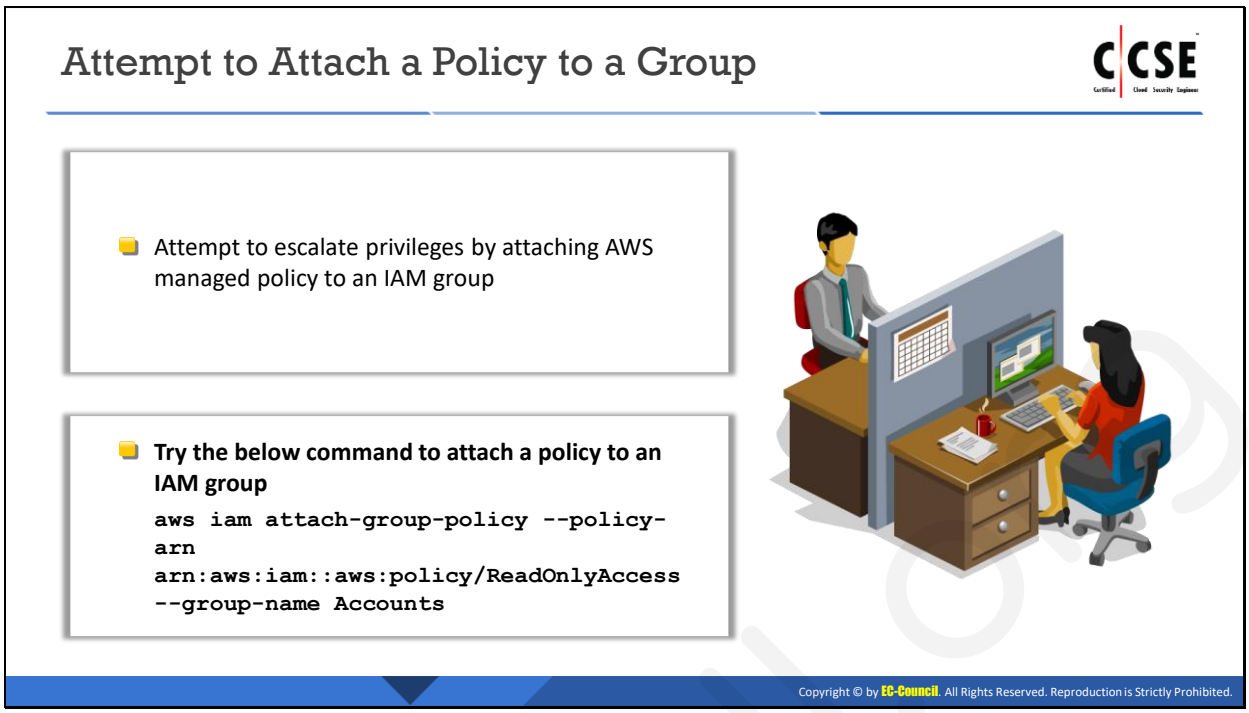
```
aws iam update-login-profile --user-name John --password <password>
```



Attempt to Attach a Policy to a User

Attempt to escalate privileges by attaching AWS managed policy to an IAM user. Try the below command to attach a policy to an IAM user:

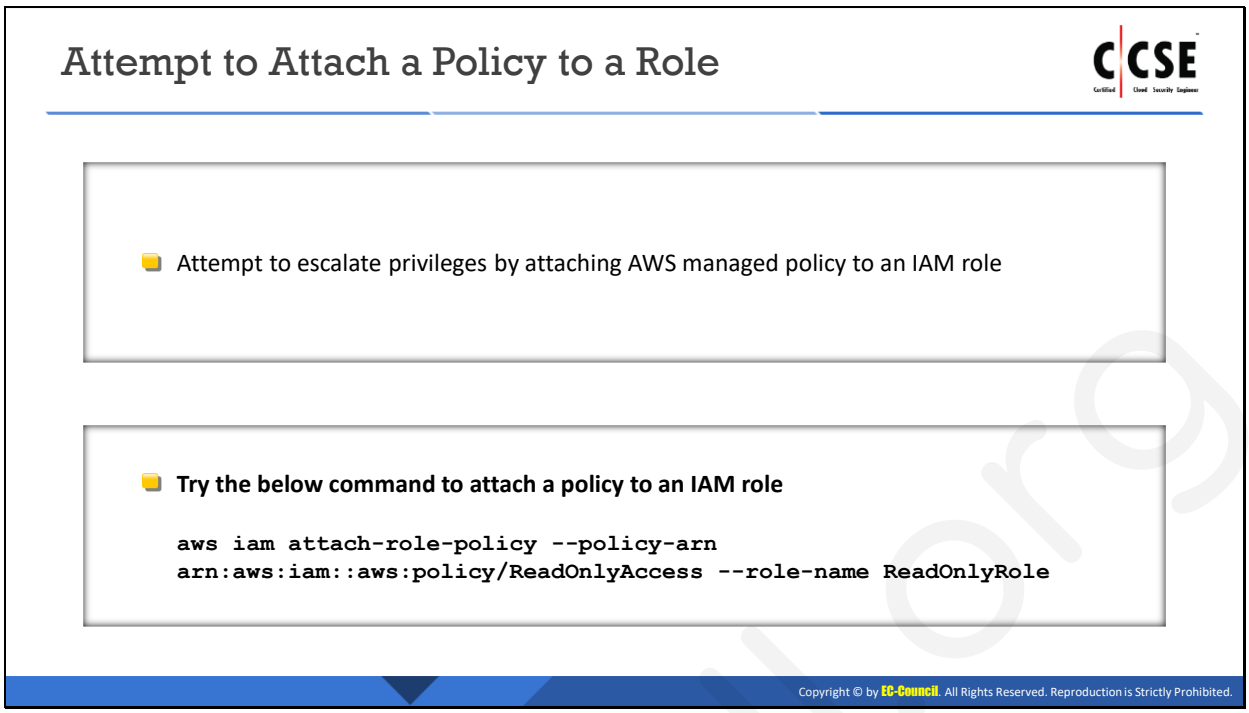
```
aws iam attach-user-policy --policy-arn arn:aws:iam:ACCOUNT-ID:aws:policy/AdministratorAccess --user-name John
```



Attempt to Attach a Policy to a Group

Attempt to escalate privileges by attaching AWS managed policy to an IAM group. Try the below command to attach a policy to an IAM group:

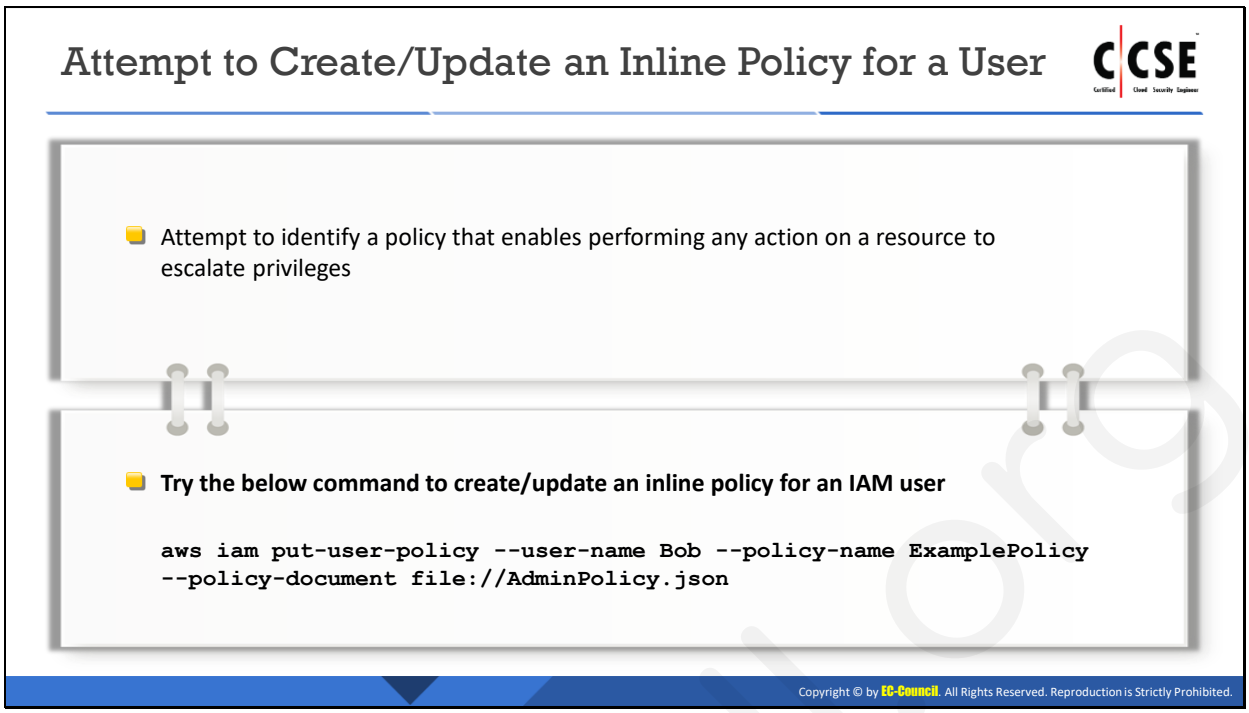
```
aws iam attach-group-policy --policy-arn
arn:aws:iam::aws:policy/ReadOnlyAccess --group-name Accounts
```



Attempt to Attach a Policy to a Role

Attempt to escalate privileges by attaching AWS managed policy to an IAM role. Try the below command to attach a policy to an IAM role:

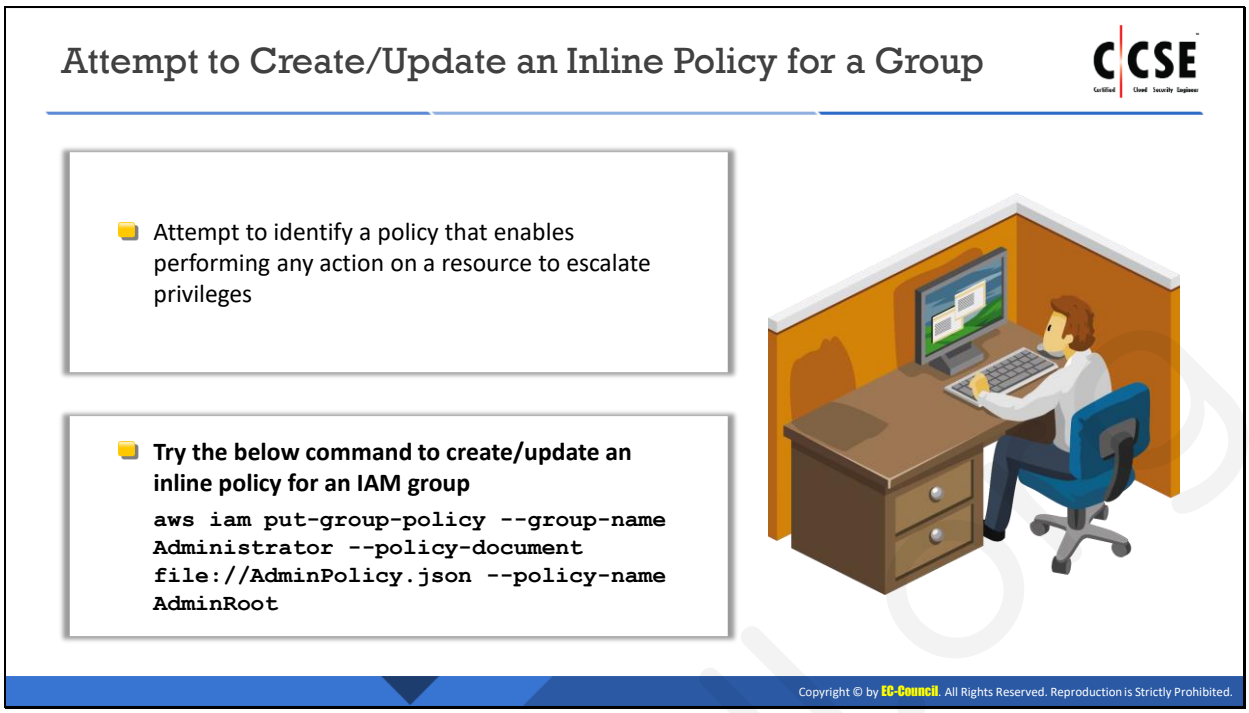
```
aws iam attach-role-policy --policy-arn
arn:aws:iam::aws:policy/ReadOnlyAccess --role-name ReadOnlyRole
```



Attempt to Create/Update an Inline Policy for a User

Attempt to identify a policy that enables performing any action on a resource to escalate privileges. Try the below command to create/update an inline policy for an IAM user:

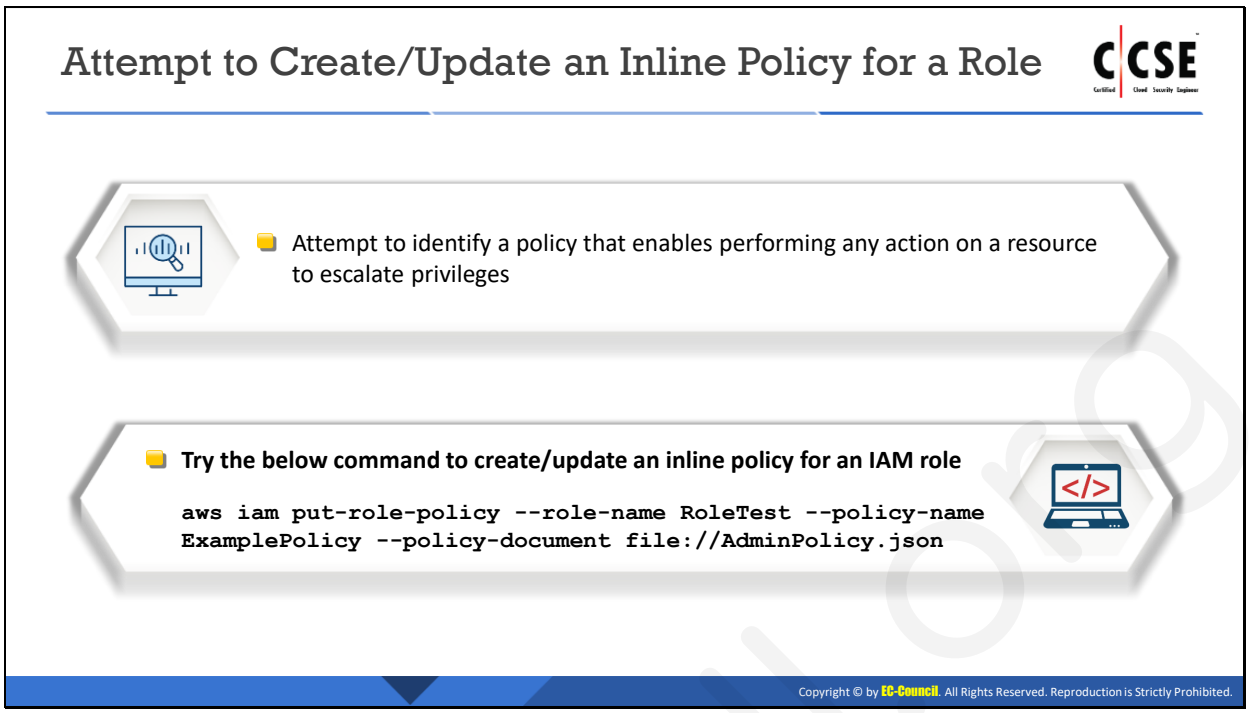
```
aws iam put-user-policy --user-name Bob --policy-name ExamplePolicy --
policy-document file://AdminPolicy.json
```



Attempt to Create/Update an Inline Policy for a Group

Attempt to identify a policy that enables performing any action on a resource to escalate privileges. Try the below command to create/update an inline policy for an IAM group:

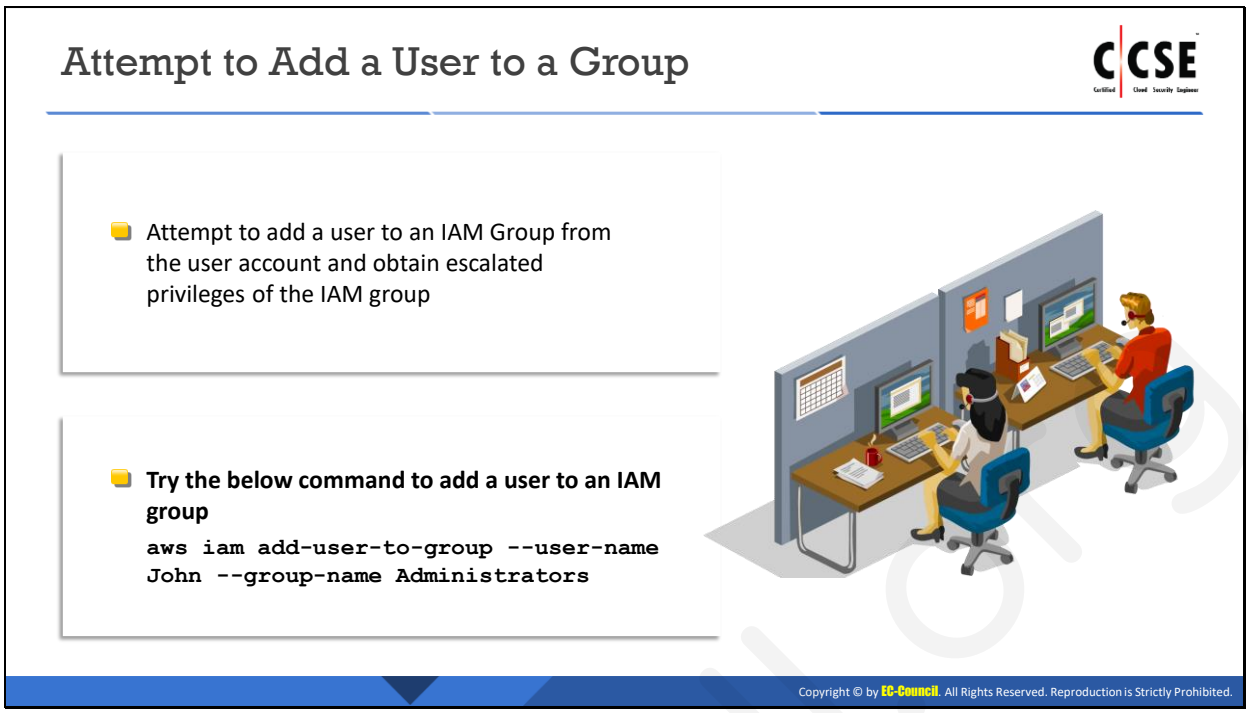
```
aws iam put-group-policy --group-name Administrator --policy-document
file://AdminPolicy.json --policy-name AdminRoot
```



Attempt to Create/Update an Inline Policy for a Role

Attempt to identify a policy that enables performing any action on a resource to escalate privileges. Try the below command to create/update an inline policy for an IAM role:

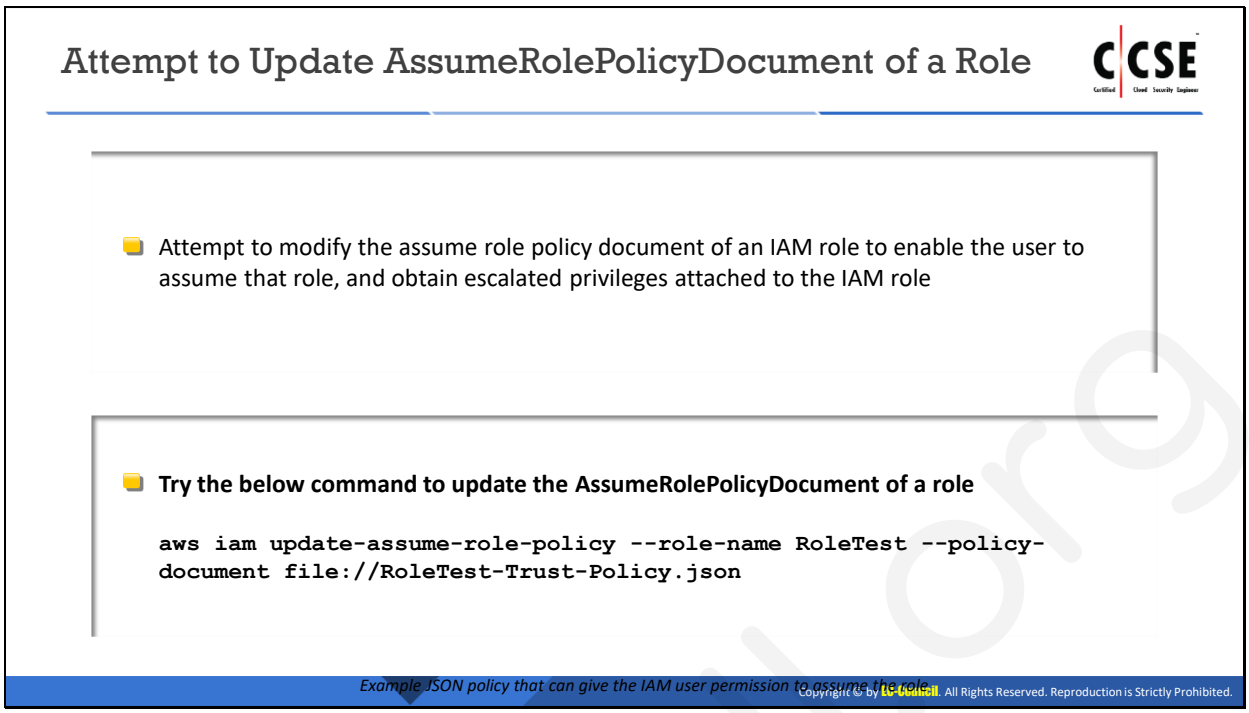
```
aws iam put-role-policy --role-name RoleTest --policy-name
ExamplePolicy --policy-document file://AdminPolicy.json
```



Attempt to Add a User to a Group

Attempt to add a user to an IAM Group from the user account and obtain escalated privileges of the IAM group. Try the below command to add a user to an IAM group:

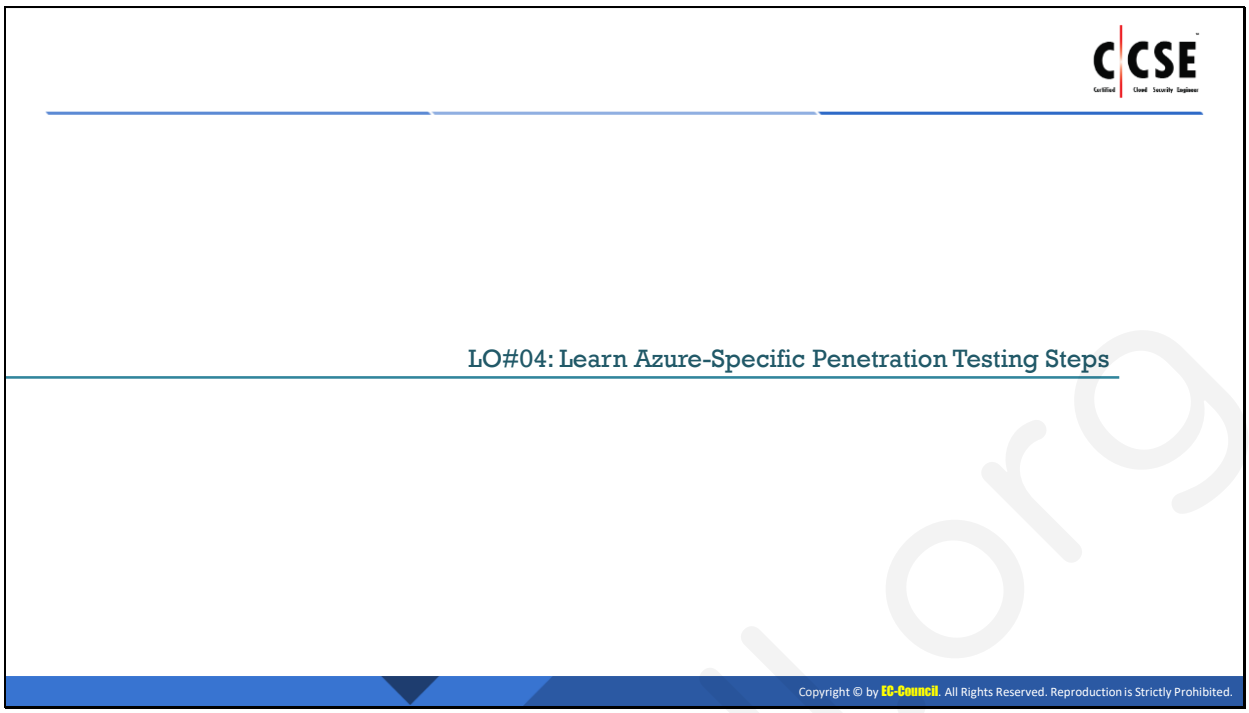
```
aws iam add-user-to-group --user-name John --group-name Administrators
```



Attempt to Update AssumeRolePolicyDocument of a Role

Attempt to modify the assume role policy document of an IAM role to enable the user to assume that role, and obtain escalated privileges attached to the IAM role. Try the below command to update the AssumeRolePolicyDocument of a role:

```
aws iam update-assume-role-policy --role-name RoleTest --policy-document file://RoleTest-Trust-Policy.json
```

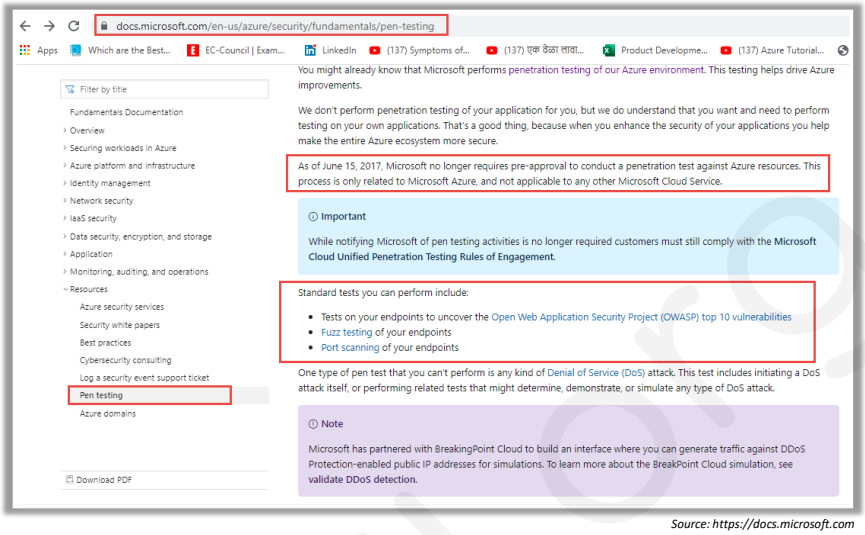


LO#04: Learn Azure-Specific Penetration Testing Steps

The objective of this section is to familiarize with and understand policies, permissions, procedures, terms, and conditions regarding Azure penetration testing.

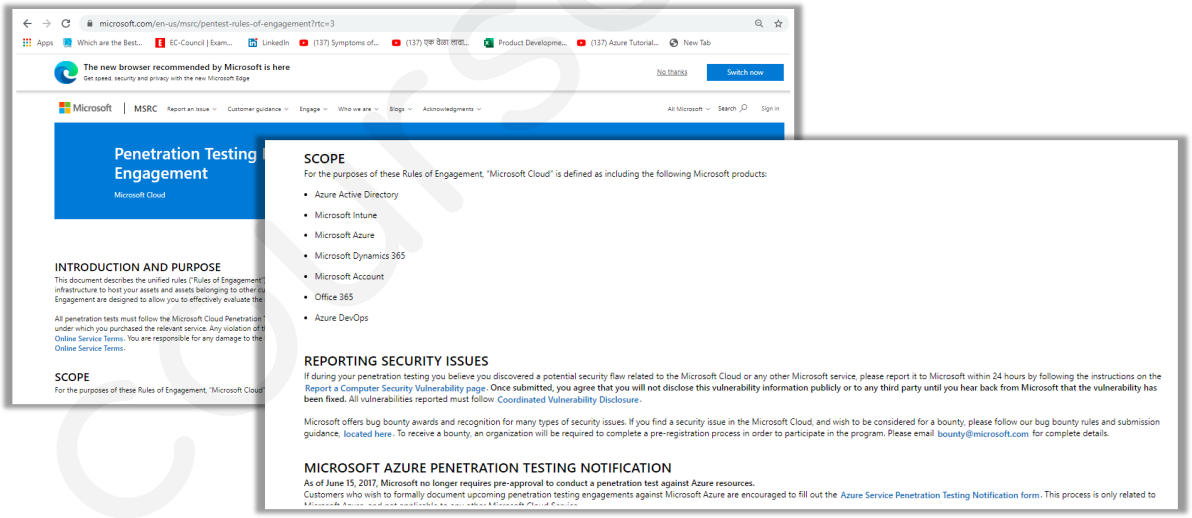
Understand Azure Penetration Testing Policy and Procedures

- Visit Microsoft Azure website to familiarize with and understand policies, permissions, procedures, rules of engagement, terms, and conditions regarding Azure penetration testing

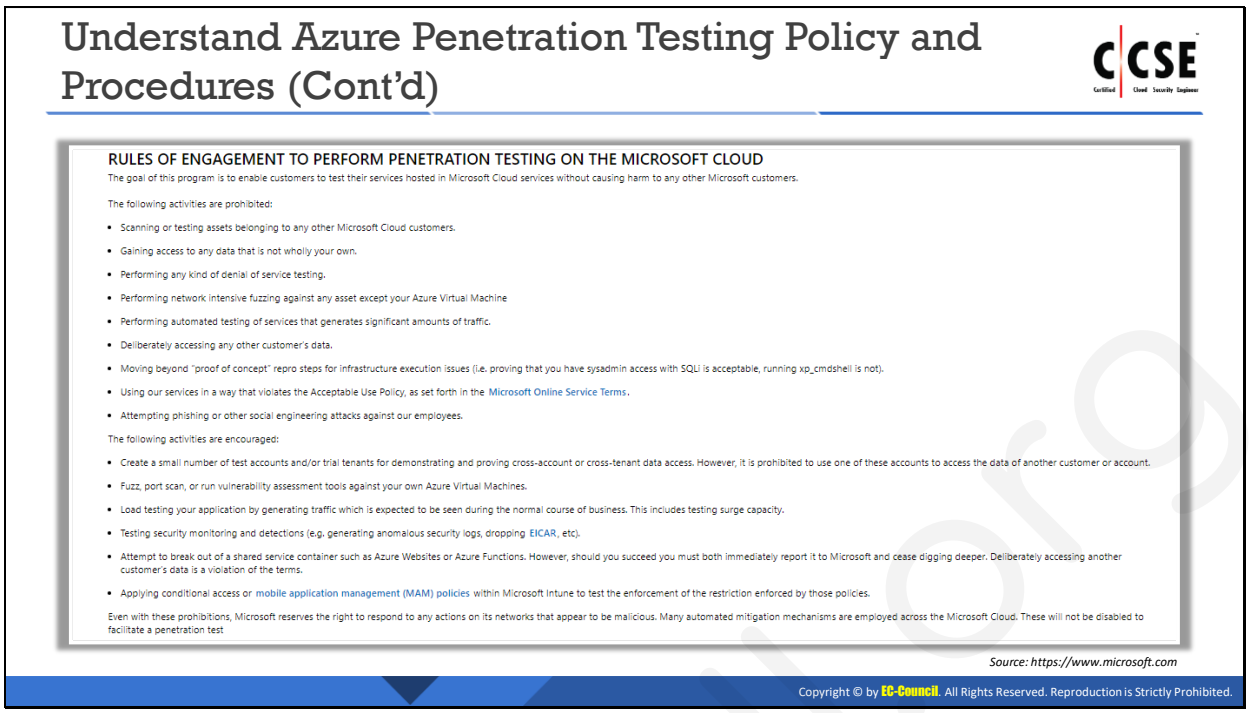


Source: https://docs.microsoft.com

Understand Azure Penetration Testing Policy and Procedures (Cont'd)



Source: https://www.microsoft.com



Understand Azure Penetration Testing Policy and Procedures

Visit Microsoft Azure website to familiarize with and understand policies, permissions, procedures, rules of engagement, terms, and conditions regarding Azure penetration testing

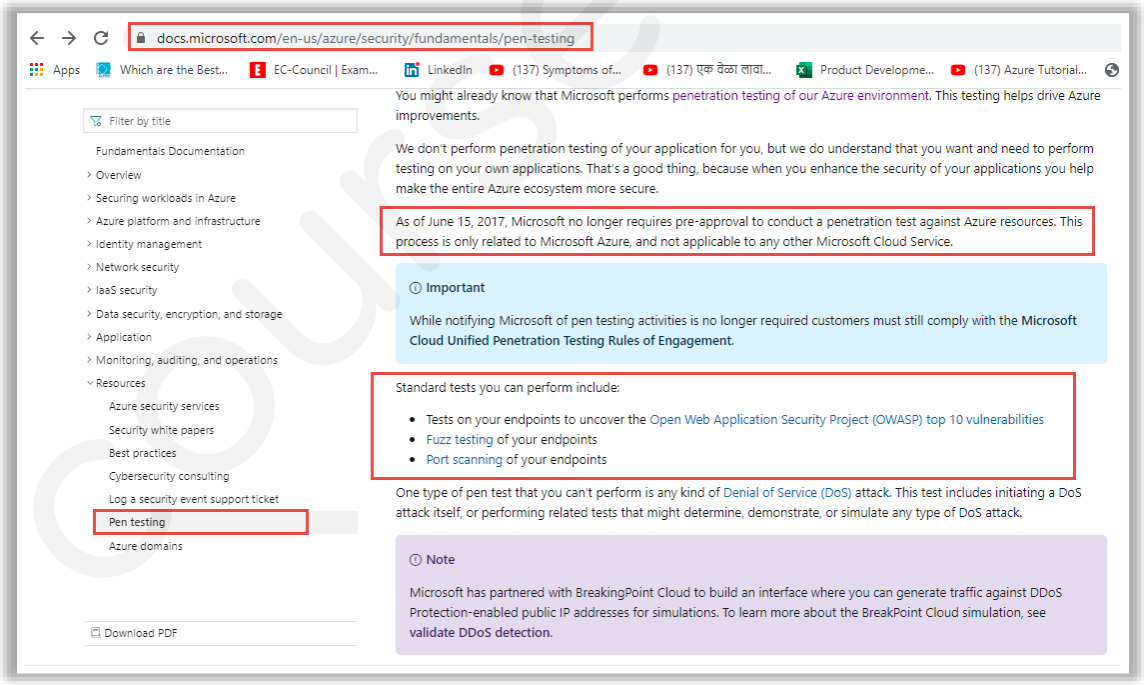


Figure 6.9: Screenshot Showing Guidelines for Azure Penetration Testing

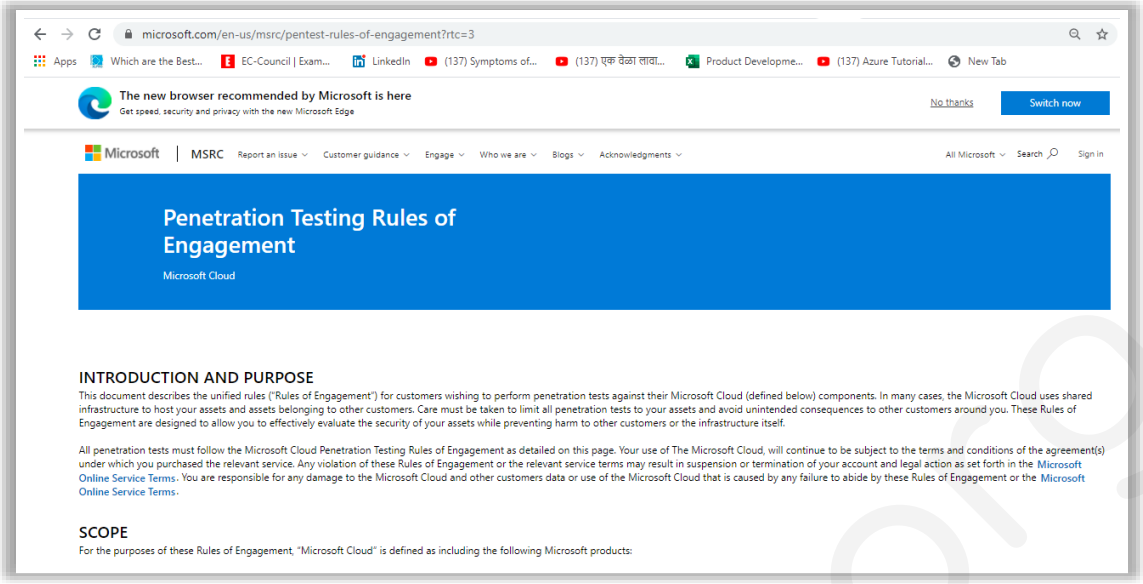


Figure 6.10: Screenshot Showing Azure's Penetration Testing Rules of Engagement

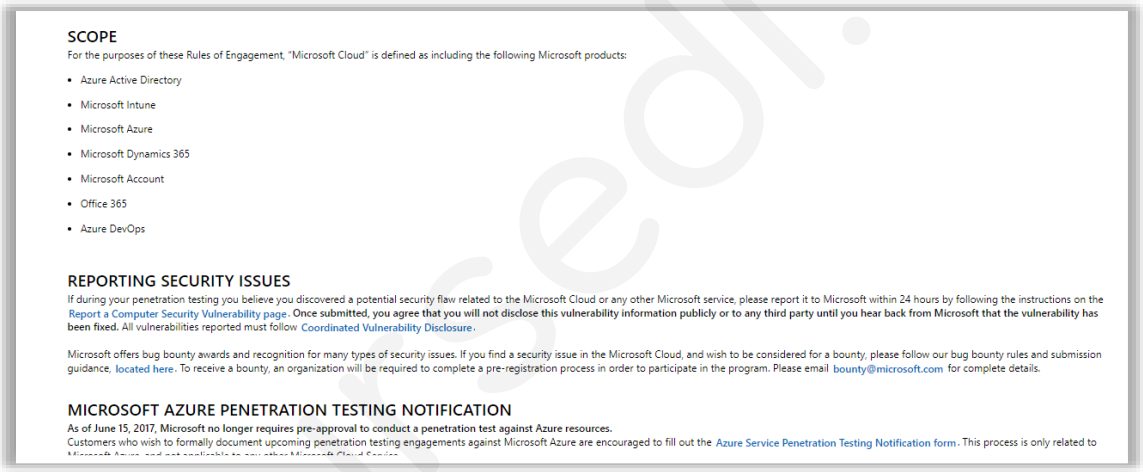
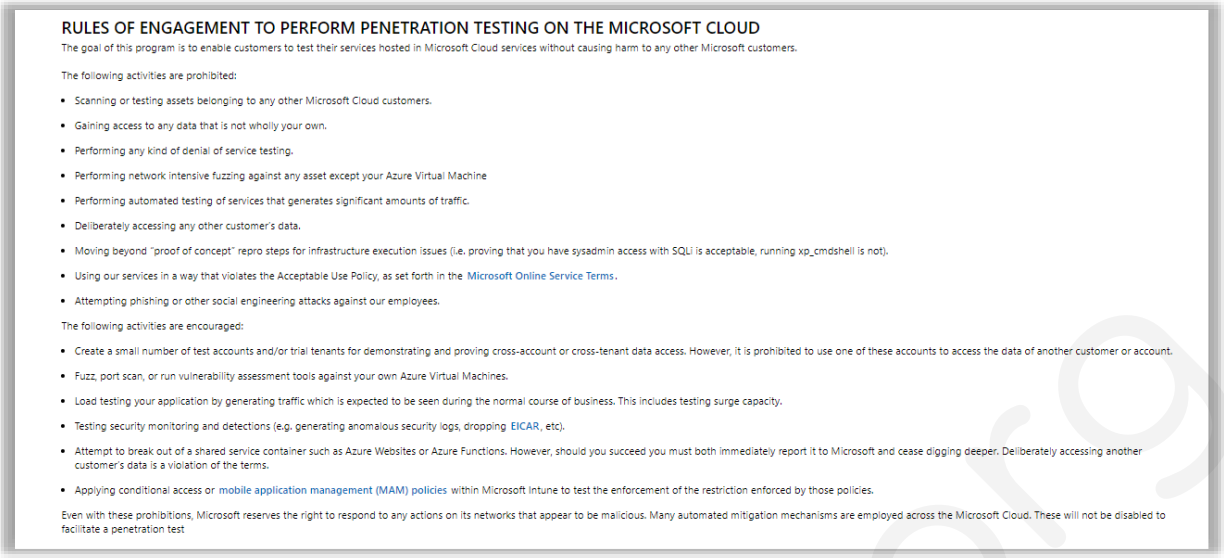


Figure 6.11: Screenshot Showing Azure's Penetration Testing Scope



RULES OF ENGAGEMENT TO PERFORM PENETRATION TESTING ON THE MICROSOFT CLOUD

The goal of this program is to enable customers to test their services hosted in Microsoft Cloud services without causing harm to any other Microsoft customers.

The following activities are prohibited:

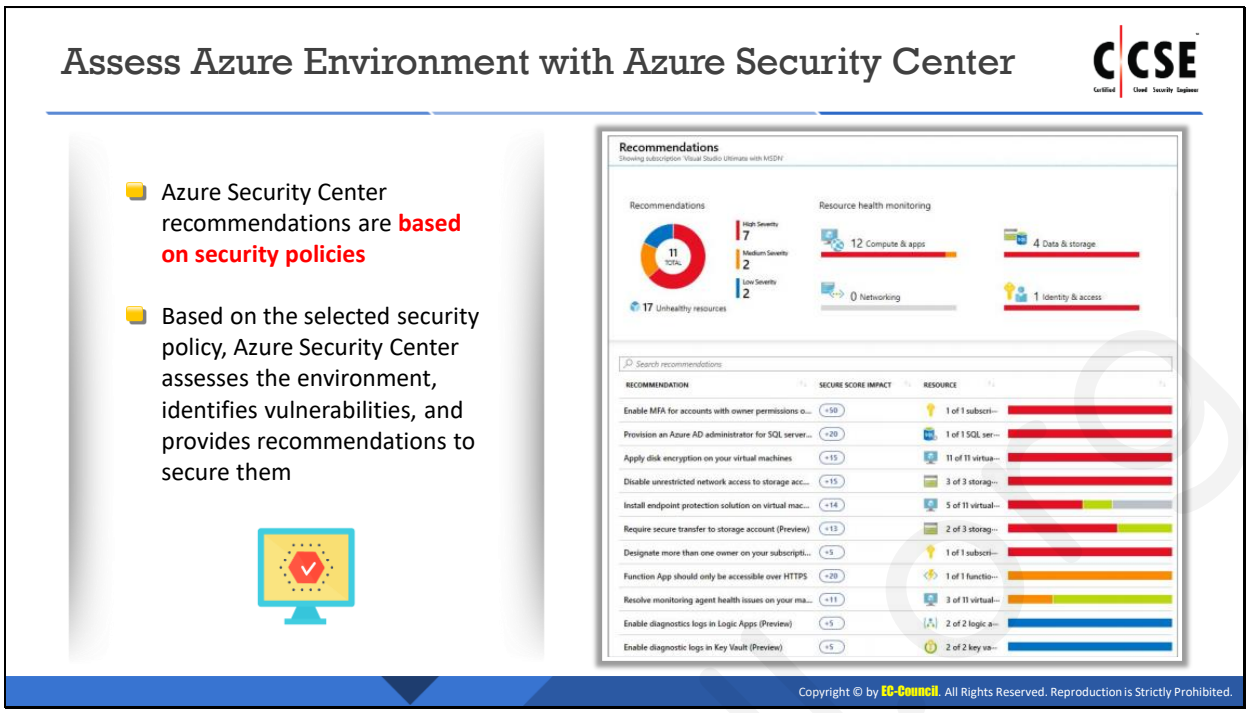
- Scanning or testing assets belonging to any other Microsoft Cloud customers.
- Gaining access to any data that is not wholly your own.
- Performing any kind of denial of service testing.
- Performing network intensive fuzzing against any asset except your Azure Virtual Machine
- Performing automated testing of services that generates significant amounts of traffic.
- Deliberately accessing any other customer's data.
- Moving beyond "proof of concept" repro steps for infrastructure execution issues (i.e. proving that you have sysadmin access with SQLi is acceptable, running xp_cmdshell is not).
- Using our services in a way that violates the Acceptable Use Policy, as set forth in the Microsoft Online Service Terms.
- Attempting phishing or other social engineering attacks against our employees.

The following activities are encouraged:

- Create a small number of test accounts and/or trial tenants for demonstrating and proving cross-account or cross-tenant data access. However, it is prohibited to use one of these accounts to access the data of another customer or account.
- Fuzz, port scan, or run vulnerability assessment tools against your own Azure Virtual Machines.
- Load testing your application by generating traffic which is expected to be seen during the normal course of business. This includes testing surge capacity.
- Testing security monitoring and detections (e.g. generating anomalous security logs, dropping EICAR, etc).
- Attempt to break out of a shared service container such as Azure Websites or Azure Functions. However, should you succeed you must both immediately report it to Microsoft and cease digging deeper. Deliberately accessing another customer's data is a violation of the terms.
- Applying conditional access or mobile application management (MAM) policies within Microsoft Intune to test the enforcement of the restriction enforced by those policies.

Even with these prohibitions, Microsoft reserves the right to respond to any actions on its networks that appear to be malicious. Many automated mitigation mechanisms are employed across the Microsoft Cloud. These will not be disabled to facilitate a penetration test

Figure 6.12: Screenshot Showing Rules of Engagement to Perform Penetrations Testing on The Microsoft Cloud



Assess Azure Environment with Azure Security Center

Azure Security Center recommendations are based on security policies. Based on the selected security policy, Azure Security Center assesses the environment, identifies vulnerabilities, and provides recommendations to secure them.

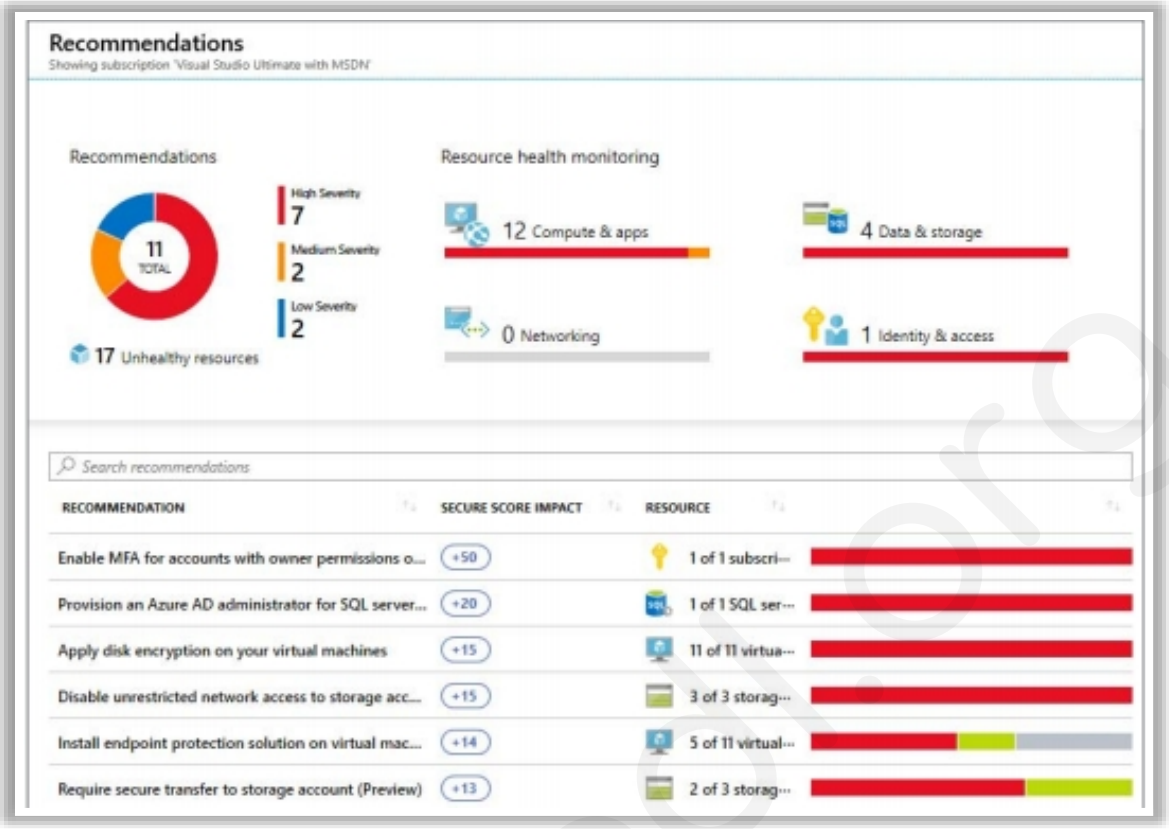
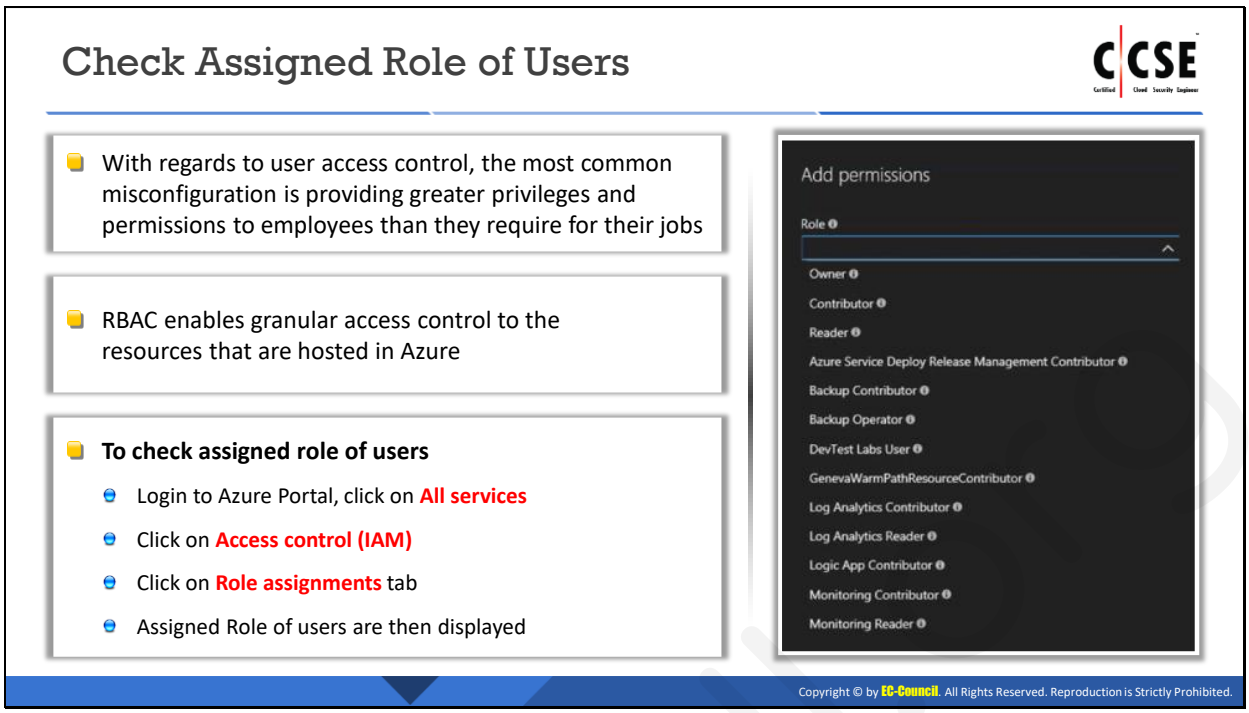
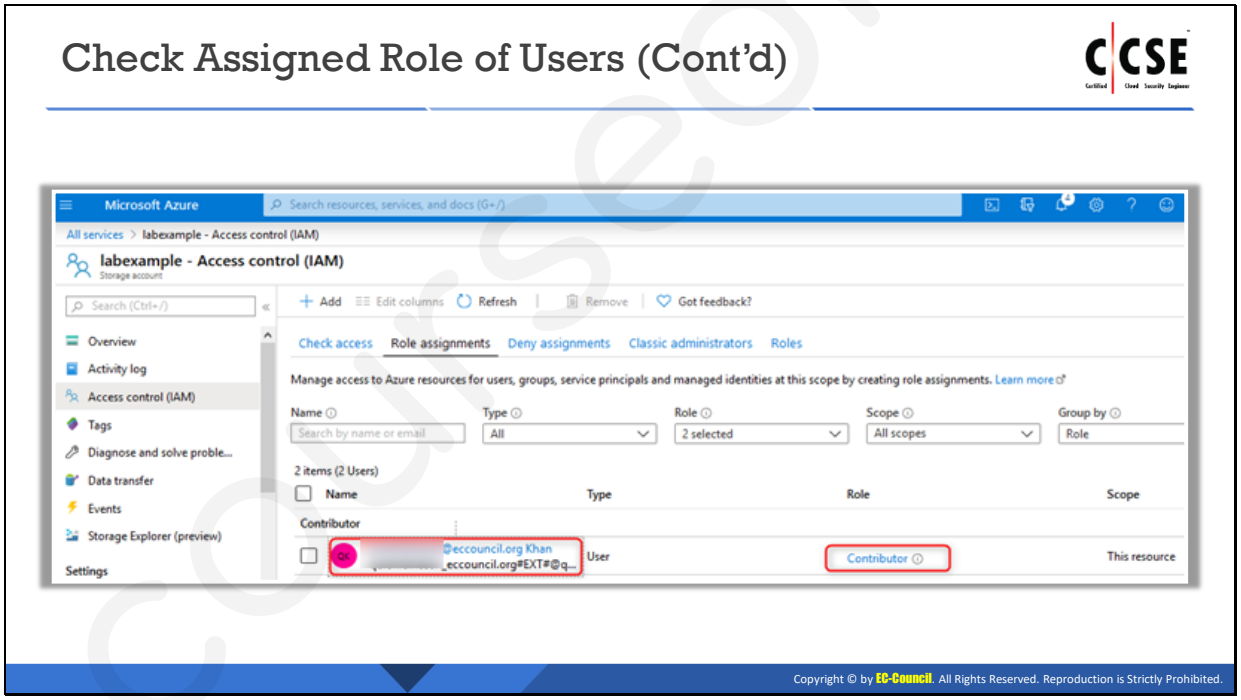


Figure 6.13: Screenshot of Azure Security Center





Check Assigned Role of Users

With regards to user access control, the most common misconfiguration is providing greater privileges and permissions to employees than they require for their jobs. RBAC enables granular access control to the resources that are hosted in Azure. To check assigned role of users:

- Login to Azure Portal, click on All services
- Click on Access control (IAM)

- Click on Role assignments tab
- Assigned Role of users are then displayed

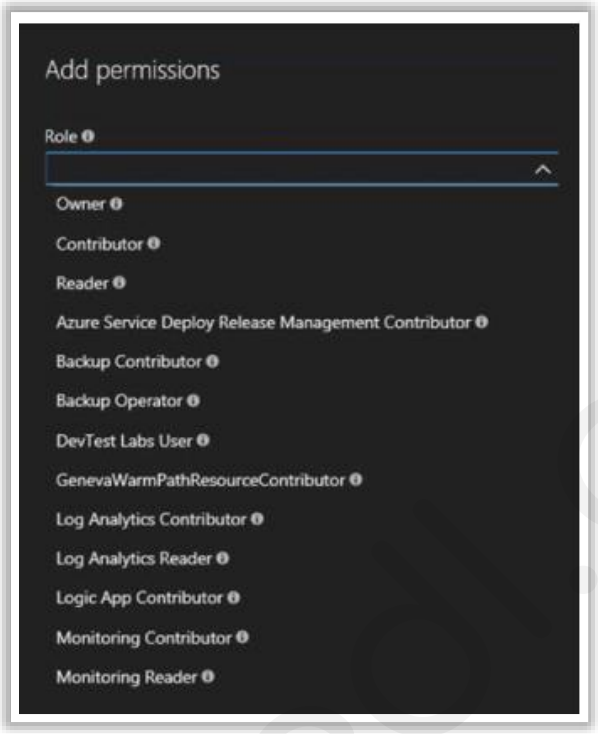


Figure 6.14: Screenshot Showing Azure Role of Users

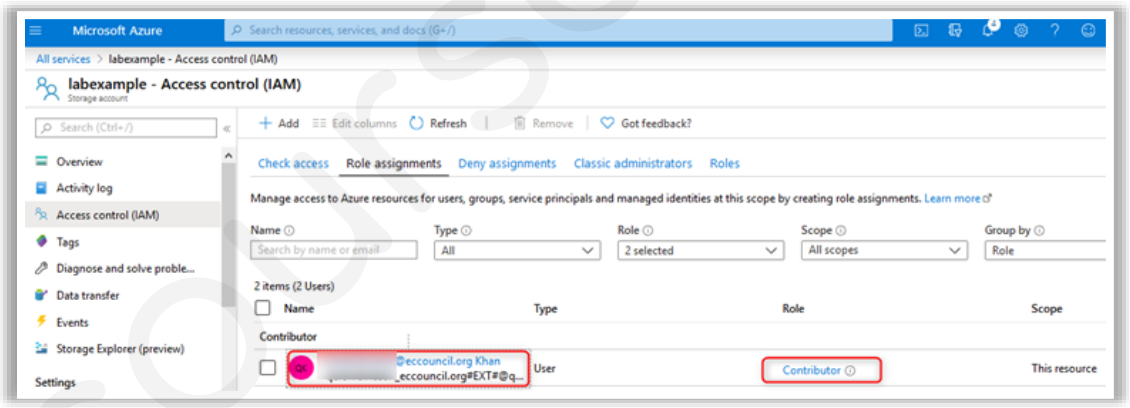
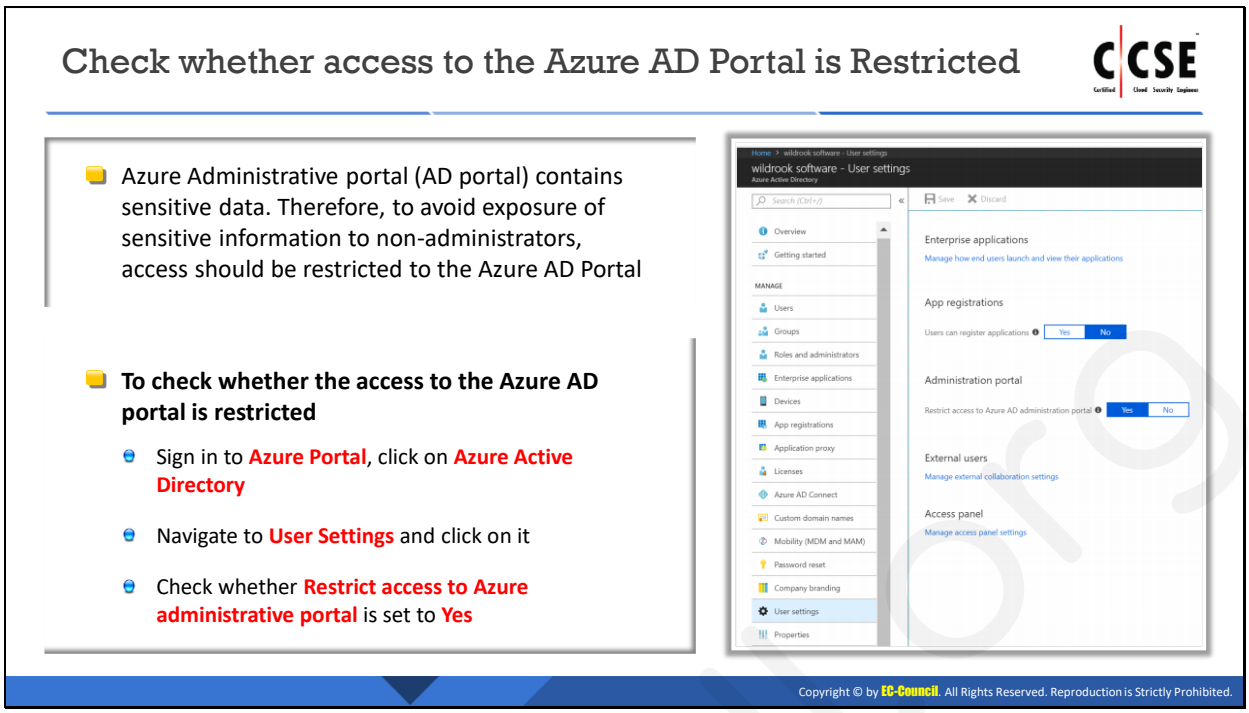


Figure 6.15: Screenshot Showing Azure User Access Control



Check Whether Access to the Azure AD Portal is Restricted

Azure Administrative portal (AD portal) contains sensitive data. Therefore, to avoid exposure of sensitive information to non-administrators, access should be restricted to the Azure AD Portal. To check whether the access to the Azure AD portal is restricted:

- Sign in to Azure Portal, click on Azure Active Directory
- Navigate to User Settings and click on it
- Check whether Restrict access to Azure administrative portal is set to Yes

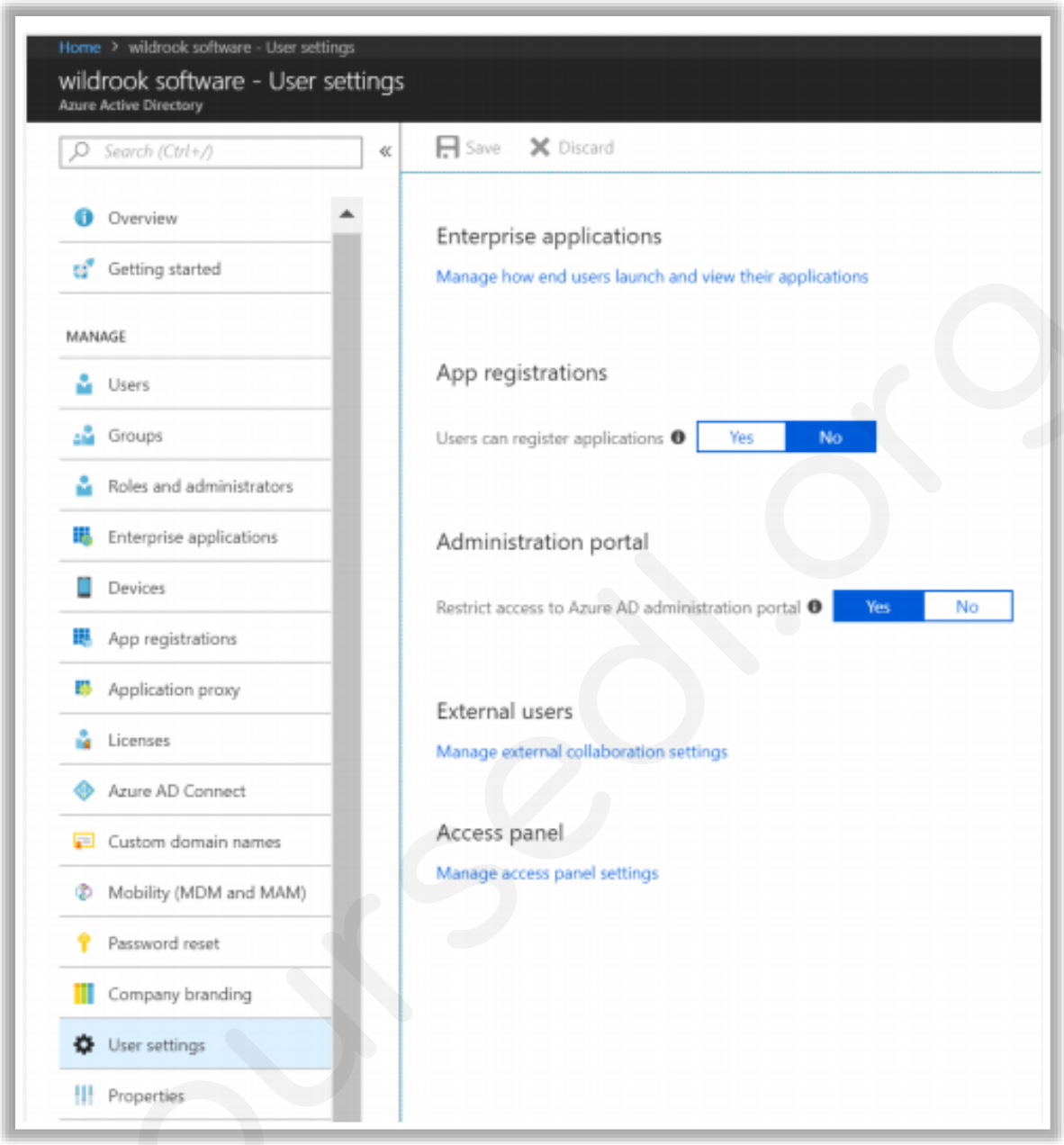
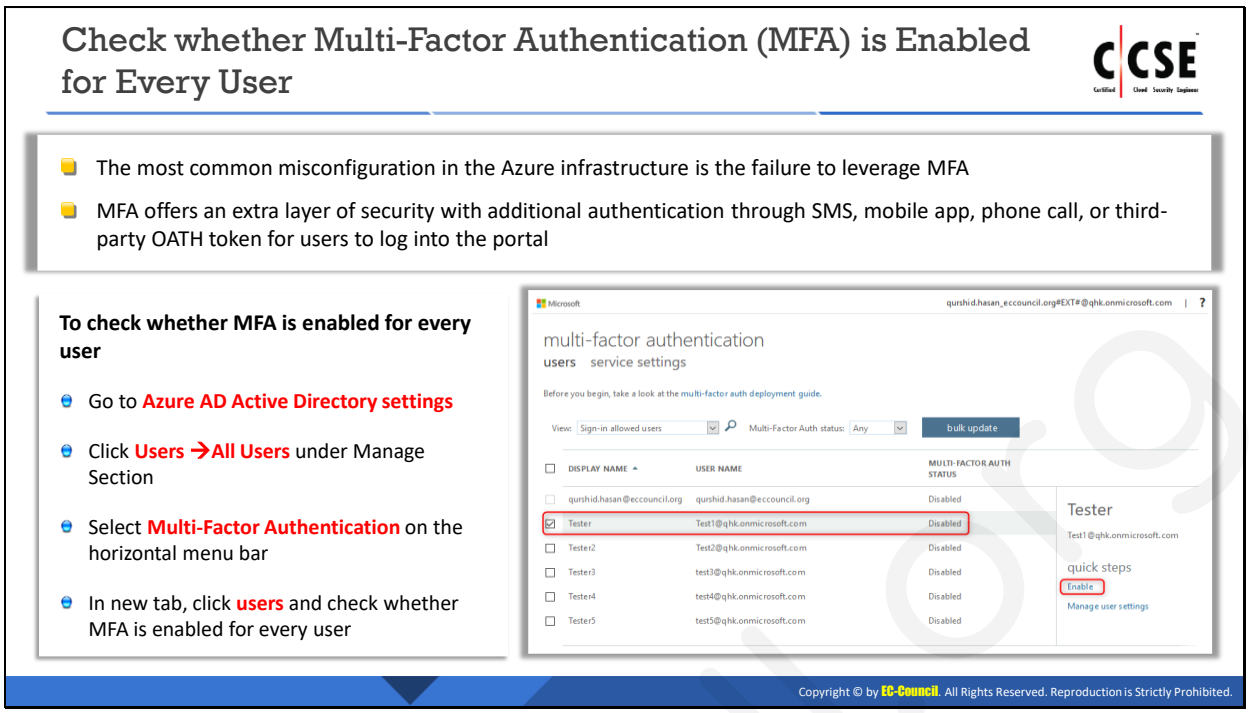


Figure 6.16: Screenshot Showing Azure AD User Settings



Check Whether Multi-Factor Authentication (MFA) is Enabled for Every User

The most common misconfiguration in the Azure infrastructure is the failure to leverage MFA. MFA offers an extra layer of security with additional authentication through SMS, mobile app, phone call, or third-party OATH token for users to log into the portal. To check whether MFA is enabled for every user:

- Go to Azure AD Active Directory settings
- Click Users àAll Users under Manage Section
- Select Multi-Factor Authentication on the horizontal menu bar
- In new tab, click users and check whether MFA is enabled for every user

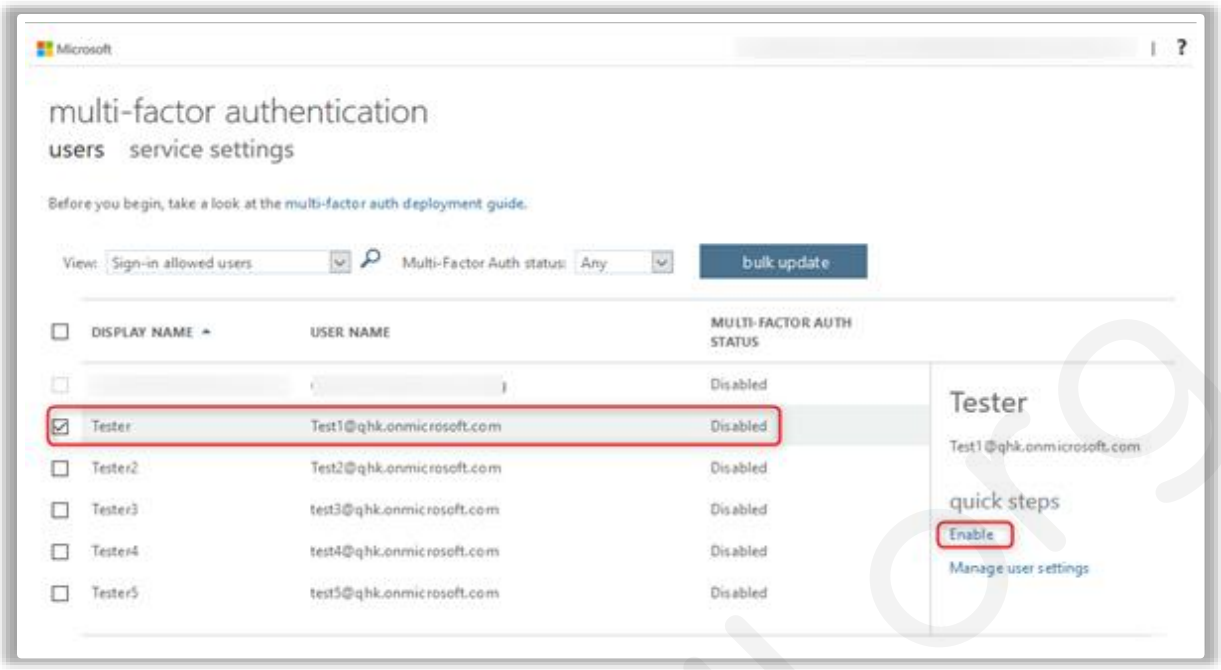
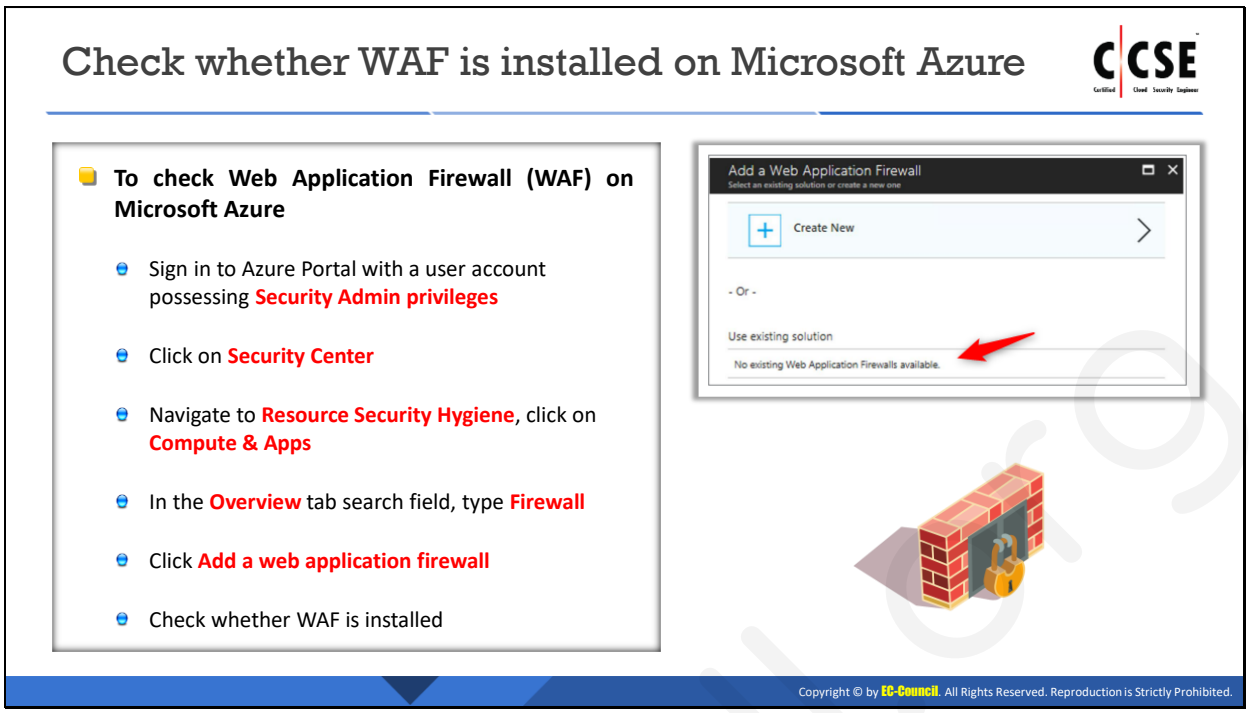


Figure 6.17: Screenshot Showing Azure Multi-Factor Authentication Settings



Check Whether WAF is Installed on Microsoft Azure

To check Web Application Firewall (WAF) on Microsoft Azure:

- Sign in to Azure Portal with a user account possessing Security Admin privileges
- Click on Security Center
- Navigate to Resource Security Hygiene, click on Compute & Apps
- In the Overview tab search field, type Firewall
- Click Add a web application firewall
- Check whether WAF is installed

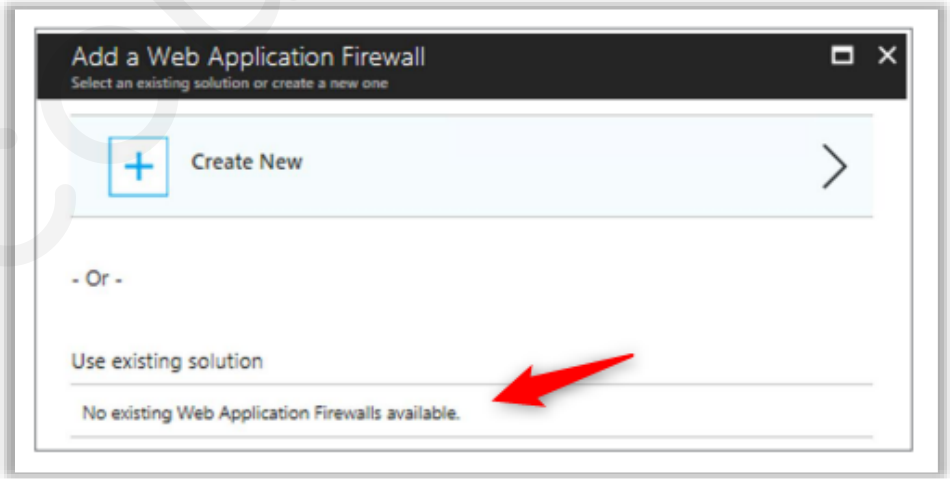
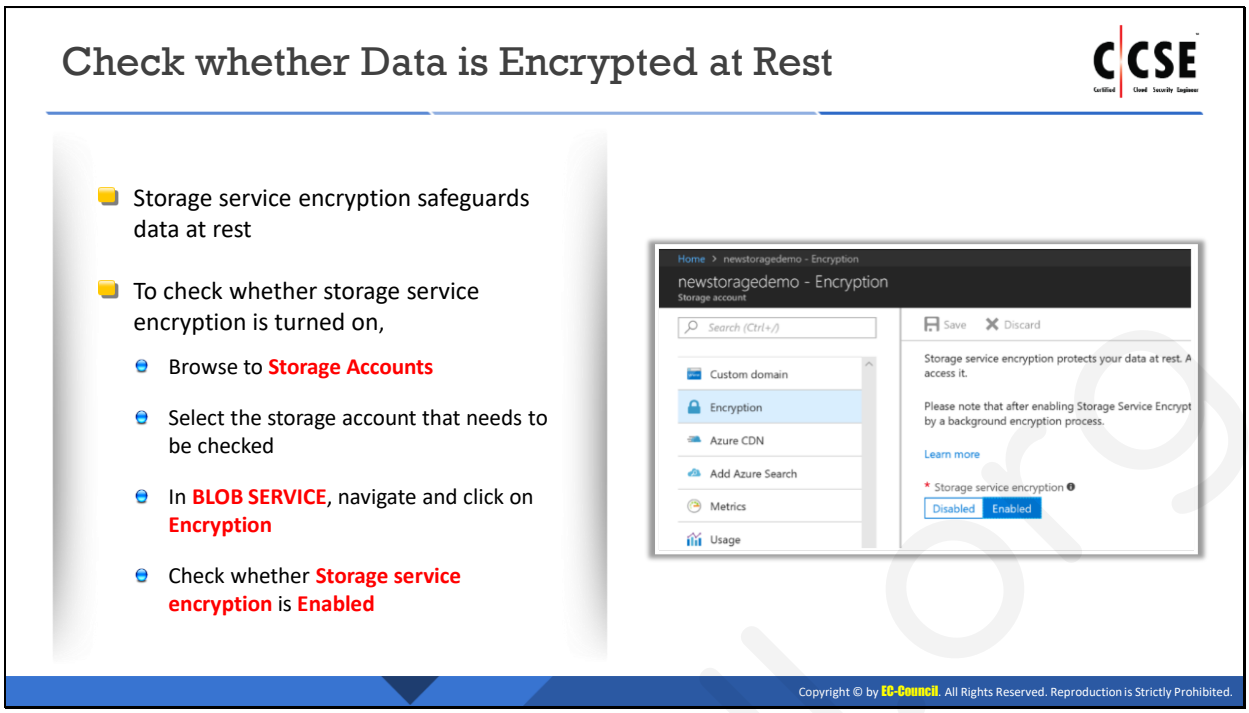


Figure 6.18: Screenshot Showing Azure WAF Settings



Check Whether Data is Encrypted at Rest

Storage service encryption safeguards data at rest. To check whether storage service encryption is turned on:

- Browse to Storage Accounts
- Select the storage account that needs to be checked
- In BLOB SERVICE, navigate and click on Encryption
- Check whether Storage service encryption is Enabled

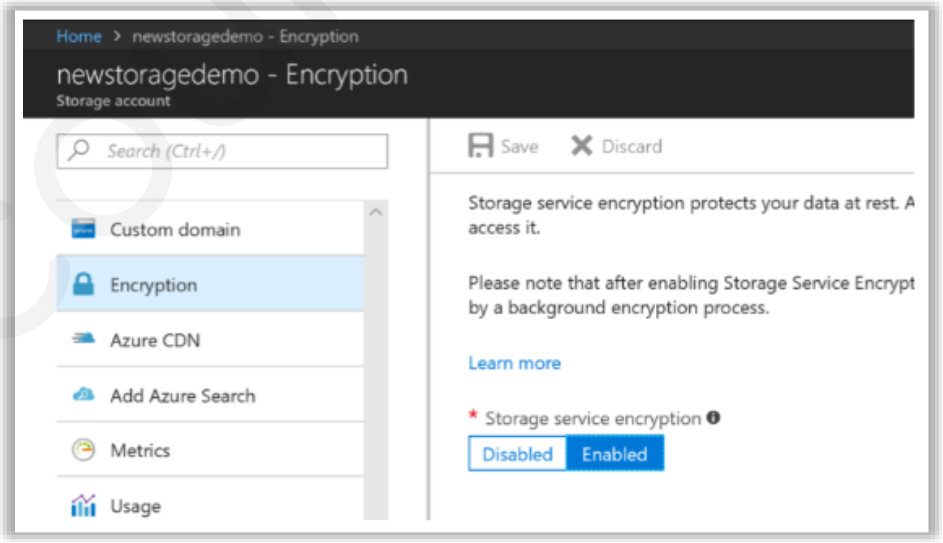
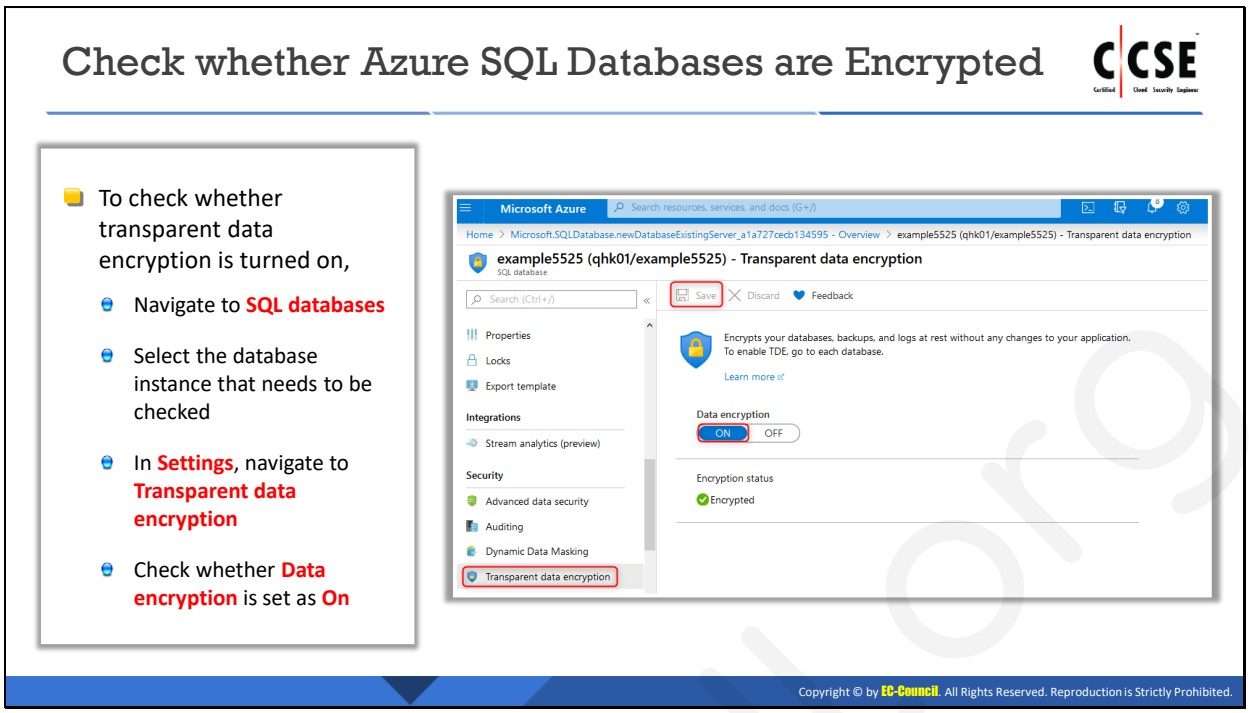


Figure 6.19: Screenshot Showing Storage Account Settings



Check Whether Azure SQL Databases are Encrypted

To check whether transparent data encryption is turned on:

- Navigate to SQL databases
- Select the database instance that needs to be checked
- In Settings, navigate to Transparent data encryption
- Check whether Data encryption is set as On

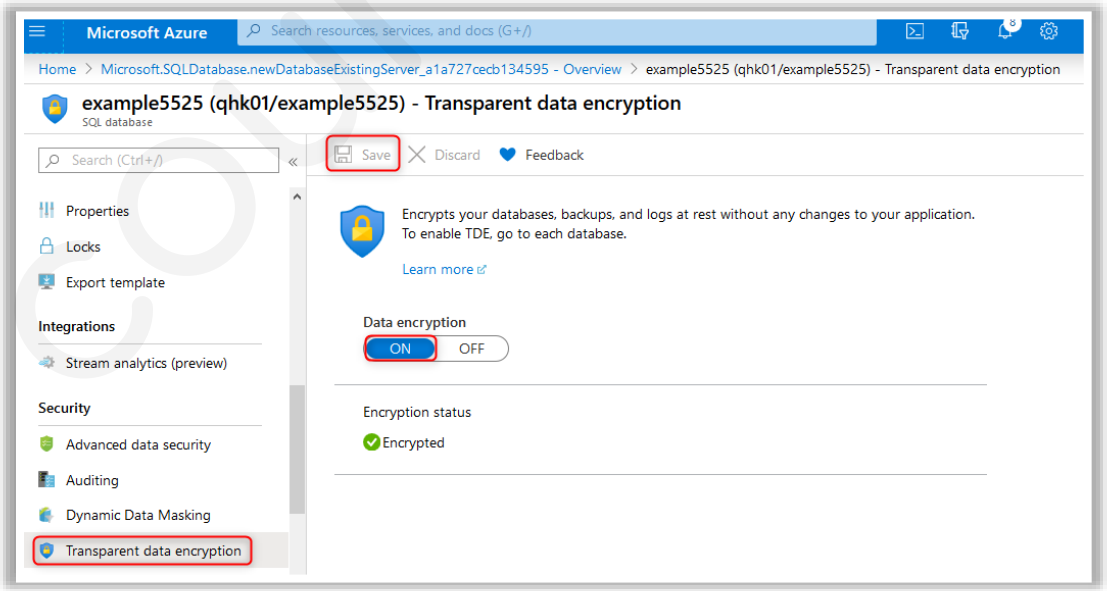
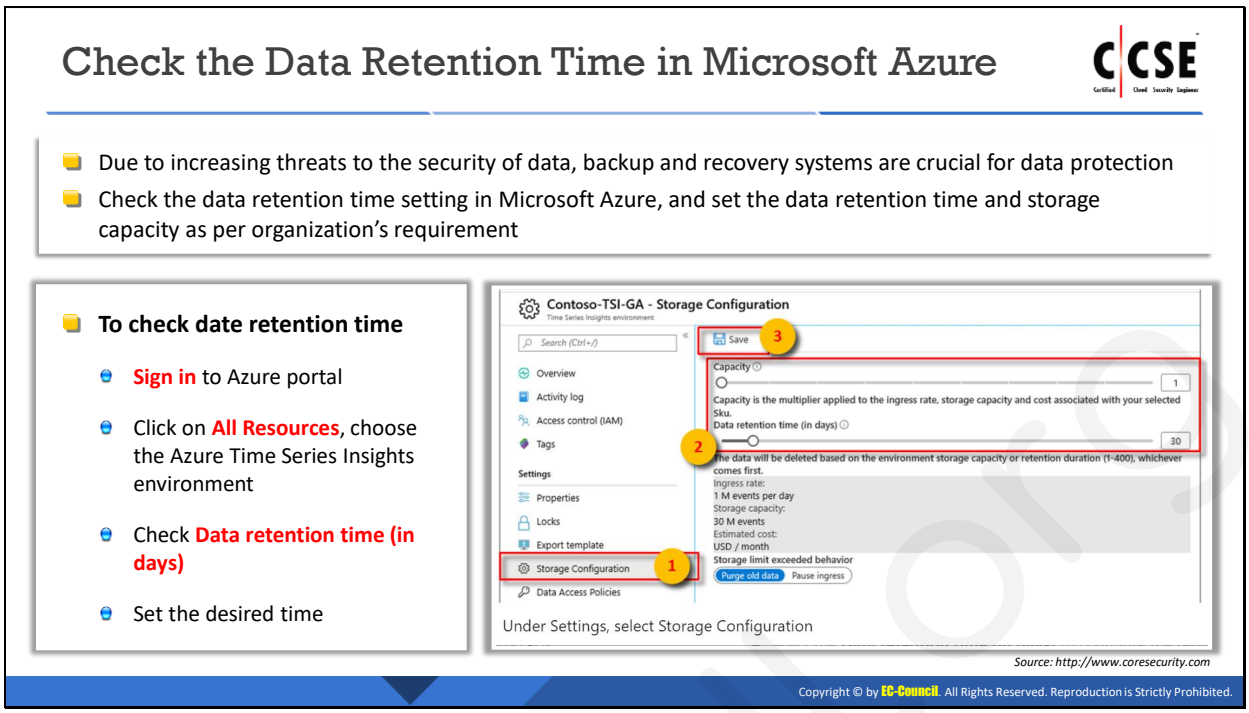


Figure 6.20: Screenshot Showing Data Encryption Settings



Check the Data Retention Time in Microsoft Azure

Due to increasing threats to the security of data, backup and recovery systems are crucial for data protection. Check the data retention time setting in Microsoft Azure, and set the data retention time and storage capacity as per organization's requirement. To check date retention time:

- Sign in to Azure portal
- Click on All Resources, choose the Azure Time Series Insights environment
- Check Data retention time (in days)
- Set the desired time

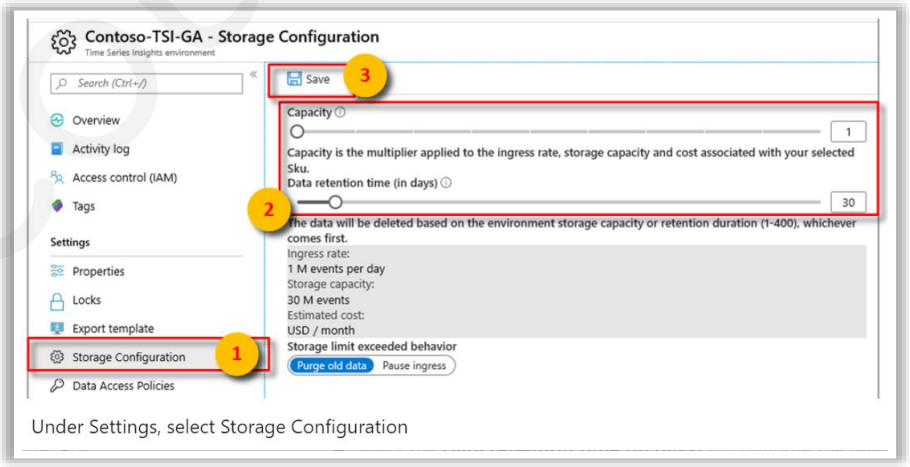
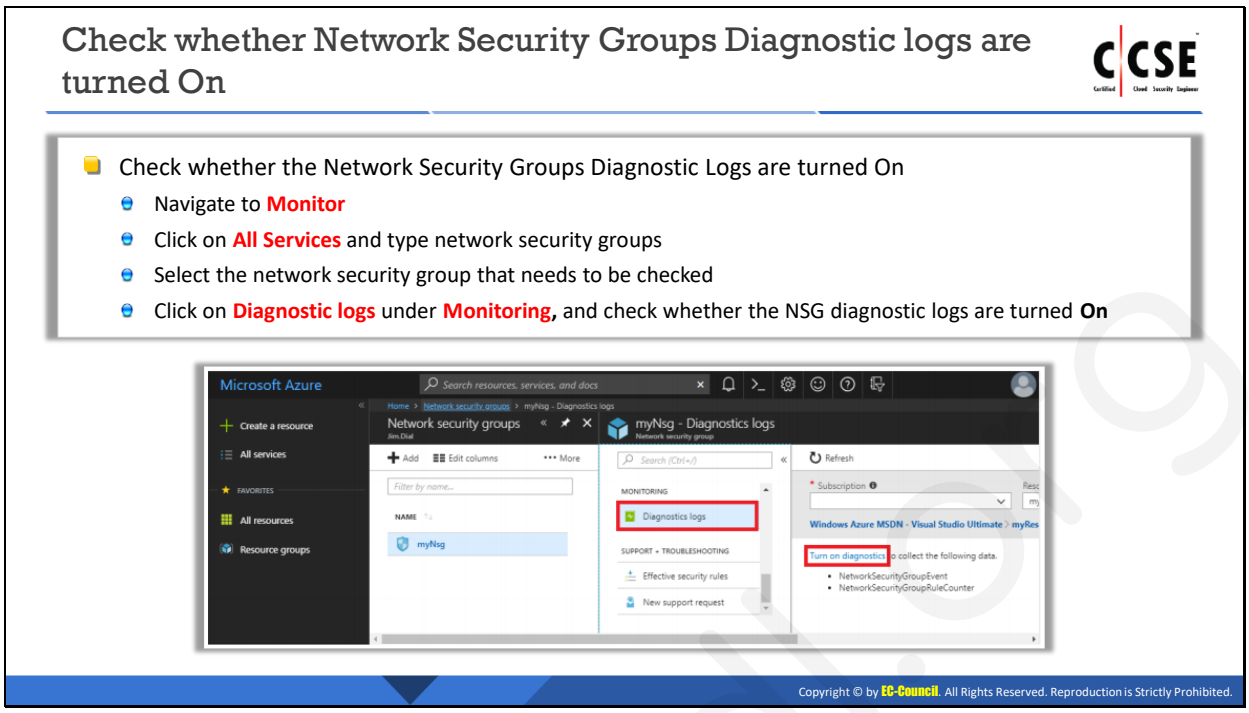


Figure 6.21: Screenshot Showing Storage Configuration Settings



Check Whether Network Security Groups Diagnostic Logs are Turned On

Check whether the Network Security Groups Diagnostic Logs are turned On:

- Navigate to Monitor
- Click on All Services and type network security groups
- Select the network security group that needs to be checked
- Click on Diagnostic logs under Monitoring, and check whether the NSG diagnostic logs are turned On

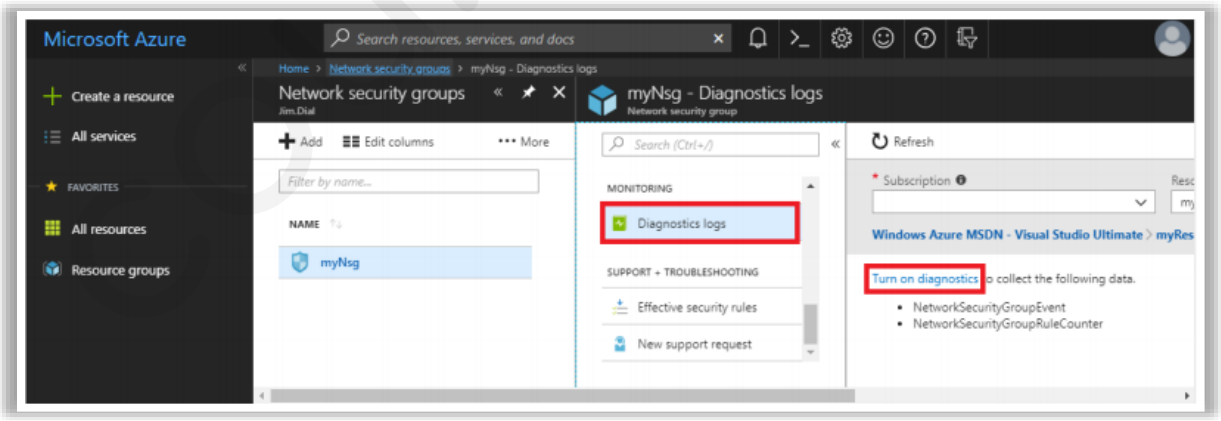
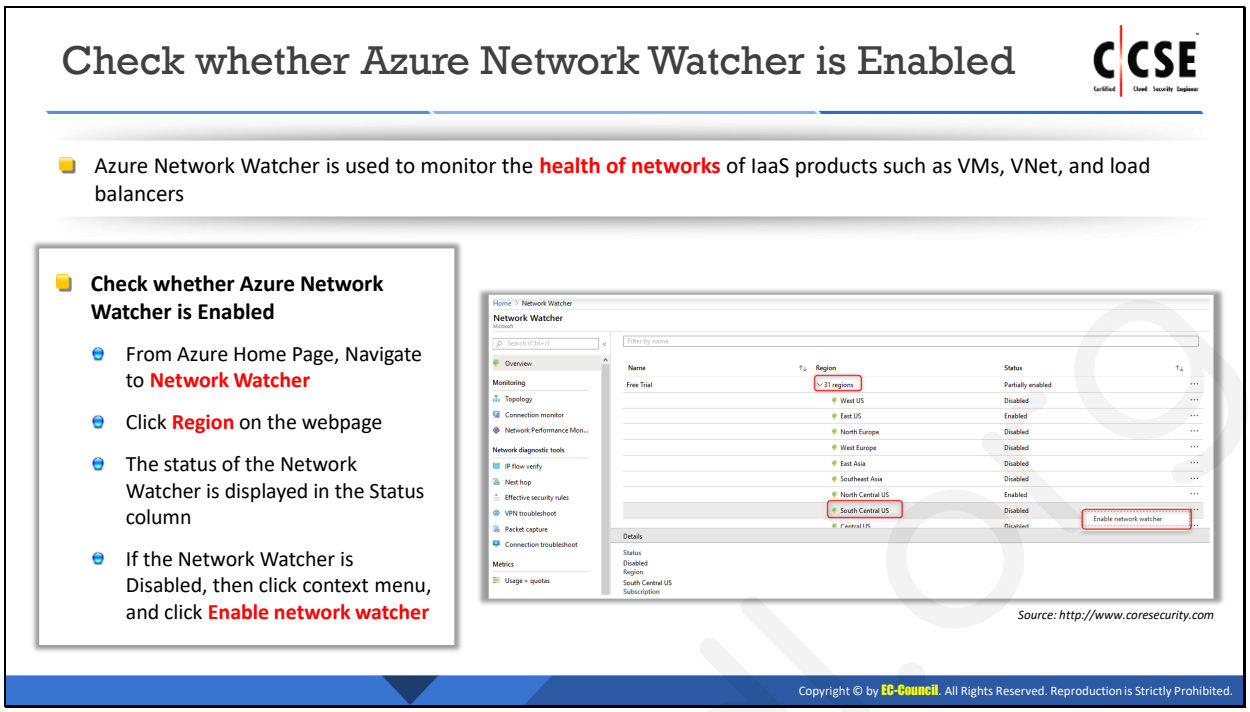


Figure 6.22: Screenshot Showing Network Security Groups



Check Whether Azure Network Watcher is Enabled

Azure Network Watcher is used to monitor the health of networks of IaaS products such as VMs, VNet, and load balancers. Check whether Azure Network Watcher is Enabled:

- From Azure Home Page, Navigate to Network Watcher
- Click Region on the webpage
- The status of the Network Watcher is displayed in the Status column
- If the Network Watcher is Disabled, then click context menu, and click Enable network watcher

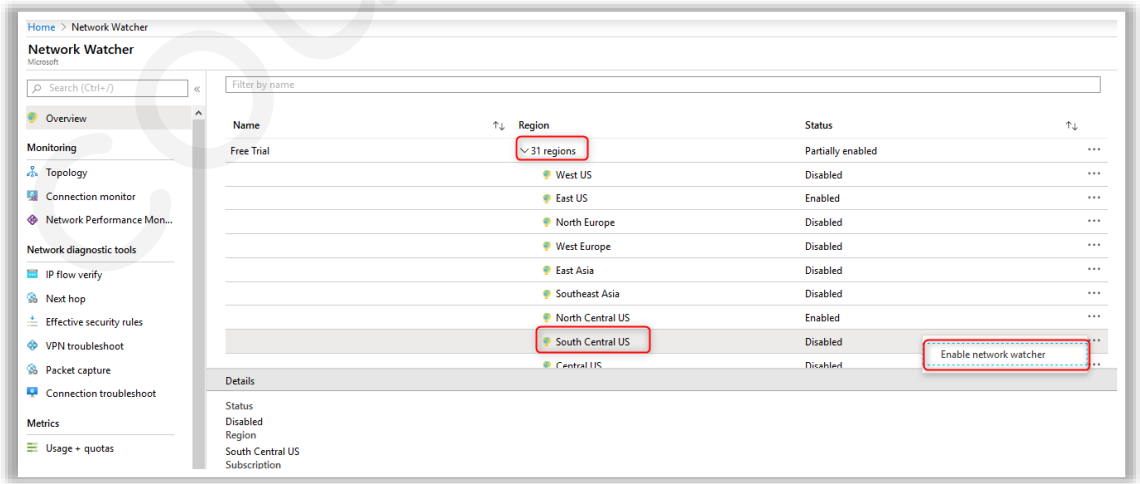
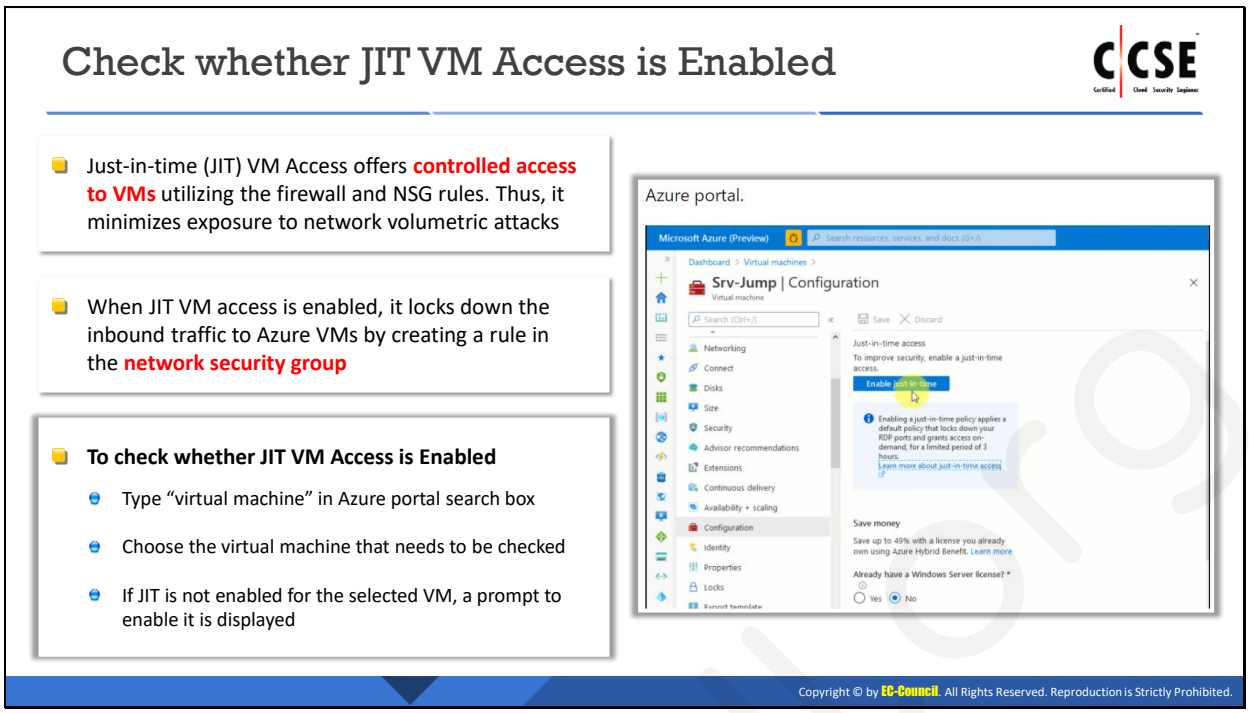


Figure 6.23: Screenshot of Network Watcher



Check Whether JIT VM Access is Enabled

Just-in-time (JIT) VM Access offers controlled access to VMs utilizing the firewall and NSG rules. Thus, it minimizes exposure to network volumetric attacks. When JIT VM access is enabled, it locks down the inbound traffic to Azure VMs by creating a rule in the network security group. To check whether JIT VM Access is Enabled:

- Type “virtual machine” in Azure portal search box
- Choose the virtual machine that needs to be checked
- If JIT is not enabled for the selected VM, a prompt to enable it is displayed

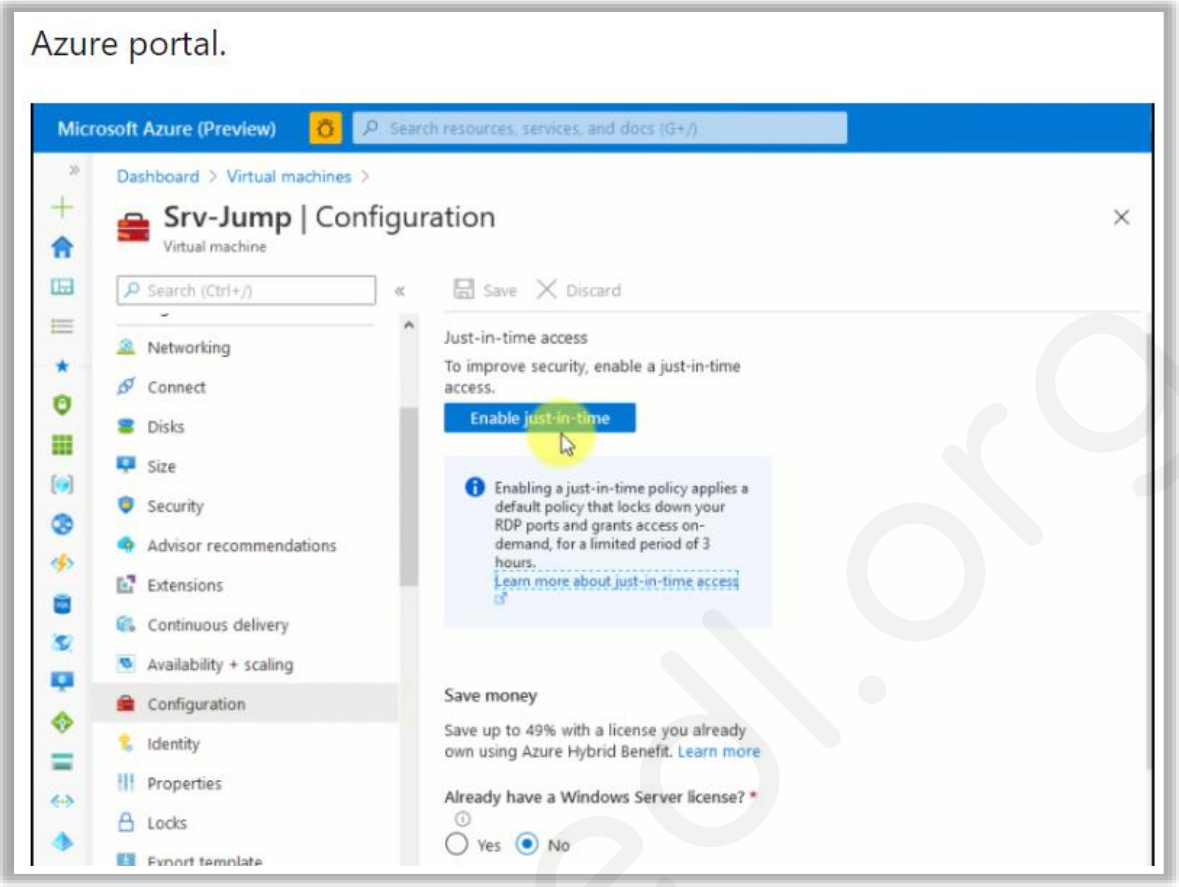
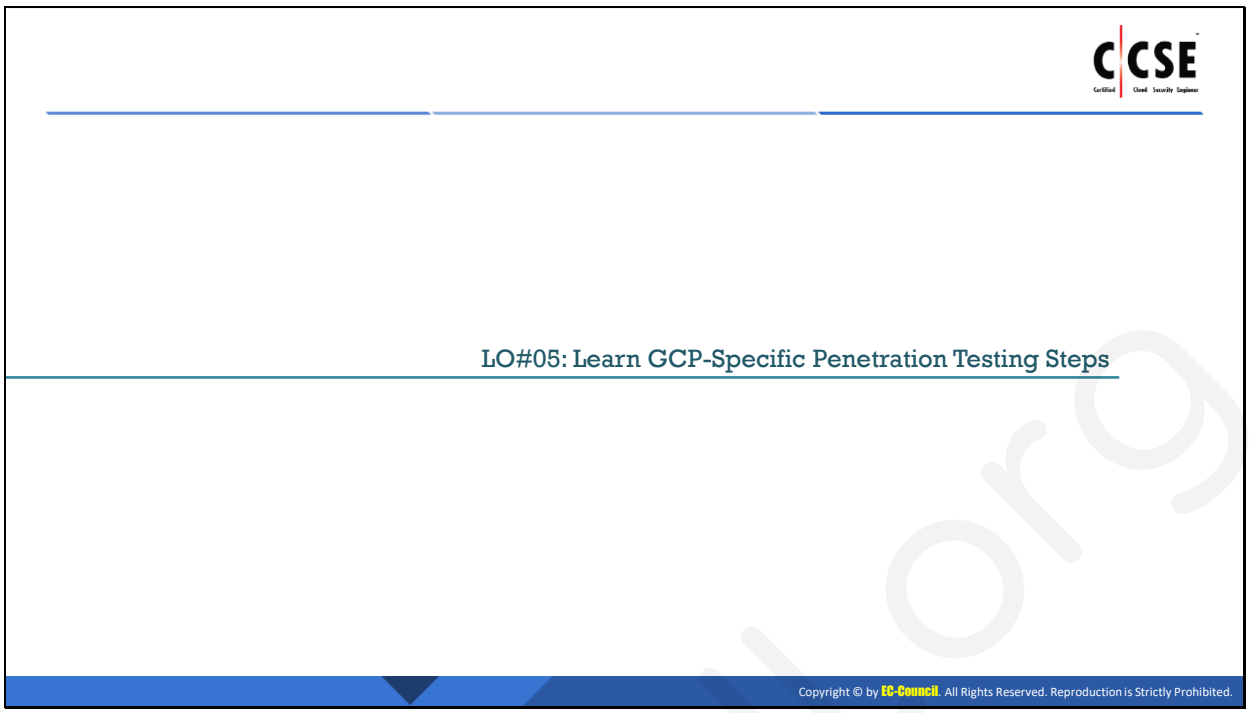
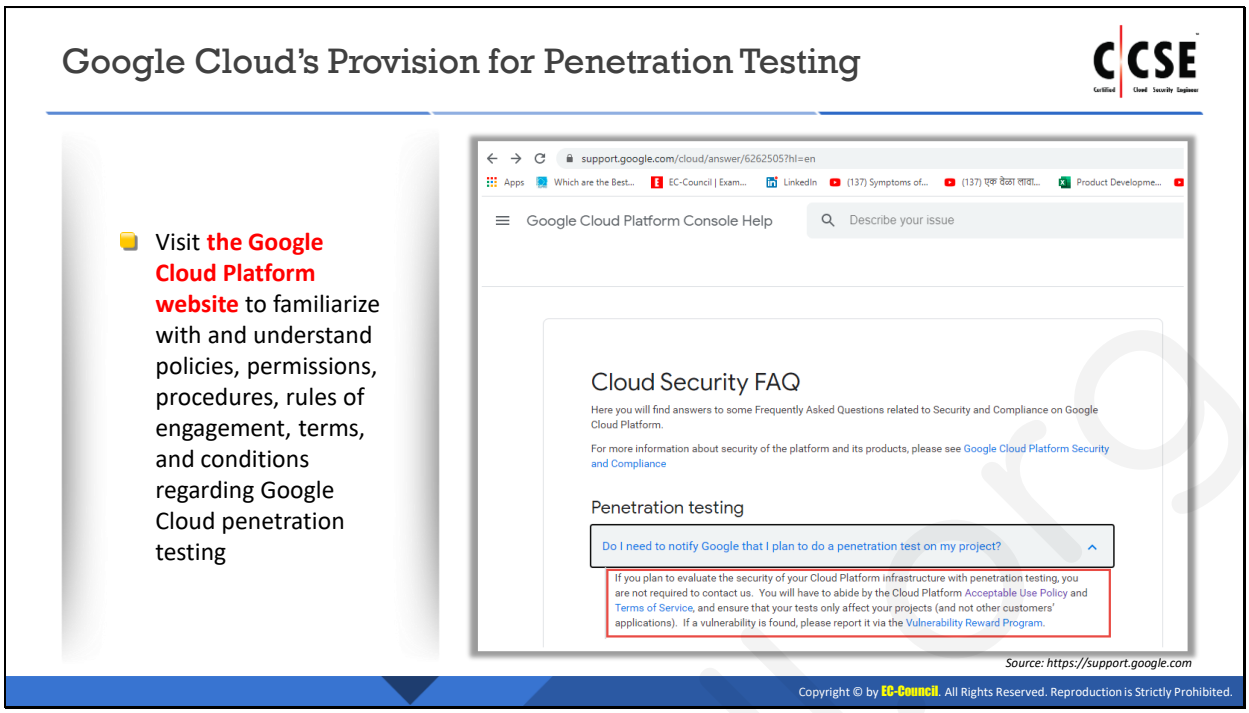


Figure 6.24: Screenshot of Azure Portal



LO#05: Learn GCP-Specific Penetration Testing Steps

The objective of this section is to familiarize with and understand policies, permissions, procedures, terms, and conditions regarding Google Cloud Platform (GCP) penetration testing.



Google Cloud's Provision for Penetration Testing

Visit the Google Cloud Platform website to familiarize with and understand policies, permissions, procedures, rules of engagement, terms, and conditions regarding Google Cloud penetration testing.

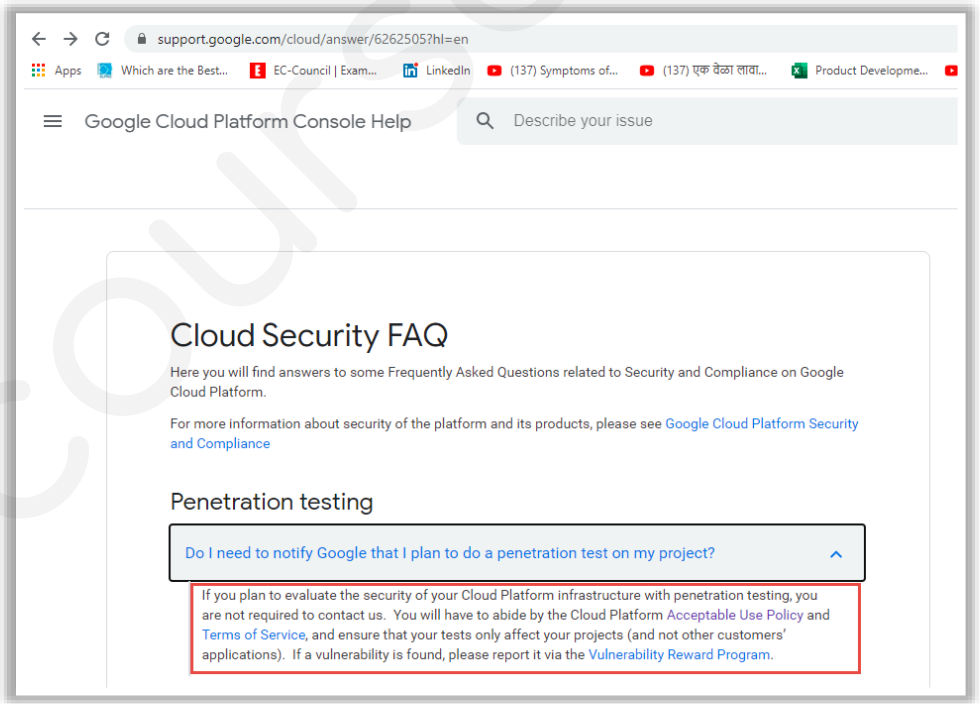
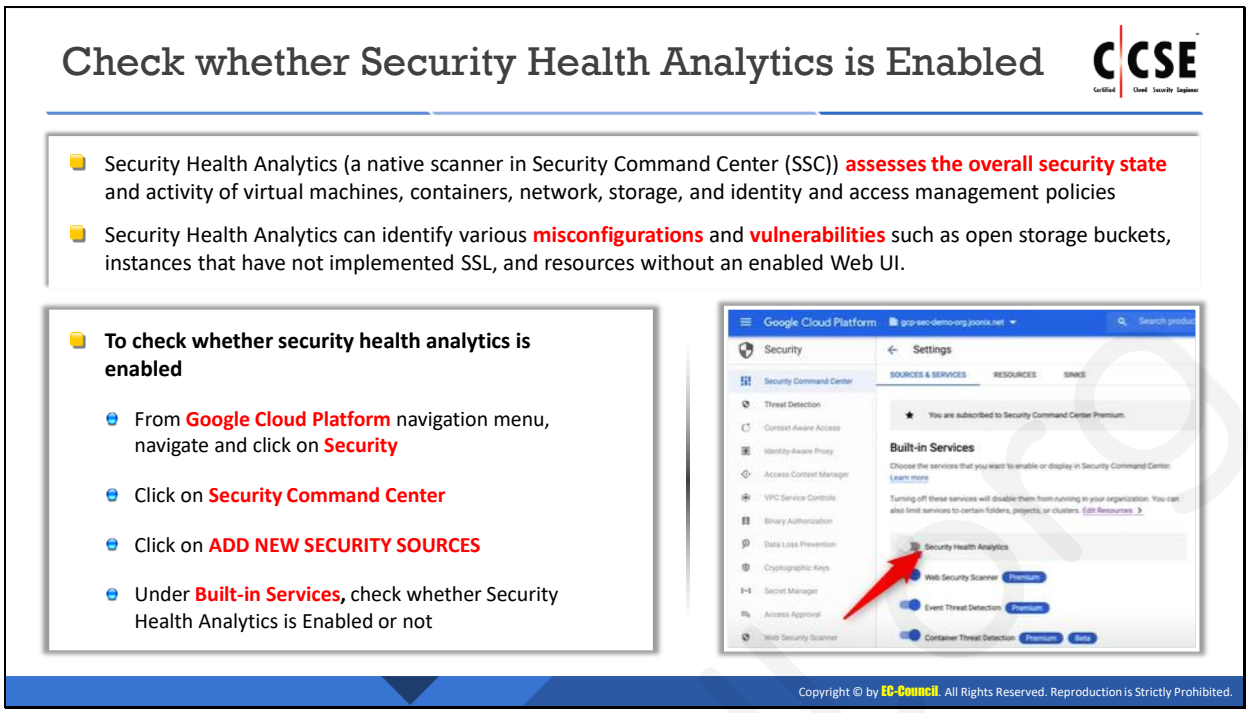


Figure 6.25: Screenshot Showing Google Cloud Platform Console Help



Check Whether Security Health Analytics is Enabled

Security Health Analytics (a native scanner in Security Command Center (SSC)) assesses the overall security state and activity of virtual machines, containers, network, storage, and identity and access management policies. Security Health Analytics can identify various misconfigurations and vulnerabilities such as open storage buckets, instances that have not implemented SSL, and resources without an enabled Web UI.

To check whether security health analytics is enabled:

- From Google Cloud Platform navigation menu, navigate and click on Security
- Click on Security Command Center
- Click on ADD NEW SECURITY SOURCES
- Under Built-in Services, check whether Security Health Analytics is Enabled or not

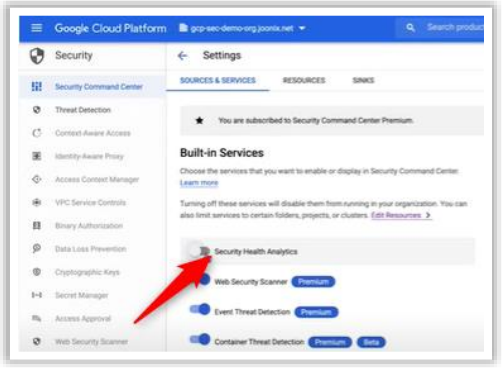
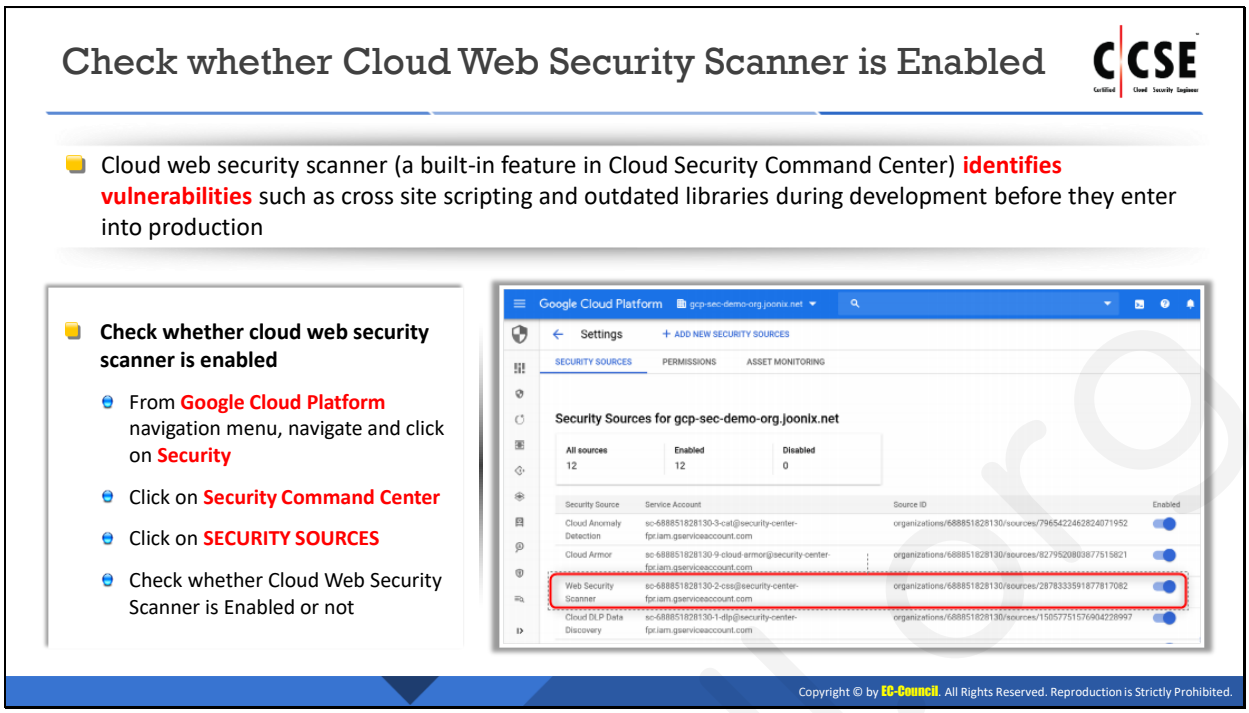


Figure 6.26: Screenshot Showing Google Cloud Platform Settings



Check Whether Cloud Web Security Scanner is Enabled

Cloud web security scanner (a built-in feature in Cloud Security Command Center) identifies vulnerabilities such as cross site scripting and outdated libraries during development before they enter into production. Check whether cloud web security scanner is enabled:

From Google Cloud Platform navigation menu, navigate and click on Security

- Click on Security Command Center
- Click on SECURITY SOURCES
- Check whether Cloud Web Security Scanner is Enabled or not

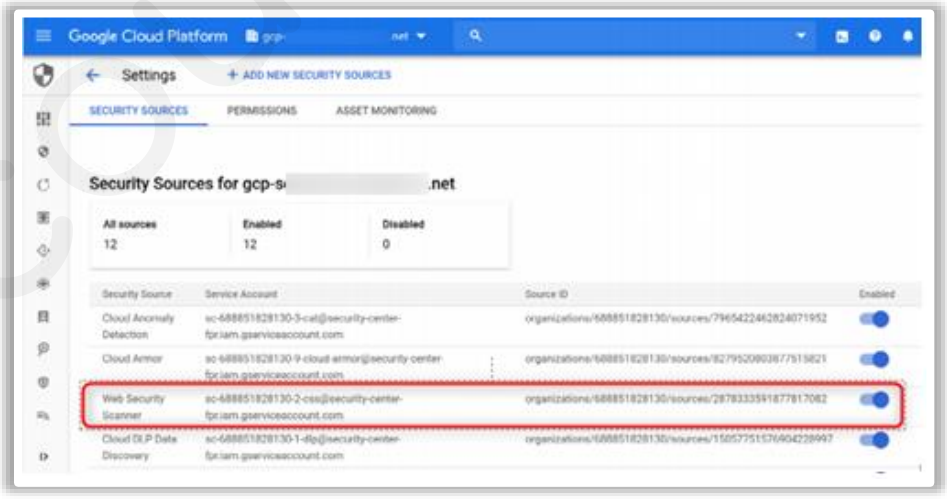
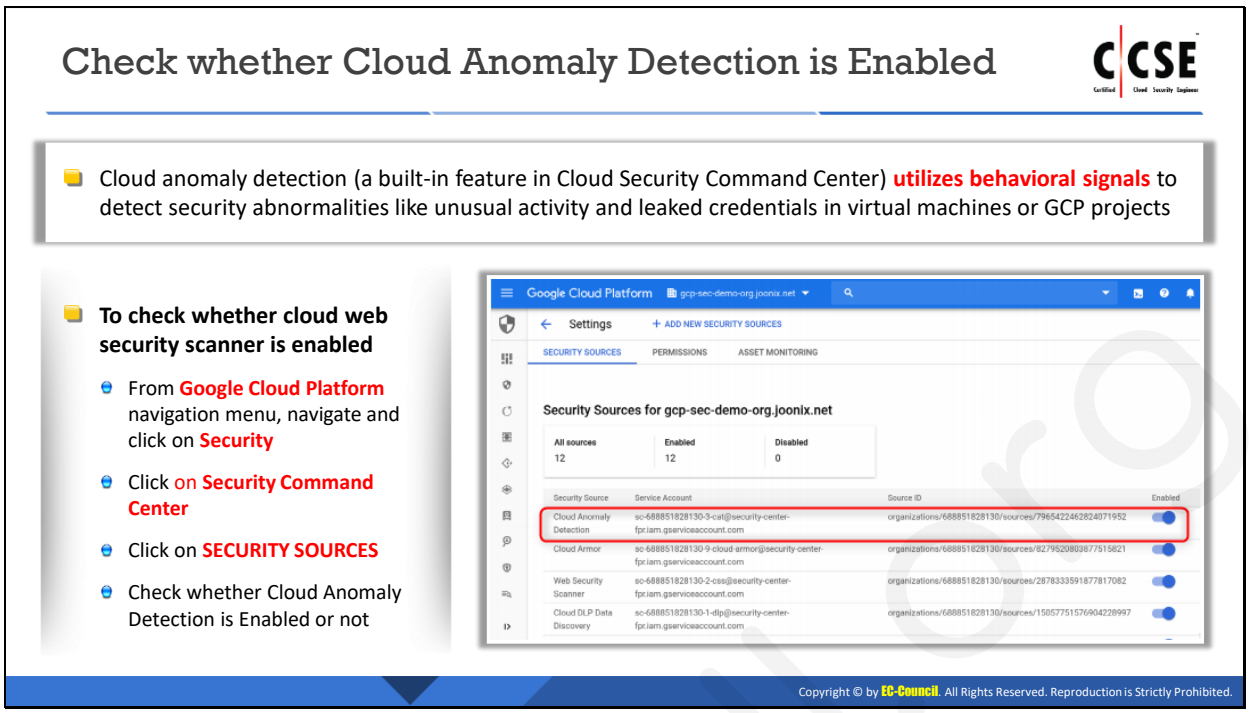


Figure 6.27: Screenshot Showing Google Cloud Platform Settings



Check Whether Cloud Anomaly Detection is Enabled

Cloud anomaly detection (a built-in feature in Cloud Security Command Center) utilizes behavioral signals to detect security abnormalities like unusual activity and leaked credentials in virtual machines or GCP projects. To check whether cloud web security scanner is enabled:

- From Google Cloud Platform navigation menu, navigate and click on Security
- Click on Security Command Center
- Click on SECURITY SOURCES
- Check whether Cloud Anomaly Detection is Enabled or not

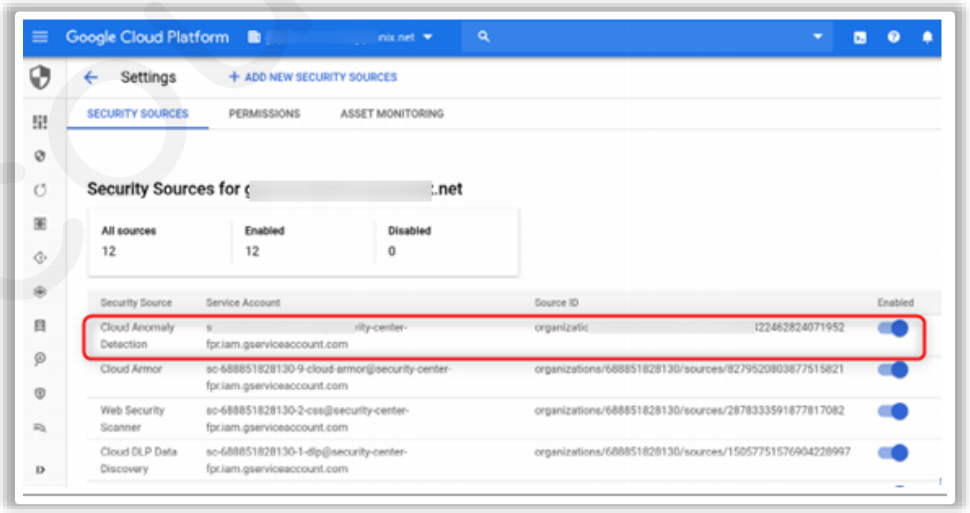
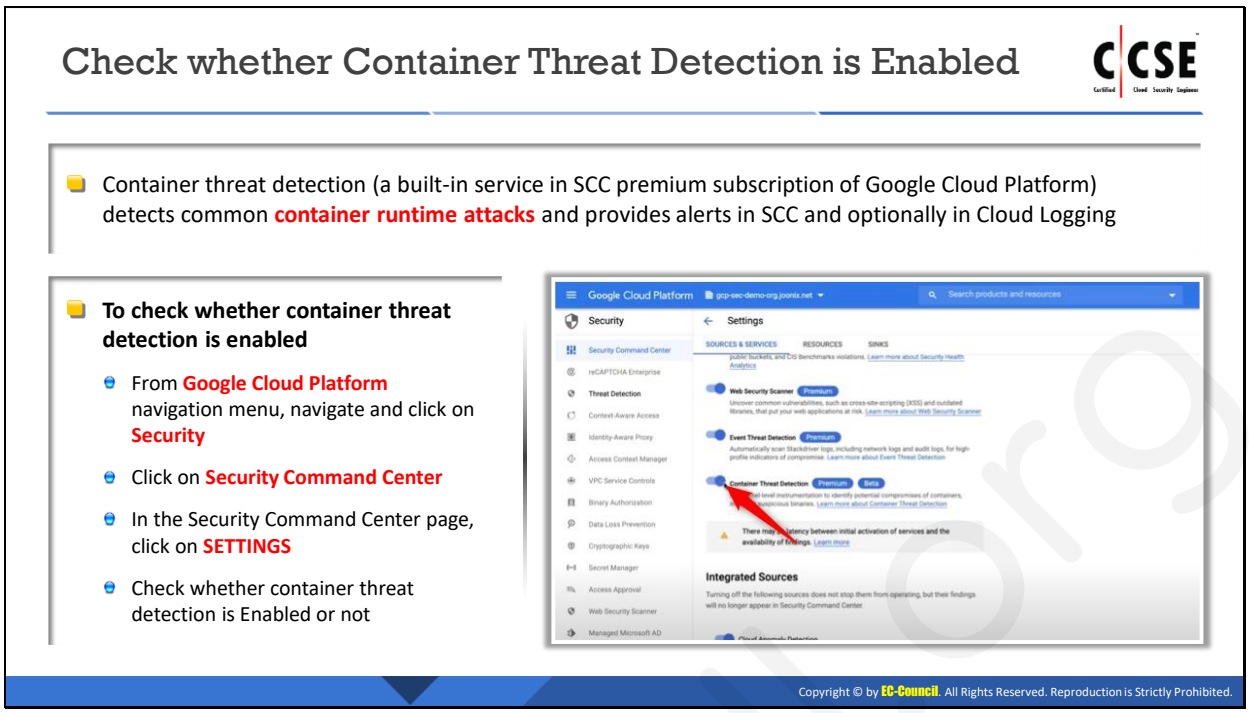


Figure 6.28: Screenshot Showing Google Cloud Platform Settings



Check Whether Container Threat Detection is Enabled

Container threat detection (a built-in service in SCC premium subscription of Google Cloud Platform) detects common container runtime attacks and provides alerts in SCC and optionally in Cloud Logging. To check whether container threat detection is enabled:

- From Google Cloud Platform navigation menu, navigate and click on Security
- Click on Security Command Center
- In the Security Command Center page, click on SETTINGS
- Check whether container threat detection is Enabled or not

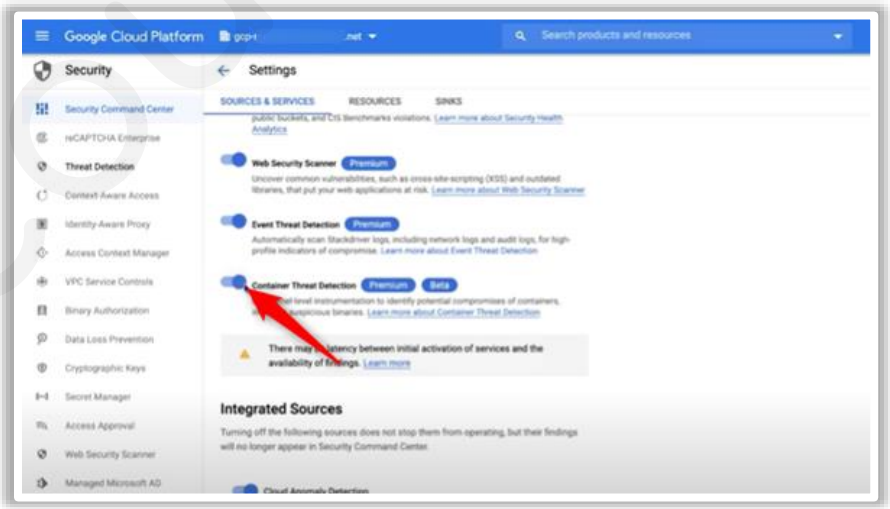
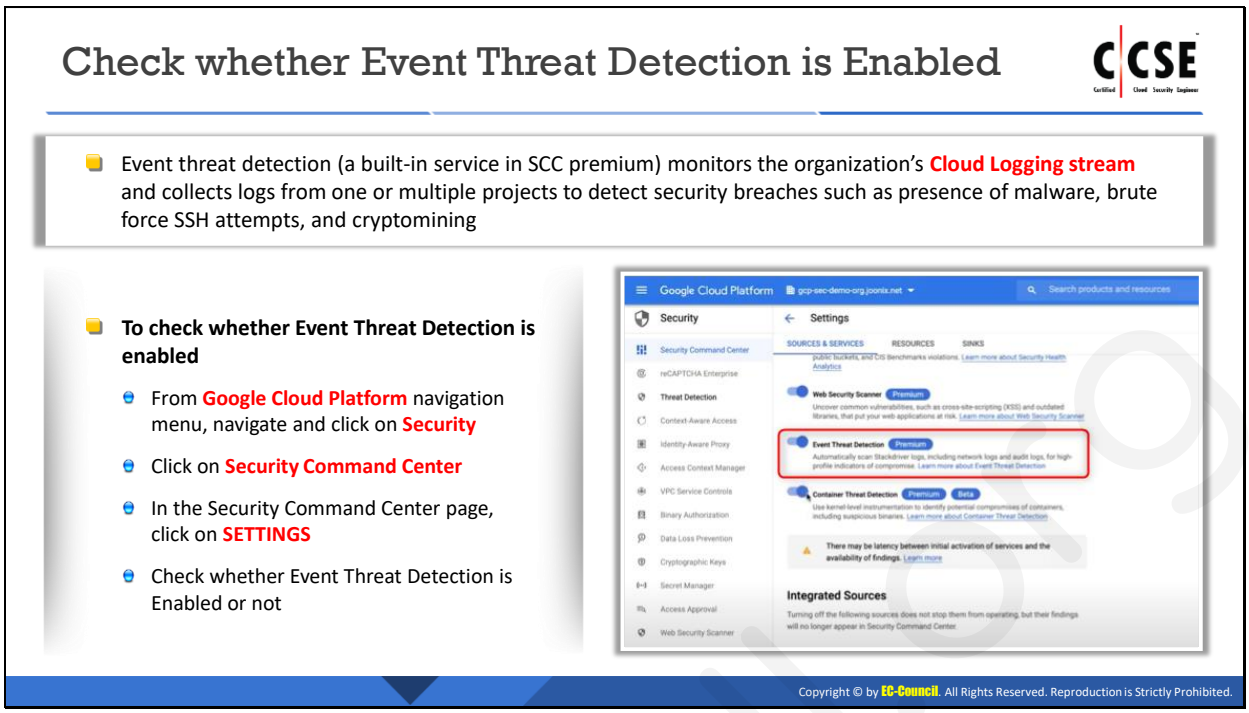


Figure 6.29: Screenshot Showing Google Cloud Platform Settings



Check Whether Event Threat Detection is Enabled

Event threat detection (a built-in service in SCC premium) monitors the organization's Cloud Logging stream and collects logs from one or multiple projects to detect security breaches such as presence of malware, brute force SSH attempts, and cryptomining. To check whether Event Threat Detection is enabled:

- From Google Cloud Platform navigation menu, navigate and click on Security
- Click on Security Command Center
- In the Security Command Center page, click on SETTINGS
- Check whether Event Threat Detection is Enabled or not

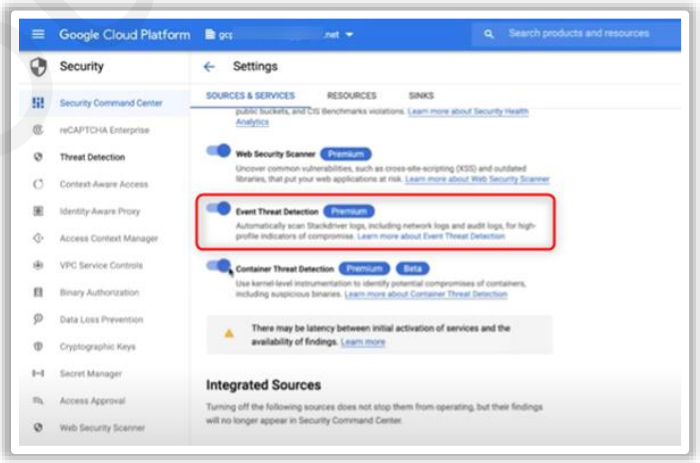
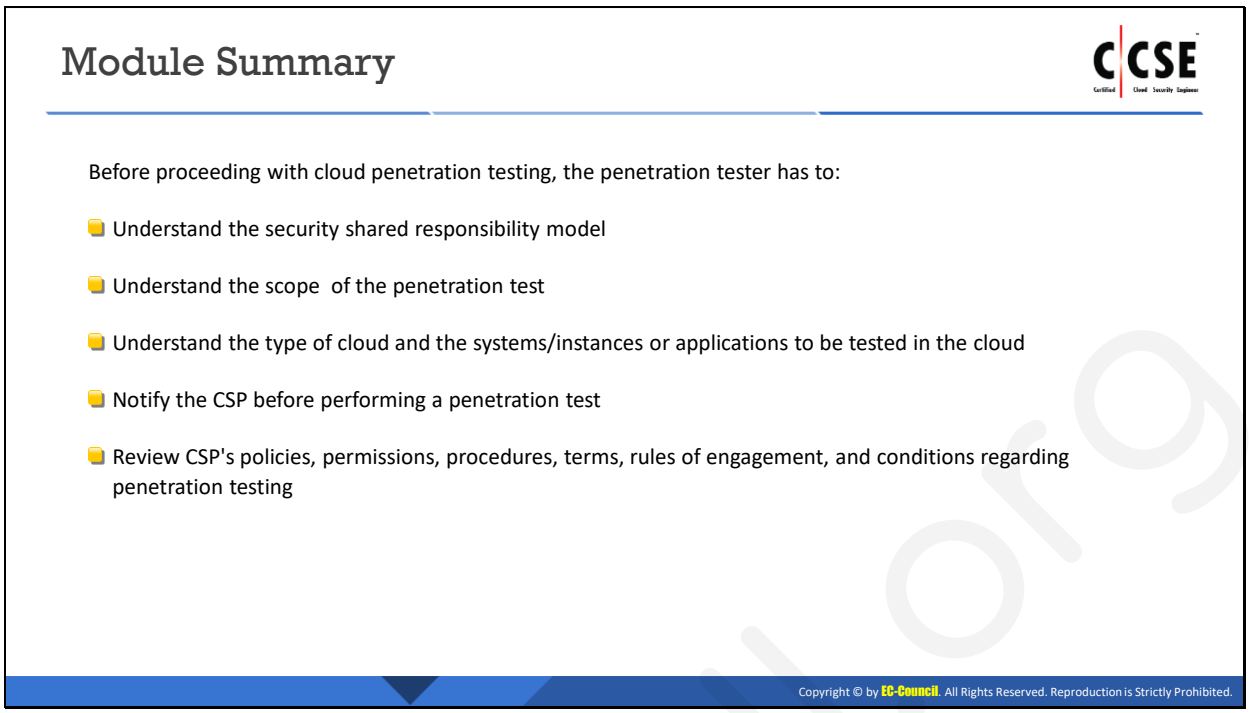


Figure 6.30: Screenshot Showing Google Cloud Platform Settings



Module Summary

In this module focuses on how to implement a comprehensive penetration testing methodology for assessing security of organization's cloud infrastructure.

The key points discussed in this module are stated below:

- Before proceeding with cloud penetration testing, the penetration tester has to:
  - Understand the security shared responsibility model
  - Understand the scope of the penetration test
  - Understand the type of cloud and the systems/instances or applications to be tested in the cloud
  - Notify the CSP before performing a penetration test
  - Review CSP's policies, permissions, procedures, terms, rules of engagement, and conditions regarding penetration testing

Penetration Testing in Cloud

Module 06

Caution: Completing the steps in this exercise will require you to sign up for third party services that may incur fees. Be sure to follow proper guidelines for removing billable services from your account before you leave your lab. Deviating from our specific instructions may result in unwanted fees for services from third-party service providers. Be sure to check with your school, instructor, or employer if education accounts are available. If you sign-up and provide your credit card, you will be responsible for any fees related to services you activate. We strongly advise you not to deviate from our explicit instructions while connected to the platforms unless you are fully aware of what the services are and what the respective third-party charges for their use.

Lab 1

Identifying Misconfigured S3 Buckets in AWS by Penetration Testing

A misconfigured AWS S3 bucket will be vulnerable to different types of threats. A penetration testing of AWS S3 buckets helps in identifying misconfigured and secure buckets.

ICON KEY

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Scenario

Amazon simple storage service (S3) is an AWS service that can be used to store data. It allows a cloud security engineer to securely store data, ensure its availability, and improve performance. If the S3 buckets are not appropriately configured, they will be vulnerable to different threats. Cloud security engineers can perform penetration testing to identify the misconfigured and secured S3 buckets.

Lab Objectives

This lab will demonstrate how to create a secure S3 bucket, create a vulnerable S3 bucket, install AWS CLI to create a penetration testing environment in Windows VM, and perform penetration testing to identify a vulnerable S3 bucket.

In this lab, you will learn to do the following:

- Create a secure S3 bucket
- Create a vulnerable S3 bucket
- Install AWS CLI in Windows VM
- Perform penetration testing to identify vulnerabilities

Lab Environment

To perform this lab, you need the following:

- **Admin Machine** VM

- **Administrative** privileges
- Registered AWS account

Lab Duration

Time: 20 minutes

Overview of AWS S3

Amazon S3 allows AWS customers to upload and retrieve data anytime, from anywhere on the web. Amazon S3 uses buckets to store objects. An object refers to any type of file, such as a text file, a video, or a photo. While adding files to Amazon S3, customers have the option to include their metadata and set permissions to control access to them. Customers can control access to each bucket (for example, who can list, create, and delete objects in a bucket), choose the geographical region where the bucket will be stored, and view the access logs for a bucket and its objects.

Cloud security engineers can provide groups of users with read/write access to S3 buckets or objects within them. By default, public access to S3 buckets is blocked. If a cloud security engineer accidentally configures a bucket for public read and write access, it becomes vulnerable and anyone with the bucket name can access the data and perform malicious activities.

Lab Tasks

Note: Web applications in a cloud environment may undergo frequent updates. As we are working on a cloud-based environment for this lab (i.e., AWS), the application interface may be updated with time. Hence, in case you happen to work on an updated version of AWS, the user interface you see on the application might differ from what you see in the lab. Consequently, the steps and screenshots demonstrated in this lab might also differ.

Note: Before starting this lab, you should create an AWS account using the following link: https://portal.aws.amazon.com/billing/signup. Once the registration is complete, perform the following tasks.

Note: You can also use any existing AWS account but be aware that it may incur significant charges to your account.

TASK 1

Creating a Secure S3 Bucket

1. Launch the **Admin Machine** VM. Log in with the following credentials: user- **Admin** and password- **admin@123**.

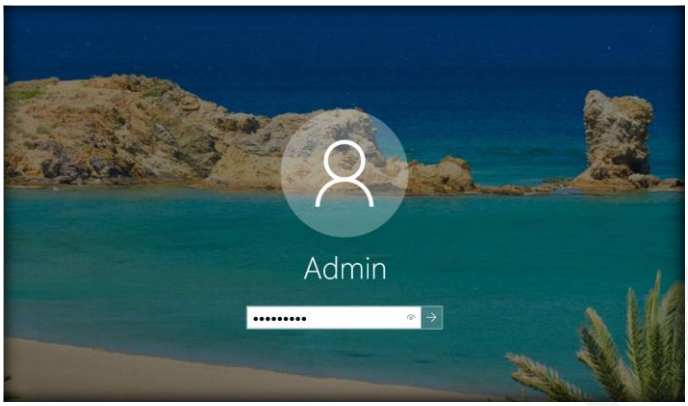


FIGURE 6.1.1: Launch Admin Machine and Log in

2. To open the browser, double-click on the **Google Chrome** icon on the desktop.

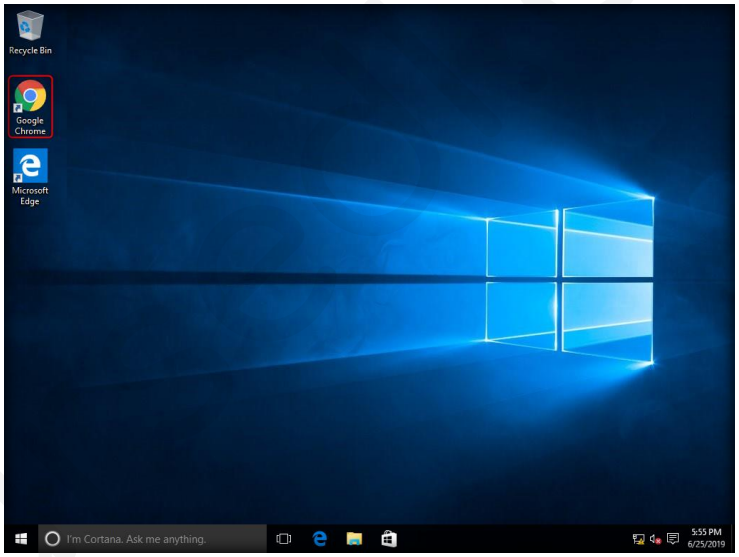


FIGURE 6.1.2: Navigating to the Chrome Browser from Taskbar

3. The **Google Chrome** browser opens. Go to the address bar, type **https://aws.amazon.com/**, and press **Enter**.

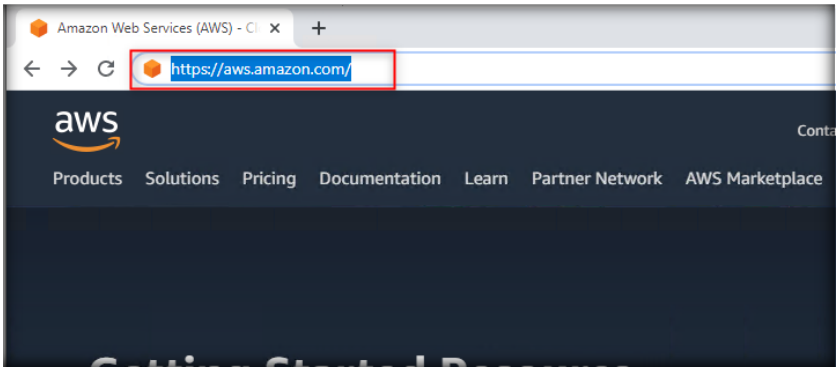


FIGURE 6.1.3: Access AWS in Browser

4. The **AWS Web Services - Cloud Computing Services** page appears now. Click on **AWS Management Console** from the **My Account** dropdown, as shown in the screenshot below.

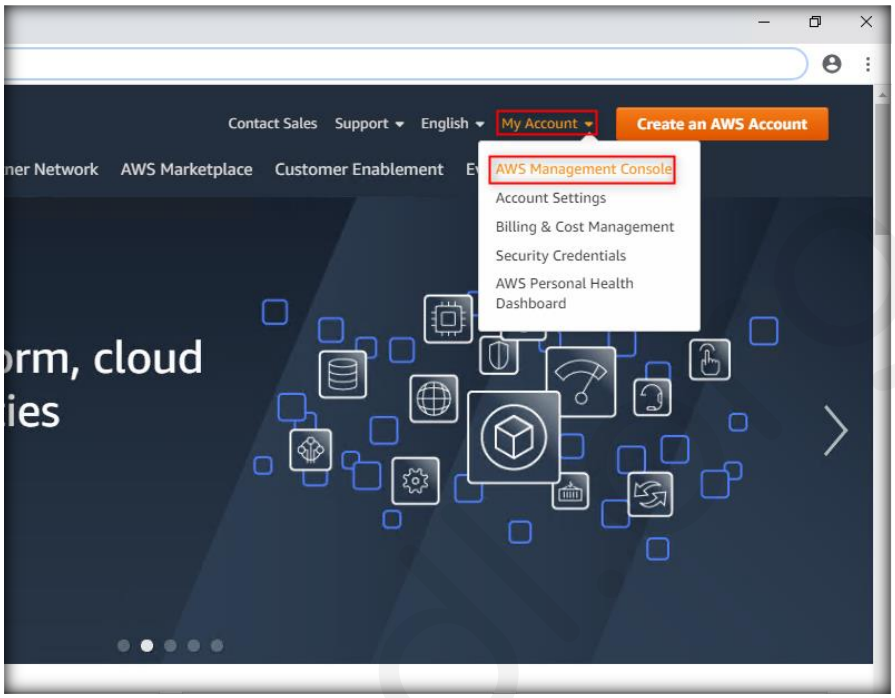


FIGURE 6.1.4: Navigate to AWS Management Console

5. The **AWS Web Services Sign-in** page appears. Choose **Root user** and type the AWS administrator account ID (Root user email address), as shown in the screenshot below, and then click on **Next**.

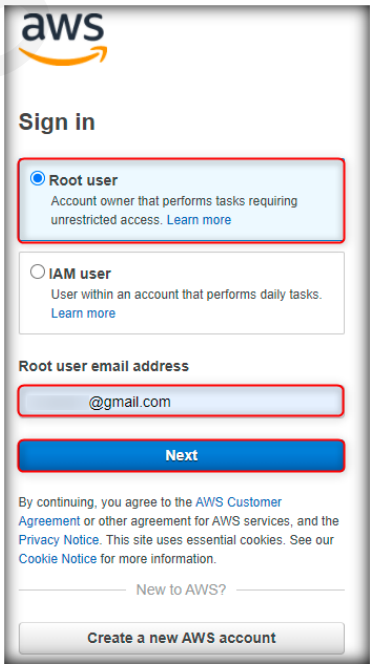


FIGURE 6.1.5: Typing the Account Name

6. In the security check page that appears, type the characters shown in the image and click **Submit**.

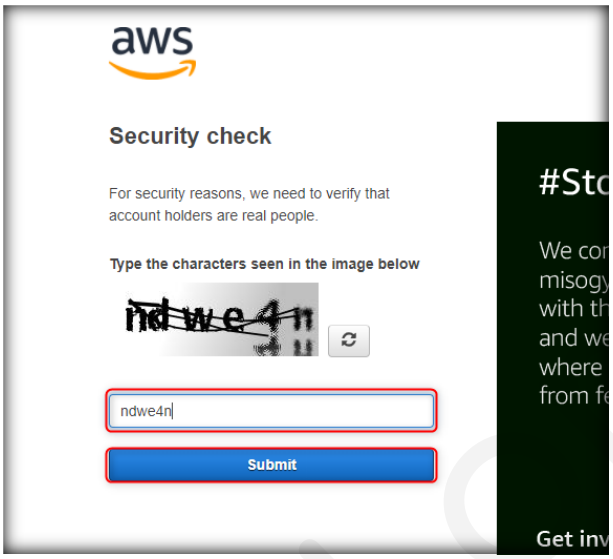


FIGURE 6.1.6: Typing Characters

7. Type the password in the **Password** field and click on **Sign in**, as shown in the screenshot below.

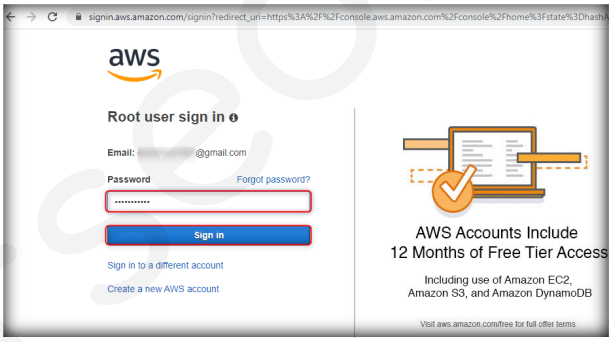


FIGURE 6.1.7: Sign-in to AWS

8. Select **Services** from the menu bar and click on **S3** under the **Storage** section of **All Services** page.

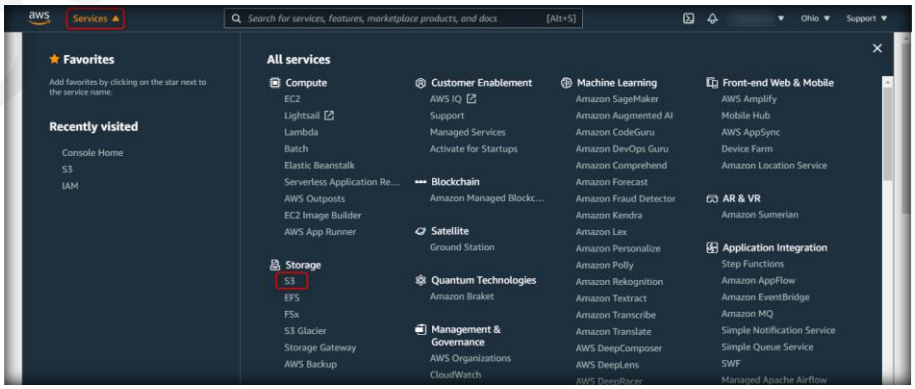


FIGURE 6.1.8: Navigate to S3

9. The **Amazon S3** page appears now. Click on **Create Bucket** at the right corner of the **Buckets** pane.

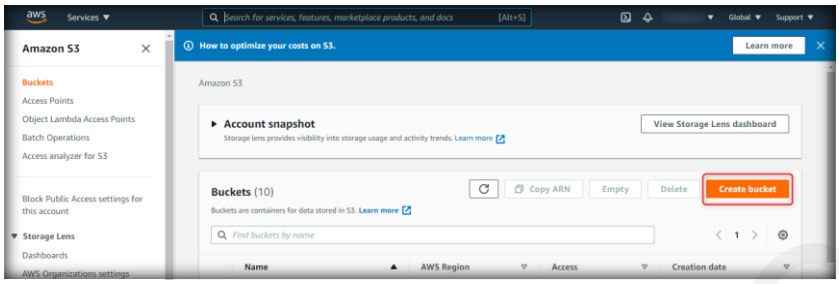


FIGURE 6.1.9: Creating Bucket

10. In the **Create bucket** window that opens, give an appropriate name and **AWS Region** for the bucket under **General configuration**. In this lab, we are providing the following **General configuration**:

Bucket name: ccsedemobucket-1

AWS Region: us-east-2

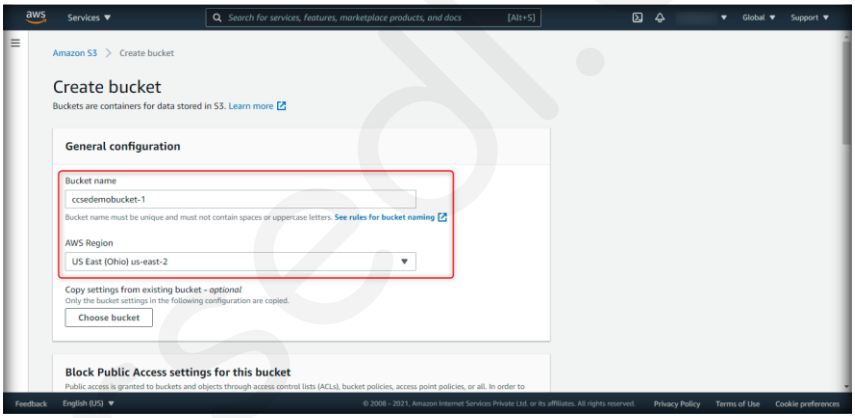


FIGURE 6.1.10: Configuring Bucket

11. Scroll down to the bottom. Under **Block Public Access settings for this bucket**, ensure the checkbox for **Block all public access** is selected.

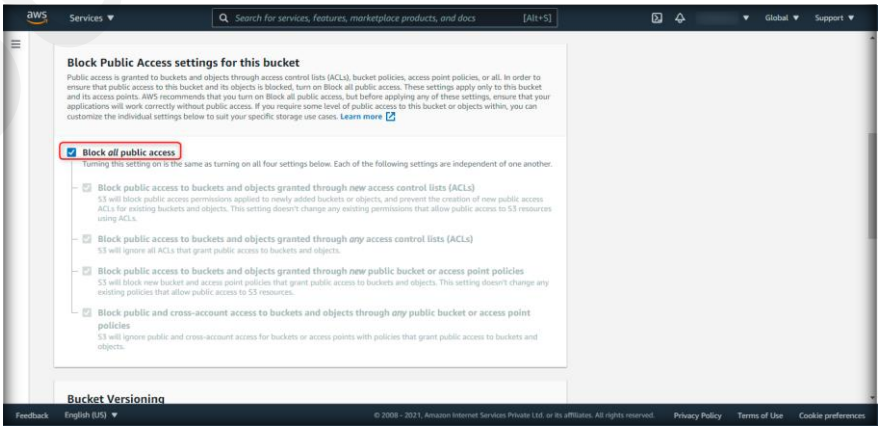


FIGURE 6.1.11: Blocking Public Access

12. Scroll down to the bottom and click on **Create bucket** to create your bucket.

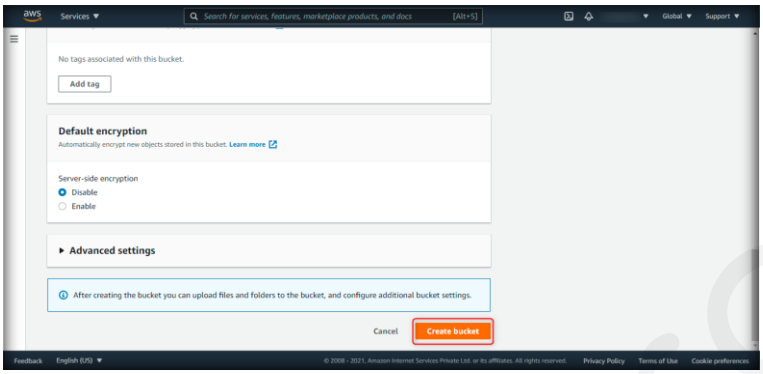


FIGURE 6.1.12: Creating Bucket

13. Now, to upload objects into the bucket, scroll down in the **Amazon S3** dashboard, locate your bucket named **ccsedemobucket-1**, and click on it.

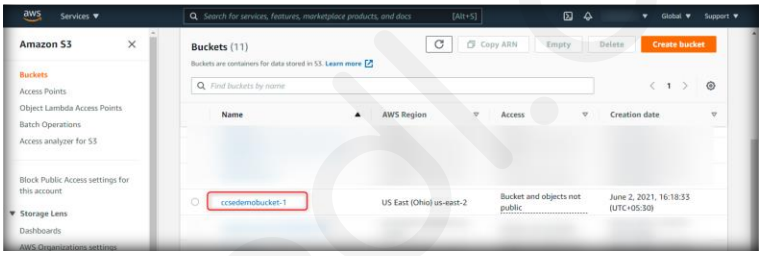


FIGURE 6.1.13: Selecting the Bucket

14. Click on **Upload** under the **Objects** tab.

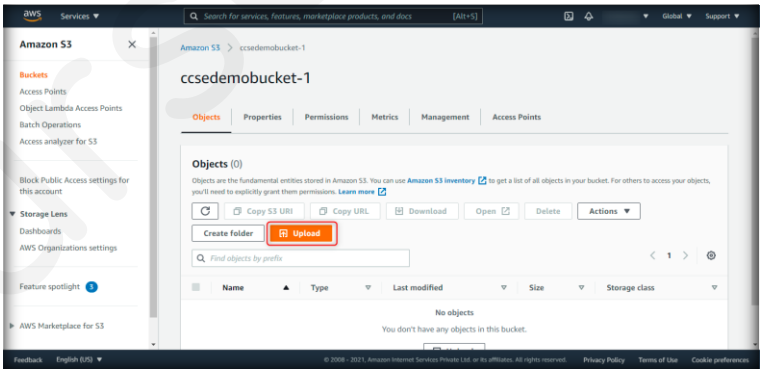


FIGURE 6.1.14: Uploading Objects

15. In the **Upload** window that appears, click on **Add files**.

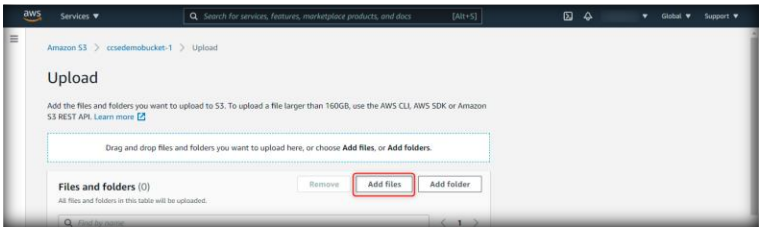


FIGURE 6.1.15: Adding Files

16. Now, create a **ccsedemo-1.txt** file, select the created text file **ccsedemo-1** from the local VM, and then click on **Open**.

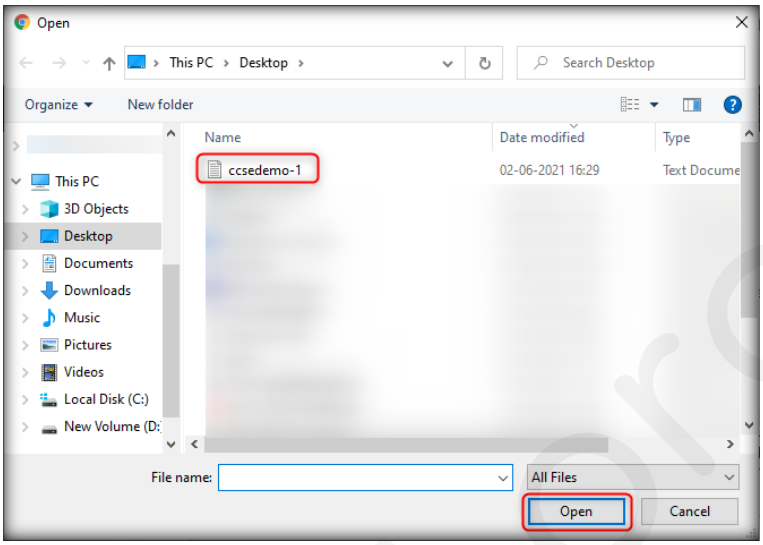


FIGURE 6.1.16: Selecting File

17. Scroll down and click on **Upload** at the bottom.

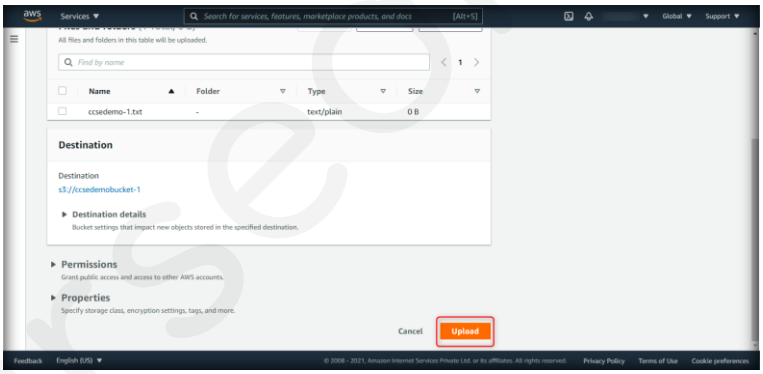


FIGURE 6.1.17: Uploading File

18. In the subsequent window that appears, click on **ccsedemo-1.txt** under **Files and folders**.

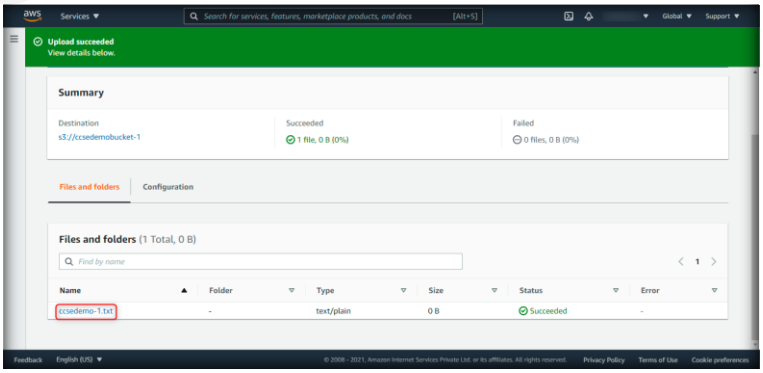


FIGURE 6.1.18: Selecting the Uploaded file

19. It will take you to another window named **ccsedemo-1.txt**. Click on the **Permissions** tab.

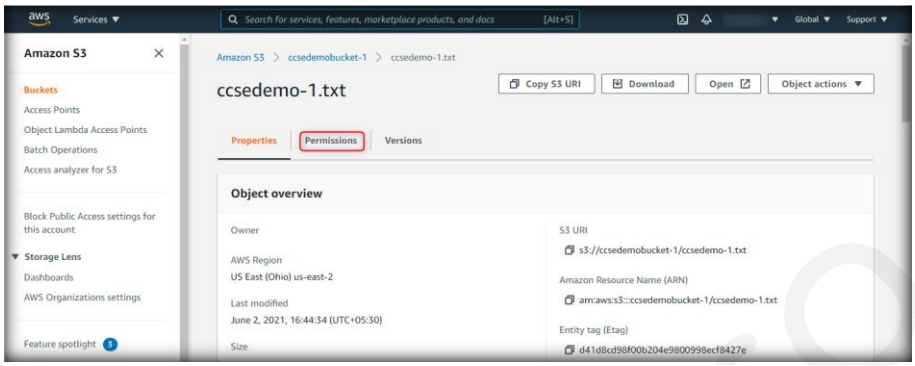


FIGURE 6.1.19: Click on Permissions

20. Click on **Edit** at the right corner of **Access control list (ACL)**.

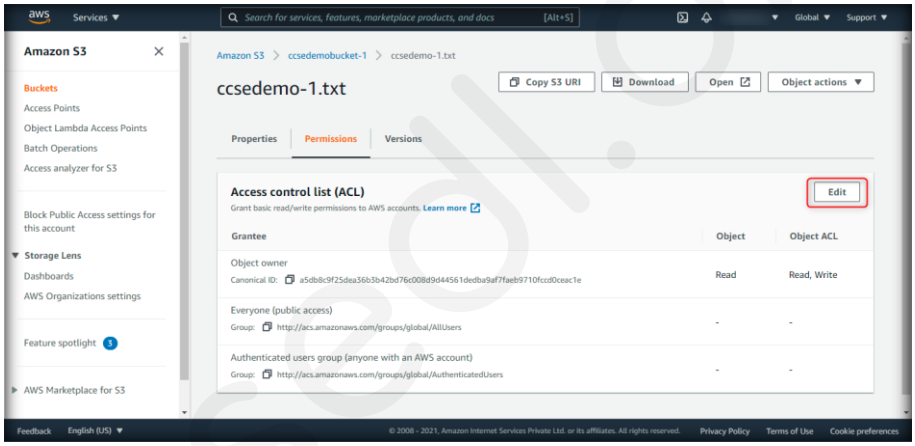


FIGURE 6.1.20: Editing ACL

21. In the **Access control list (ACL)** window that appears, you will see that the **Read/Write** options are not active for **Everyone (public access)**. This ensures the S3 bucket named **ccsedemobucket-1** is secure and its objects cannot be accessed by everyone.

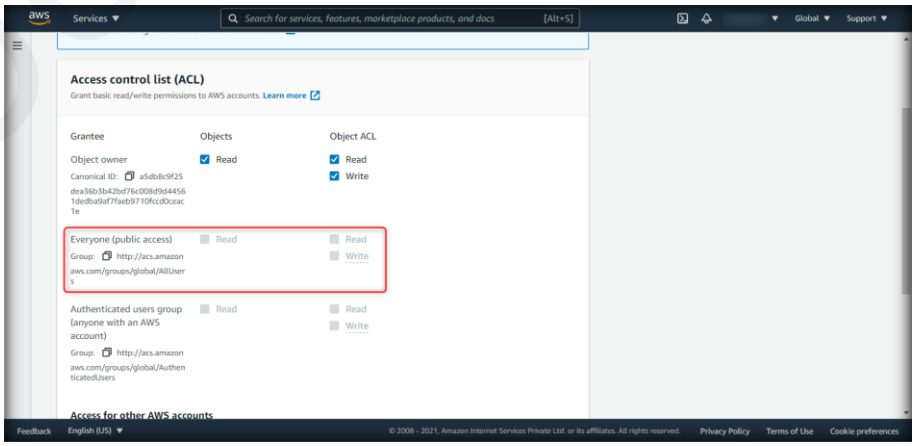


FIGURE 6.1.21: Verifying the Public Access Permissions

TASK 2

Creating vulnerable S3 bucket

22. Now, you need to create a vulnerable S3 bucket. To do this, click on **Services** at the top of the AWS console. Then click on **S3** under **Storage**.

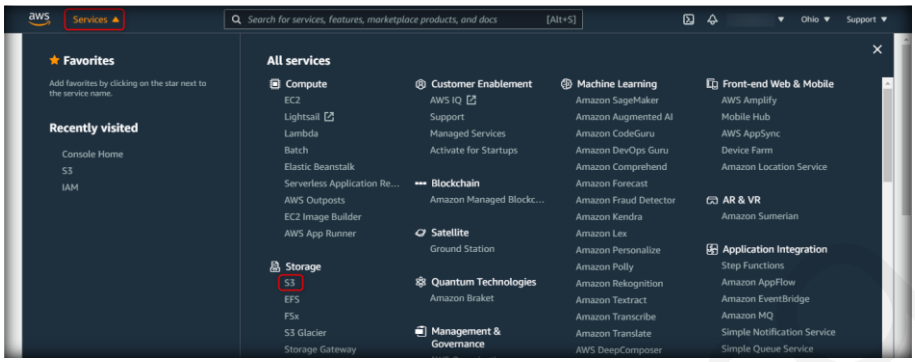


FIGURE 6.1.22: Navigating to S3

23. In the **Amazon S3** dashboard, click on **Create bucket** at the right corner of the **Buckets** section.

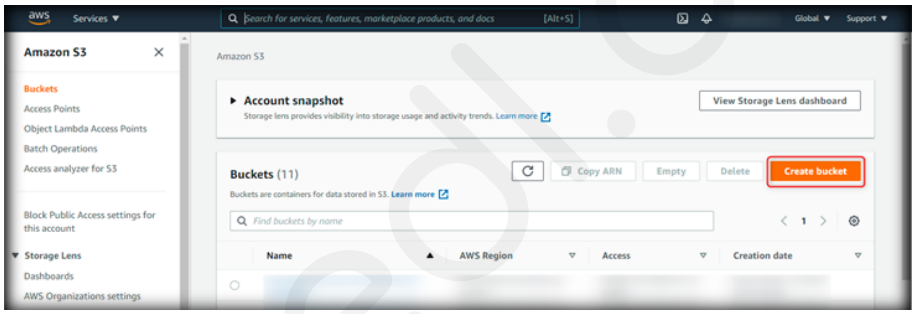


FIGURE 6.1.23: Click on Create bucket

24. In the **Create bucket** window, provide **an** appropriate bucket name and region under **General configuration**. In this lab, we have used the following general configuration:

Bucket name: ccsedemobucket-2

AWS Region: us-east-2

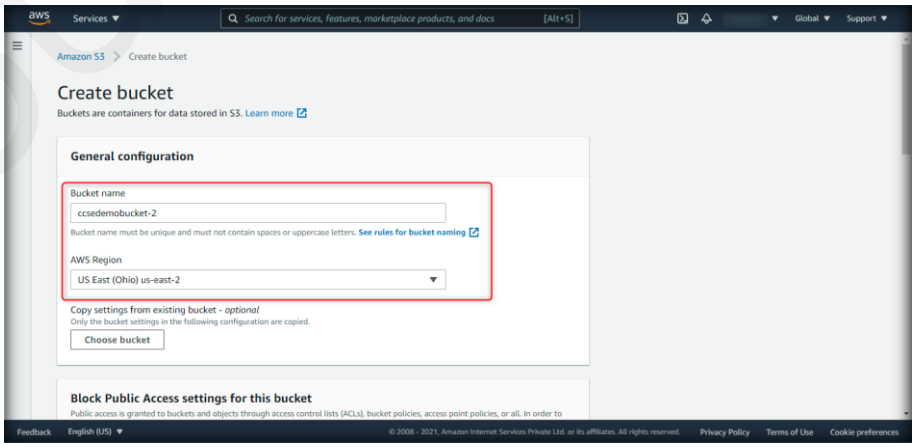


FIGURE 6.1.24: Configuring the bucket

25. Scroll down and uncheck **Block all public access** under **Block Public Access settings for this bucket**.

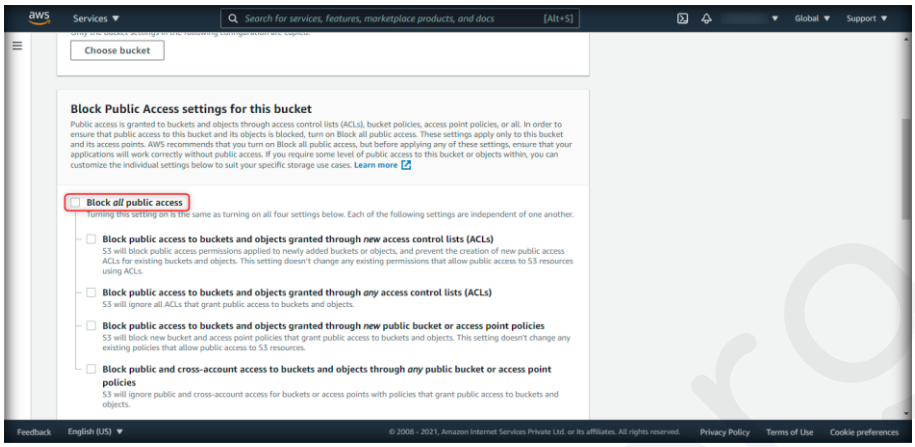


FIGURE 6.1.25: Uncheck block all public access

26. Scroll down and select the acknowledgement checkbox.

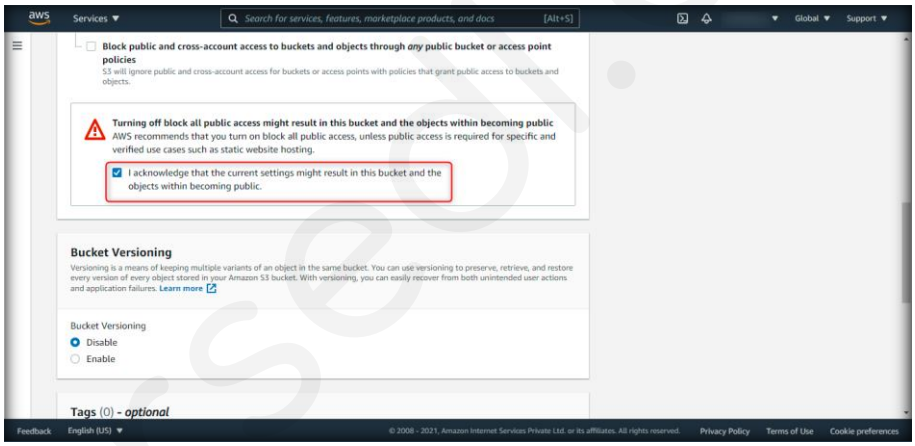


FIGURE 6.1.26: Selecting Acknowledgement

27. Scroll down and click on **Create bucket**.

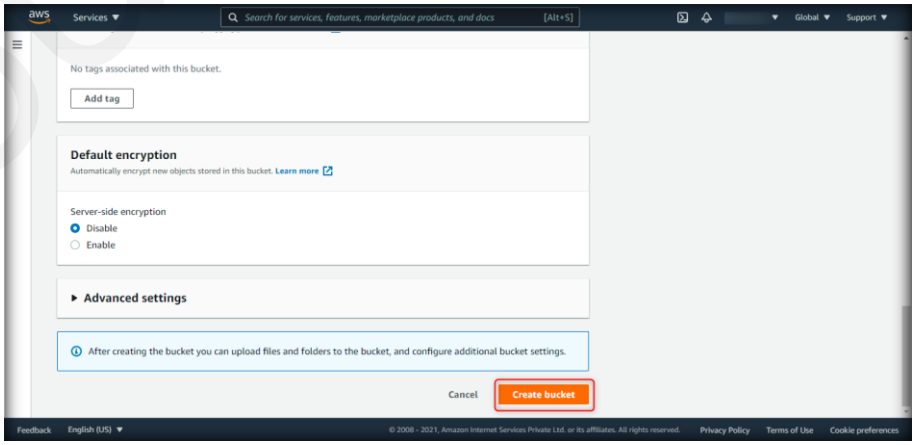


FIGURE 6.1.27: Creating Bucket

28. Now, you will be in the **Amazon S3** dashboard. Scroll down in the **Buckets** table to locate the bucket you created, **ccsedemobucket-2**, and click on it.

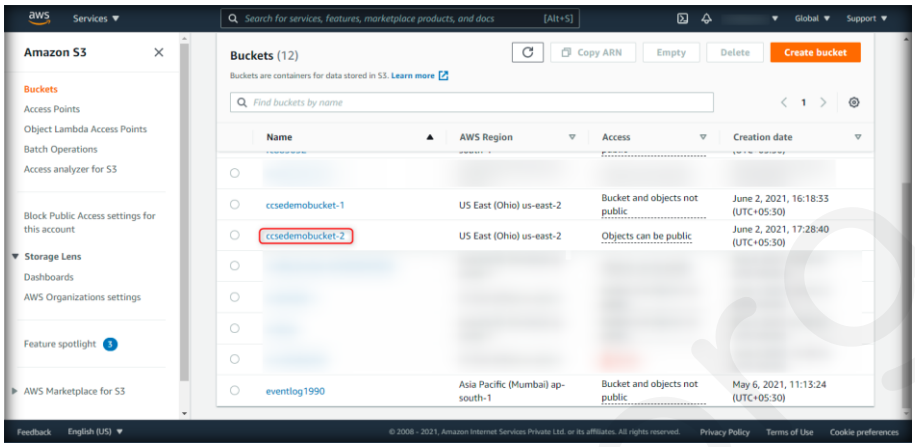


FIGURE 6.1.28: Selecting the bucket

29. In the **Objects** pane, click on **Upload**.

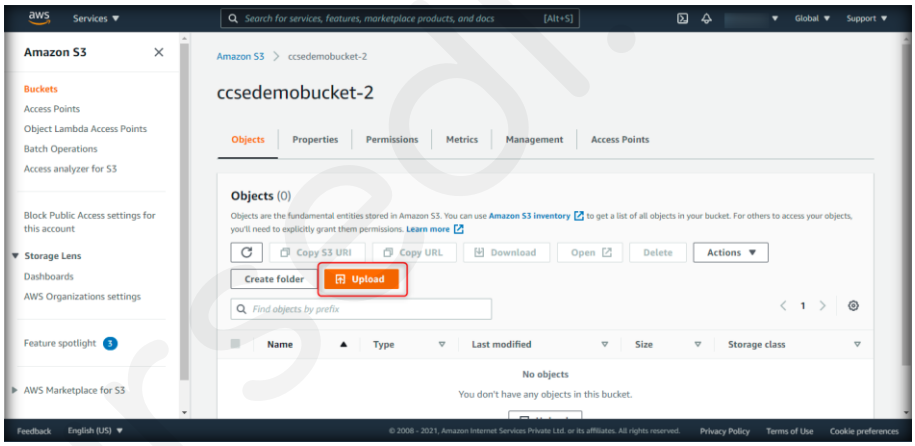


FIGURE 6.1.29: Uploading the file

30. Click on **Add files** in the subsequent **Upload** window.

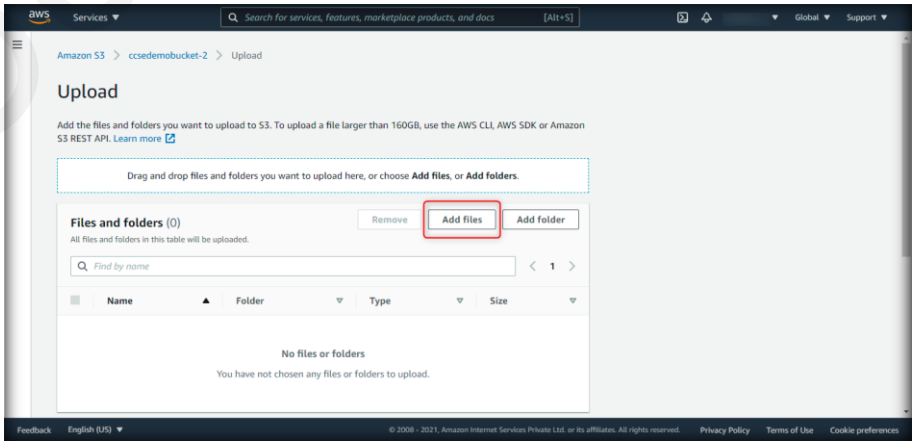


FIGURE 6.1.30: Adding the file

31. Create a **ccsedemo-2.txt** file and select it from the local VM's desktop.

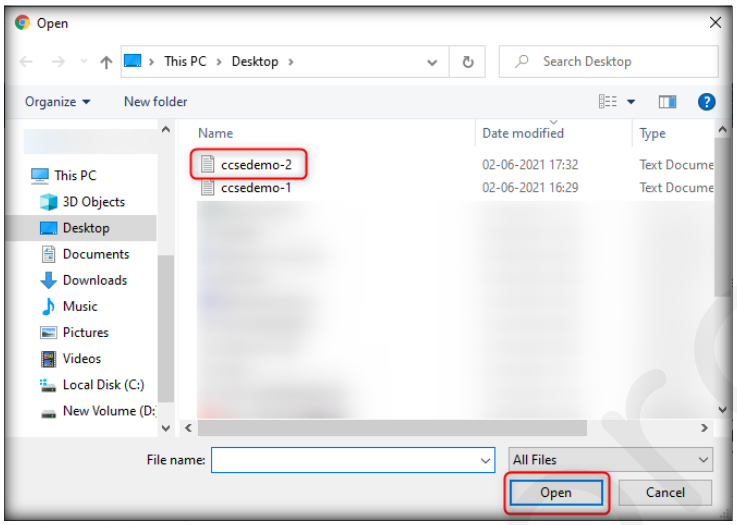


FIGURE 6.1.31: Selecting the file

32. Scroll down and click on **Upload**.

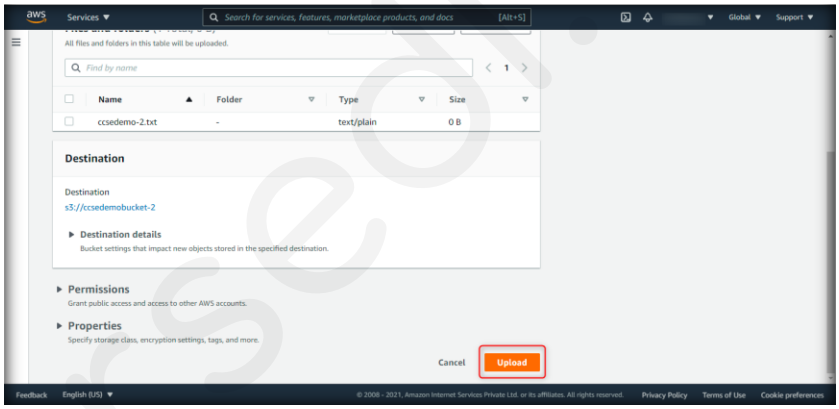


FIGURE 6.1.32: Uploading the file

33. After the text file is successfully uploaded, scroll down and click on **ccsedemo-2.txt** under **Files and folders**.

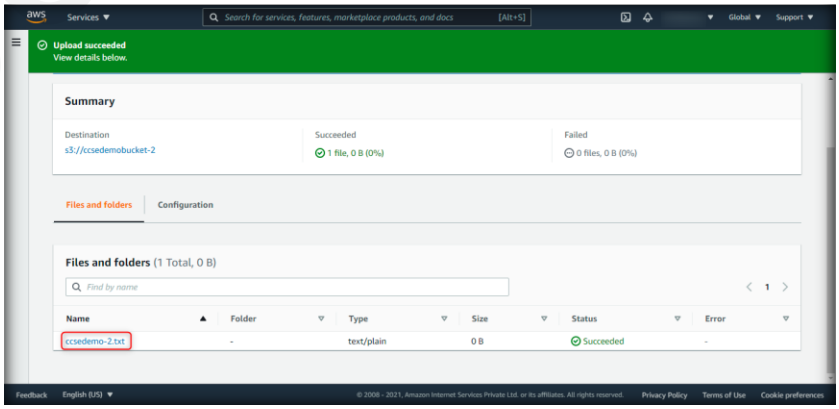


FIGURE 6.1.33: Click on the file

34. You will be taken to a new window that displays **ccsedemo-2.txt** properties. Click on the **Permissions** tab.

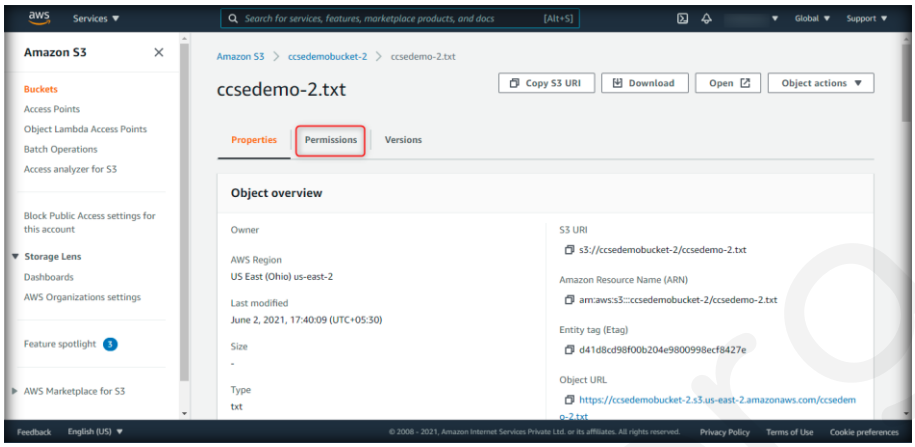


FIGURE 6.1.34: Click on permissions

35. Click on **Edit** to the right corner of **Access control list (ACL)**.

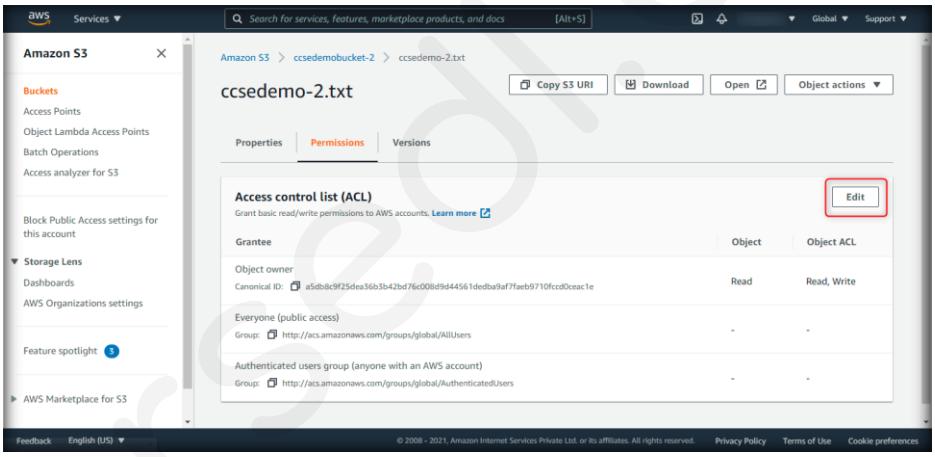


FIGURE 6.1.35: Editing ACL

36. Under **Access control list (ACL)**, select the **Read** checkboxes for **Objects** and **Object ACL** next to **Everyone (public access)**.

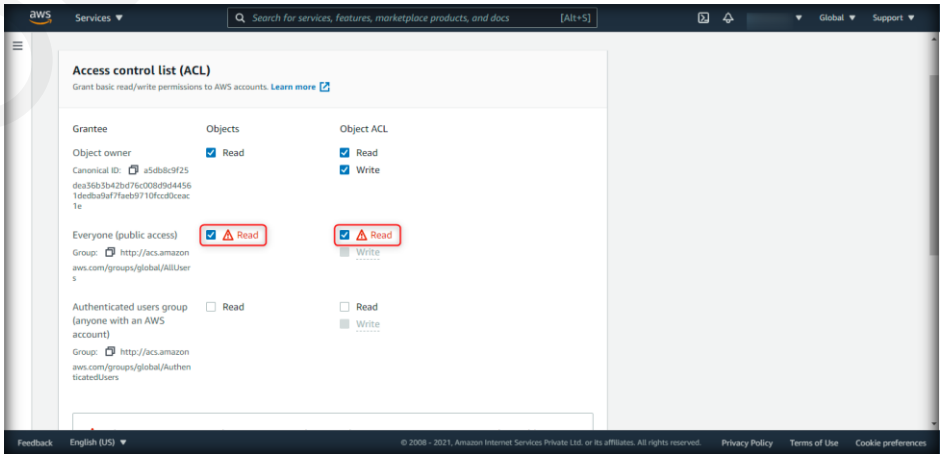


FIGURE 6.1.36: Selecting Read permissions

37. Scroll down and select the acknowledgement checkbox. Then, click on **Save changes**.

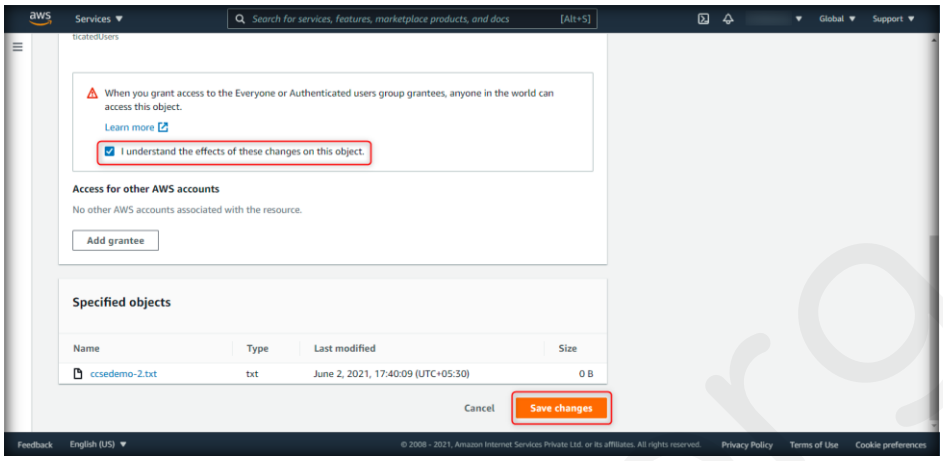


FIGURE 6.1.37: Saving Changes

38. Now, you need to enable full access to everyone for **ccsedemobucket-2** using **CloudShell**. Click on the **CloudShell** icon at the top right corner of AWS console.

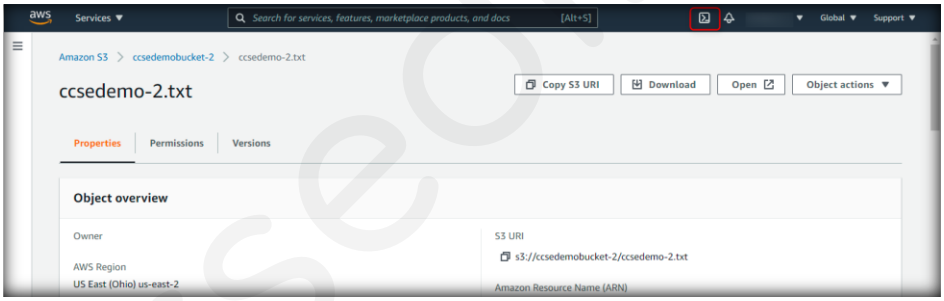


FIGURE 6.1.38: Selecting CloudShell

39. In the **CloudShell** window, type the following command to enable write access control list and press **Enter**.

```
aws s3api put-bucket-acl --bucket
ccsedemobucket-2 --grant-write-acp
uri=http://acs.amazonaws.com/groups/global/AllU
sers
```

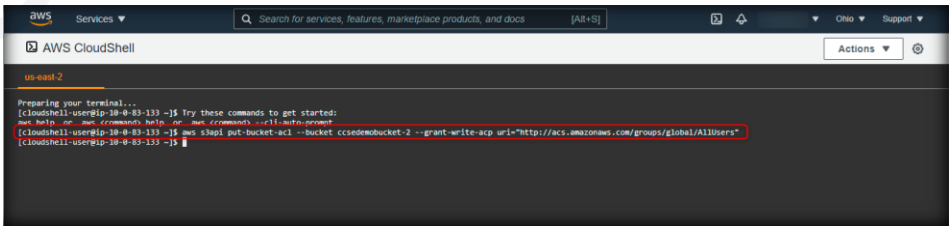


FIGURE 6.1.39: Adding Write permissions

40. Next, run the following command to grant full control to the resources.

```
aws s3api put-bucket-acl --bucket
ccsedemobucket-2 --grant-full-control
uri=http://acs.amazonaws.com/groups/global/AllU
sers
```

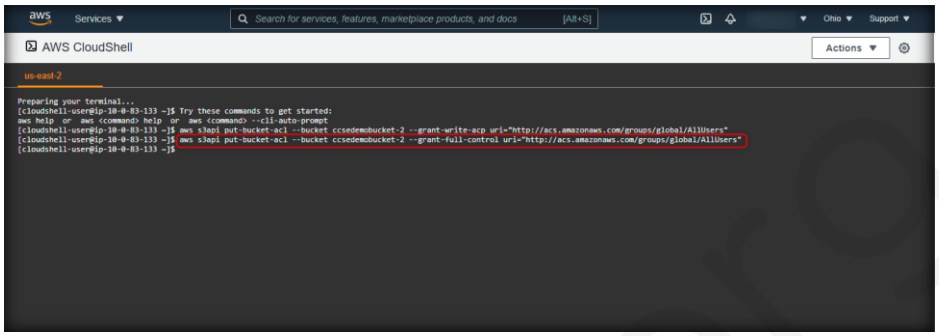


FIGURE 6.1.40: Granting Full Control

41. Now, go back to AWS console, click on **Services,** and then click on **S3** under **Storage**.

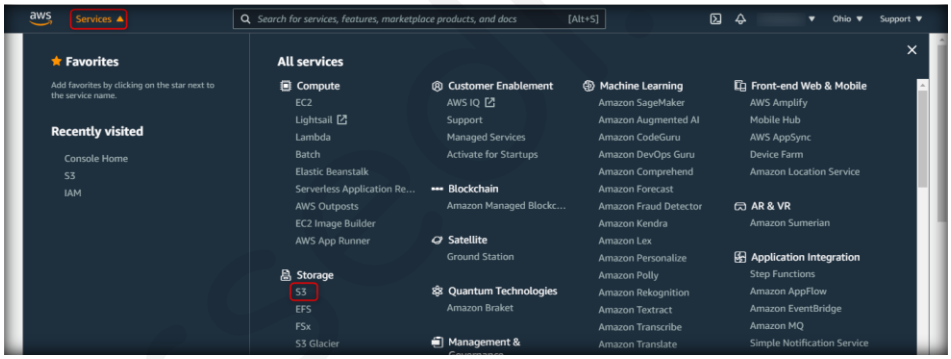


FIGURE 6.1.41: Navigating to S3

42. Click on **ccsedemobucket-2**.

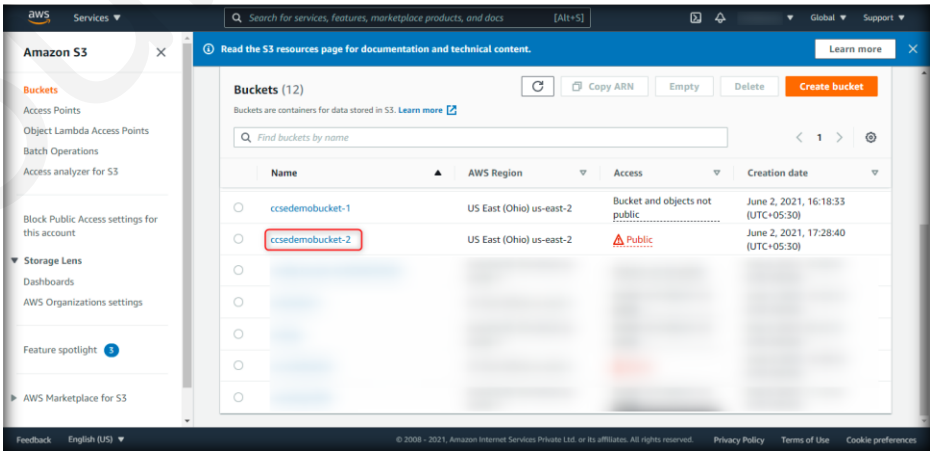


FIGURE 6.1.42: Selecting the S3 bucket

43. Click on **Permissions**.

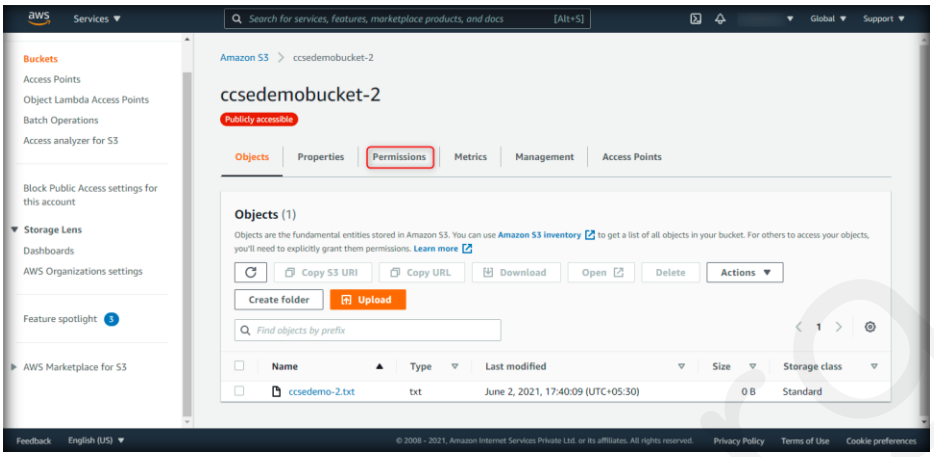


FIGURE 6.1.43: Click on Permissions

44. Scroll down and click on **Edit** in the **Access control list (ACL)** section.

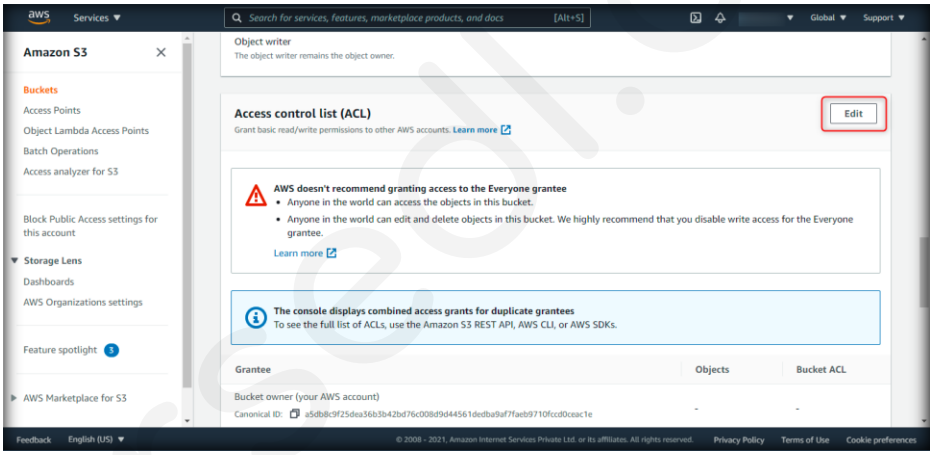


FIGURE 6.1.44: Editing ACL

45. You will see that all permissions are granted for **Everyone (public access)** for **Objects** and **Bucket ACL**.

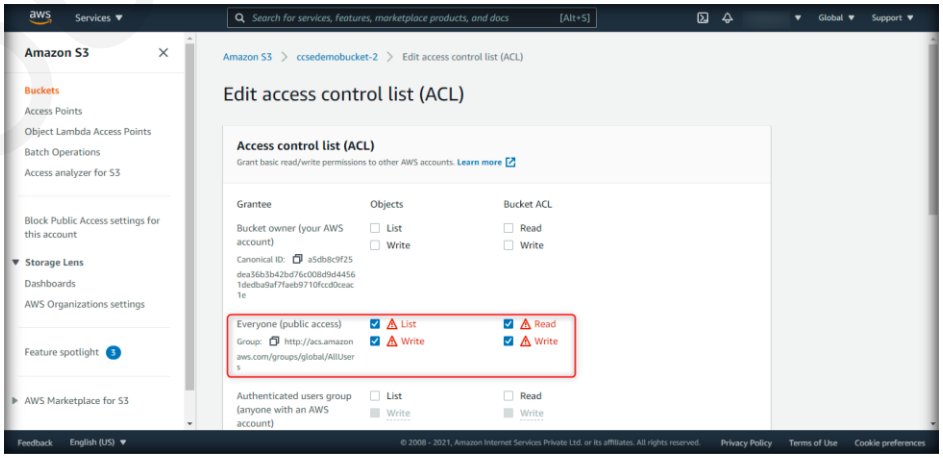


FIGURE 6.1.45: Viewing ACL

TASK 3

Installing AWS CLI in Windows VM

46. You have created a vulnerable bucket with full public access.
47. Next, To create a penetration testing environment, you need to install AWS CLI. To do this, navigate to **D:\CCSE\CCSE Tools\CCSE Module 06 Penetration Testing in Cloud\AWS CLI** and double-click on **AWSCLI64PY3.exe** on the setup file.
48. In the **AWS Command Line Interface Setup** window, click on **Next**.

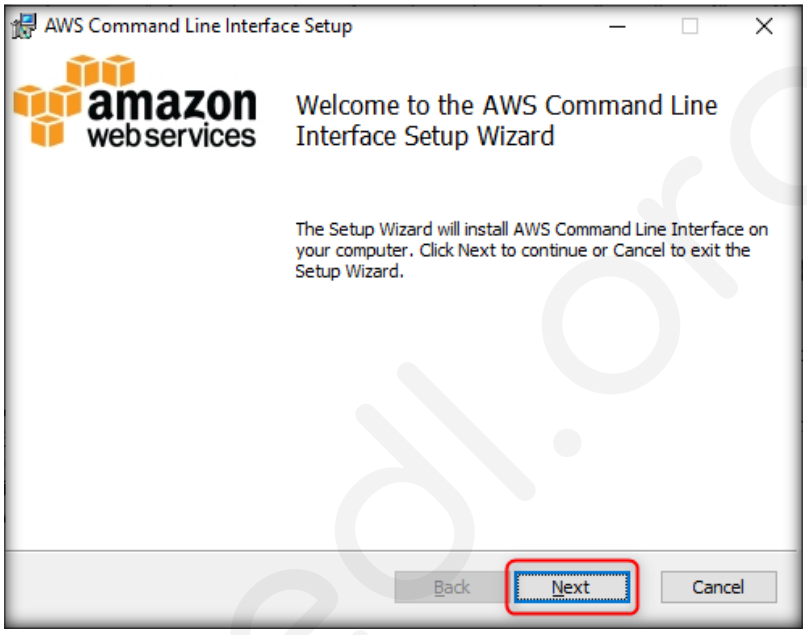


FIGURE 6.1.46: Installation of AWS CLI

49. Now, select the checkbox for accepting the license agreement and click on **Next**.

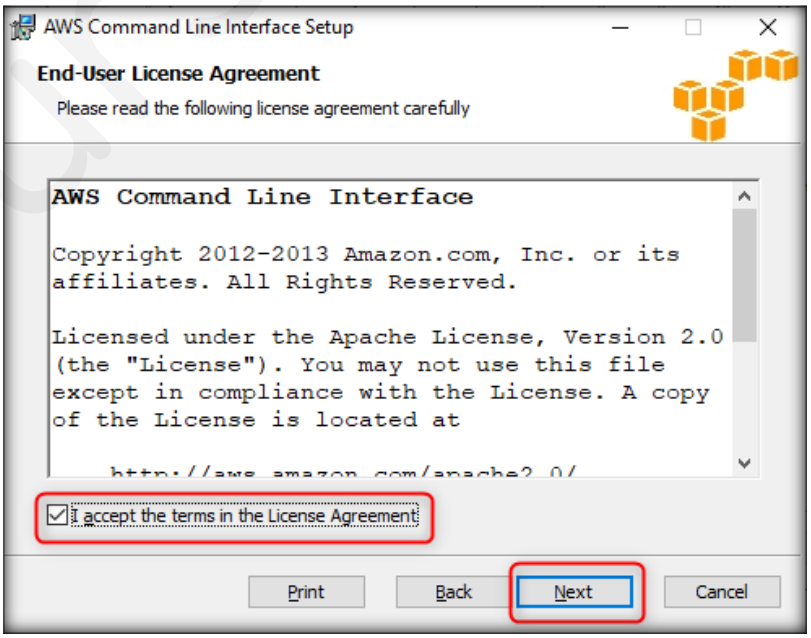


FIGURE 6.1.47: Accept License Agreement and continue

50. In the **Custom Setup** window, click on **Next**.

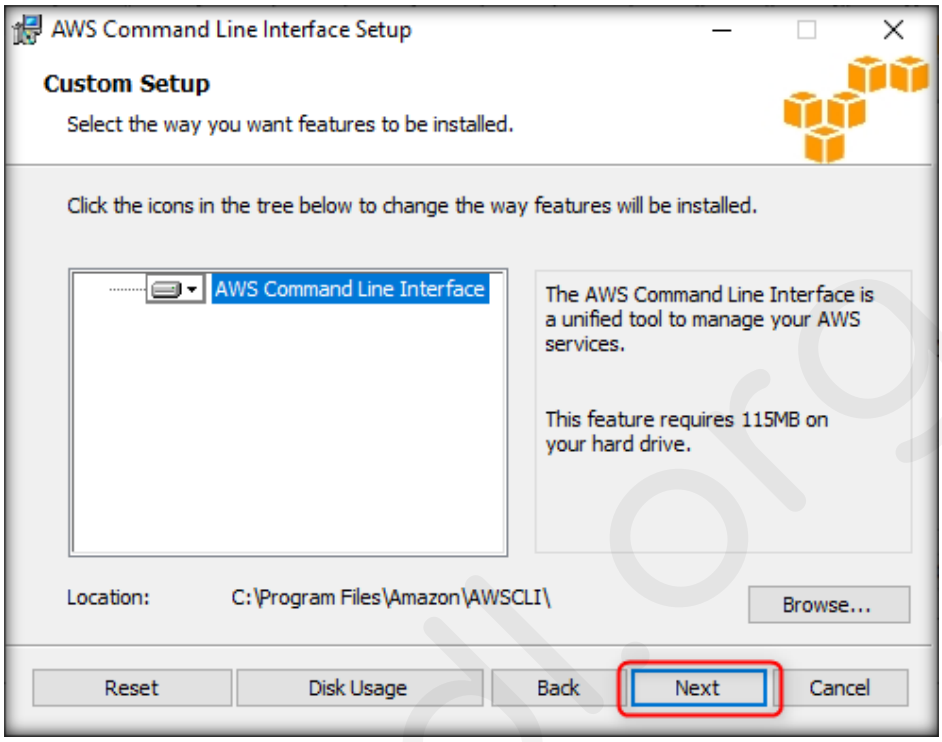


FIGURE 6.1.48: Continue Installation

51. In the next window, click on **Install**.

Note: If a **User Account Control** pop-up appears, click on **Yes**.

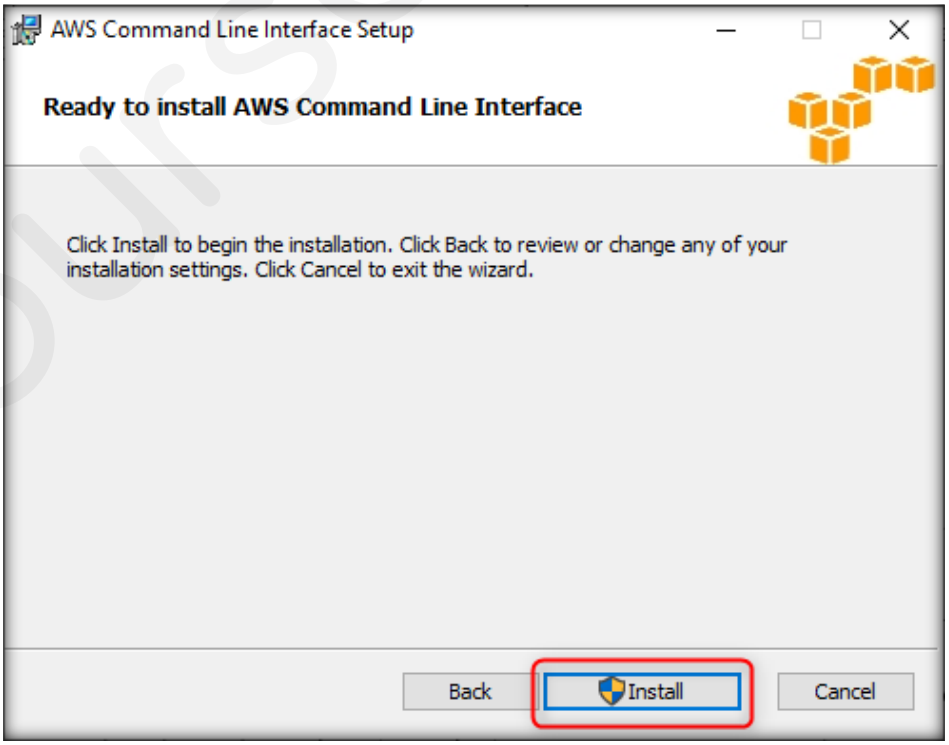


FIGURE 6.1.49: Install AWS CLI

52. When the installation is complete, click on **Finish**.

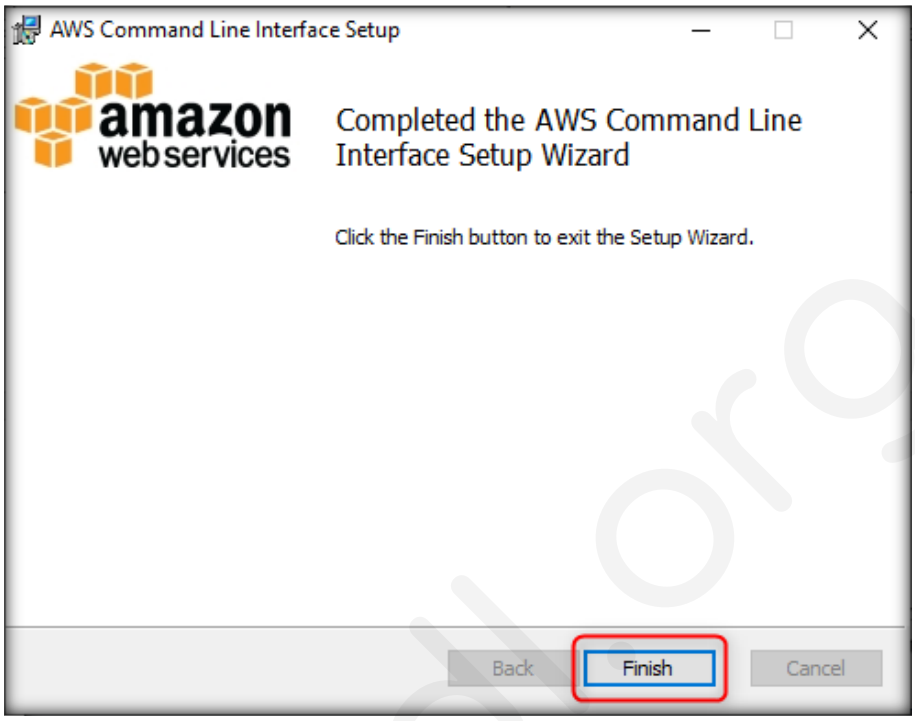


FIGURE 6.1.50: Finish Installation

53. Now, in the local VM machine, search for **cmd** at the bottom and click on **open**.

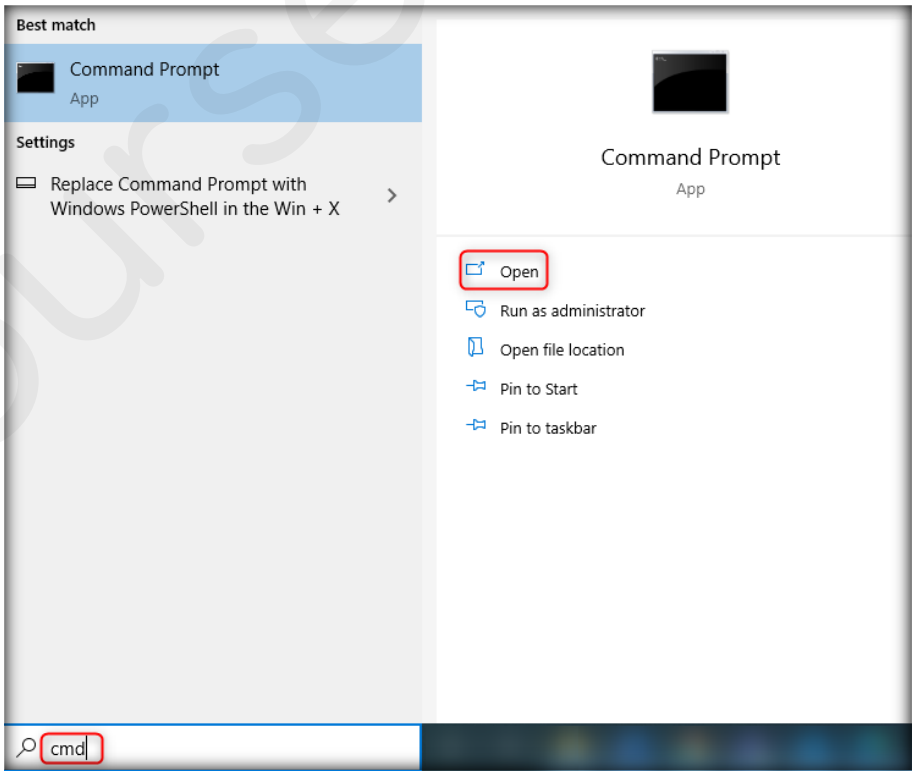


FIGURE 6.1.51: Open Command Prompt

54. Run the following command in the command prompt to confirm the installation.

```
aws --version
```

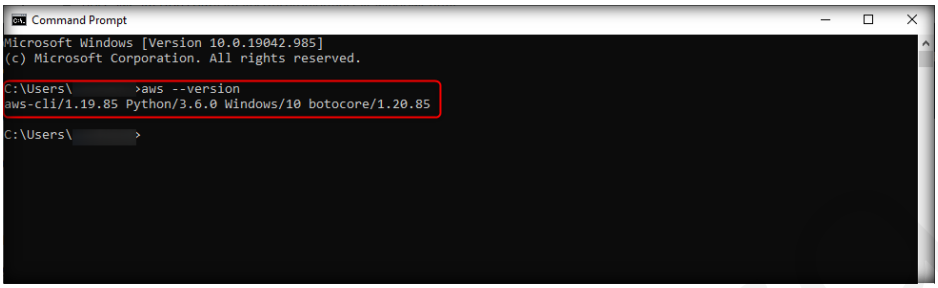


FIGURE 6.1.52: Running the Installation command

55. Now, you need to set the environment variable. Type the command that is highlighted in red in the following screenshot to get the location of the AWS CLI file path. Then, copy the path of the AWS CLI file.

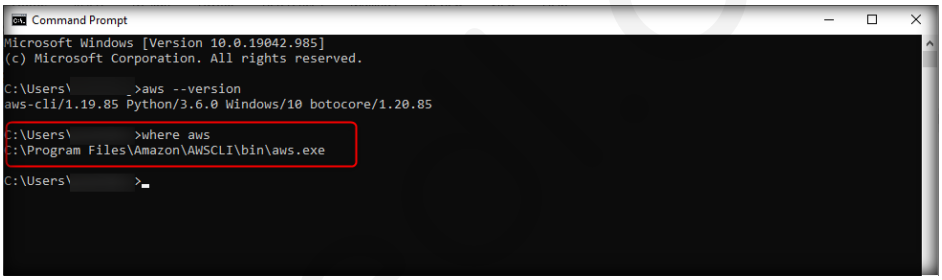


FIGURE 6.1.53: Finding AWS CLI Location

56. To execute the AWS CLI from command prompt, search for **Environment Variables** in the start menu of the local **Windows** VM. Then, click **Open** for **Edit the system environment variables**.

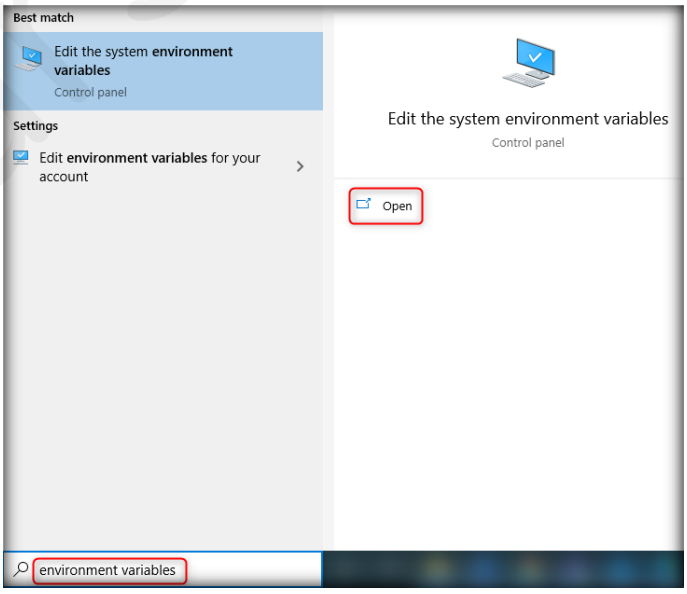


FIGURE 6.1.54: Opening Environment Variables

57. In the **Advanced** tab of the **System Properties** window, click on **Environment Variables**.

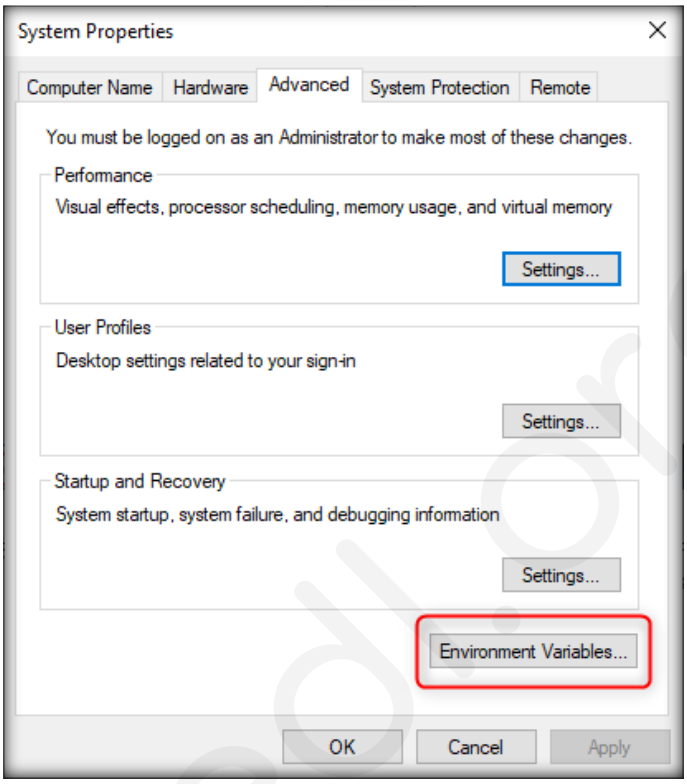


FIGURE 6.1.55: Editing Environment Variable

58. In the next window, select **Path** and click on **Edit** under **User variables**.

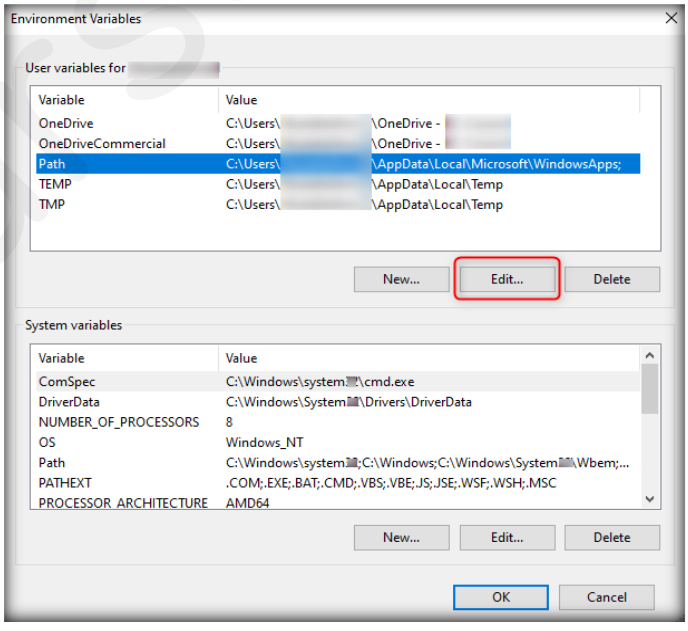


FIGURE 6.1.56: Editing Environment Variable

59. In the **Edit environment variable** window that appears, click on **New**.

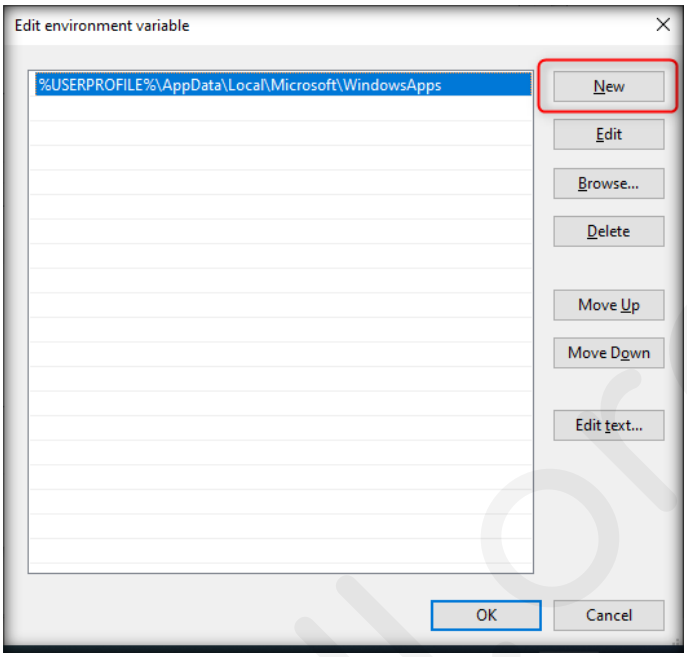


FIGURE 6.1.57: Adding New Environment variable

60. Paste the AWS CLI location you copied from the command line and click on **OK.**

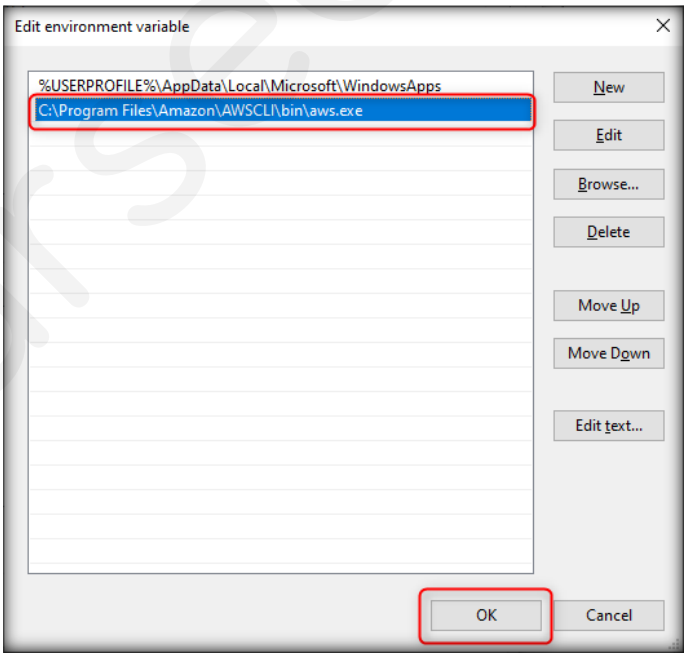


FIGURE 6.1.58: Finish Adding Environment Variable

61. Click **OK** again.

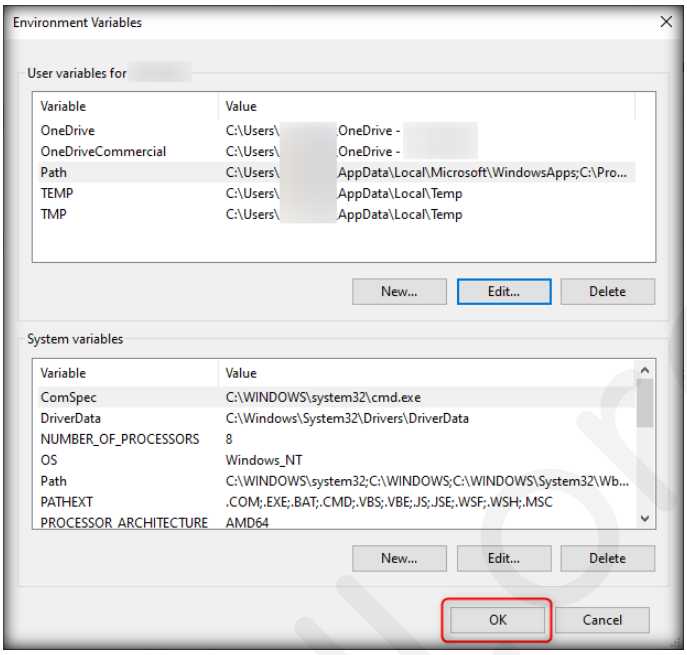


FIGURE 6.1.59: Click OK

62. Click **OK** again.

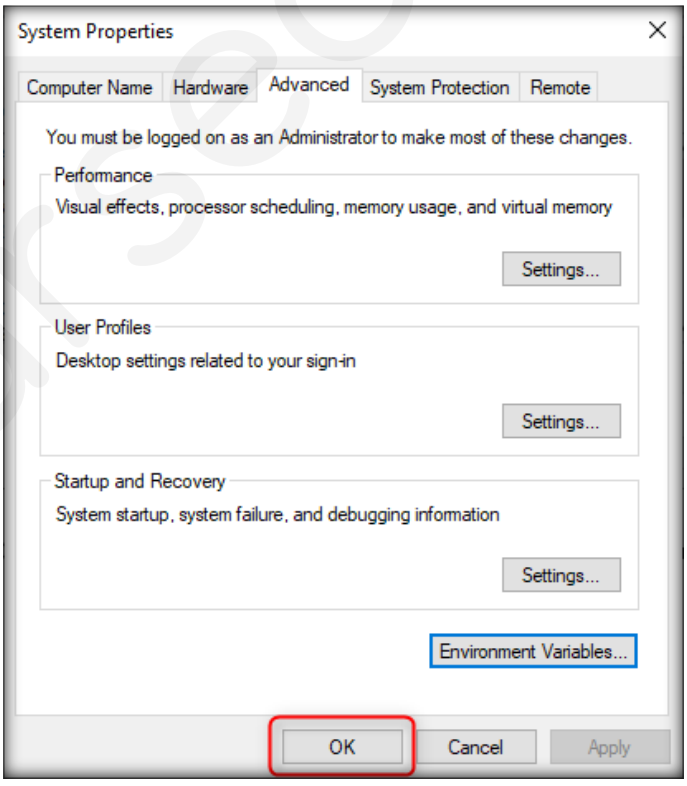


FIGURE 6.1.60: Click OK

TASK 4

Performing Penetration Testing

63. Close **Command Prompt** and launch it again.
64. After you have installed the AWS CLI, open the command prompt and type the following command to list the objects in the S3 bucket named `ccsedemobucket-1`.

```
aws --no-sign-request s3 ls
s3://ccsedemobucket-1
```

65. You will get an `Access Denied` output because the bucket has been secured and configured properly.

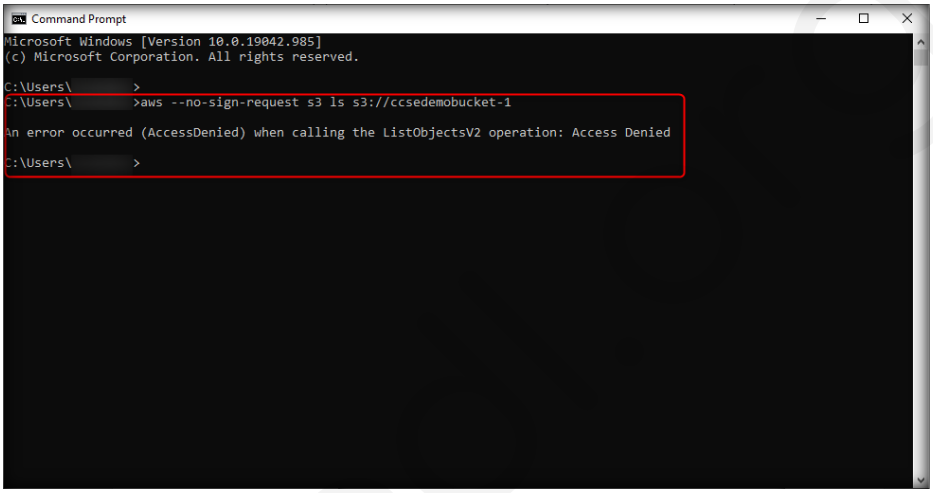


FIGURE 6.1.61: Listing Objects in S3 Bucket

66. Next, to get the access control list of the S3 bucket named `ccsedemobucket-1`, run the following command:

```
aws s3api get-bucket-acl --bucket
ccsedemobucket-1 --no-sign-request
```

67. You will get an `Access Denied` output because the bucket has been appropriately configured and secured.

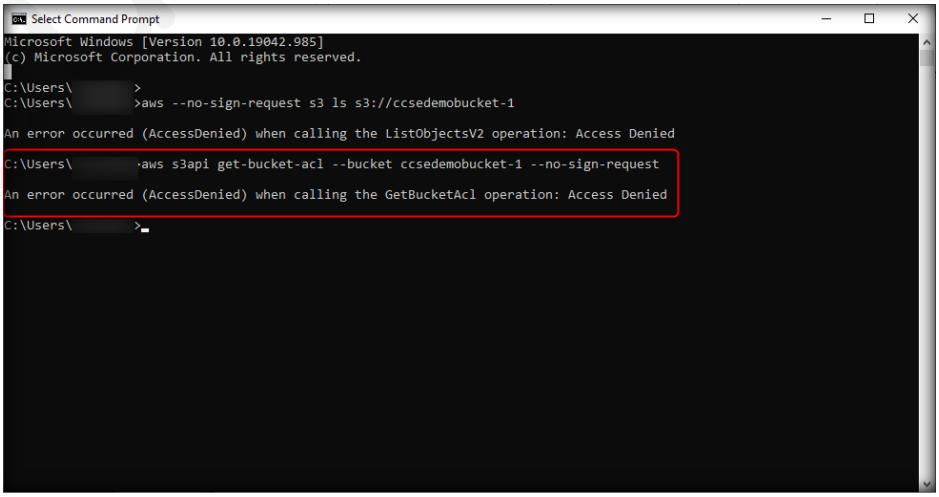


FIGURE 6.1.62: To Get ACL of S3 Bucket

68. Now, type the following command to perform a penetration testing on the second S3 bucket named `ccsedemobucket-2`:

```
aws --no-sign-request s3 ls
s3://ccsedemobucket-2
```

69. Upon running the command, you will get the text file object named `ccsedemo-2.txt` stored in the S3 bucket named `ccsedemobucket-2` because this bucket is misconfigured with public access.

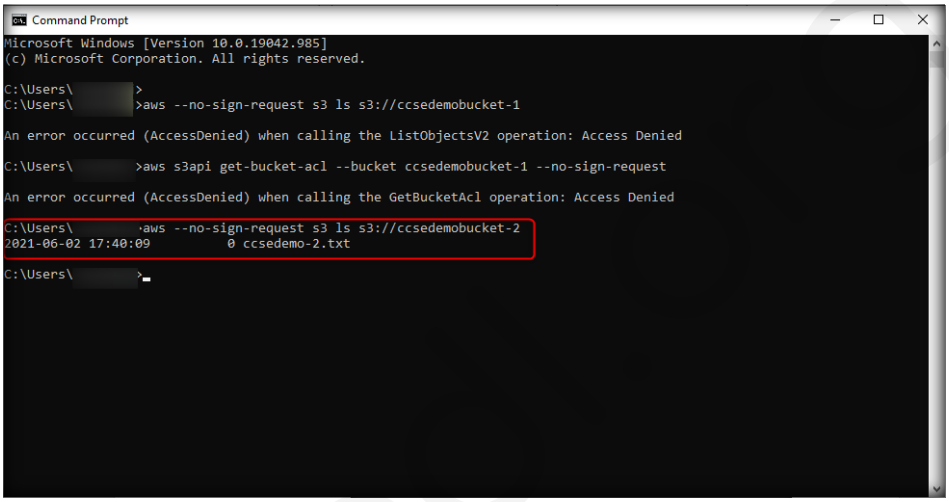


FIGURE 6.1.63: Listing Objects in the bucket

70. Now, to get the access control list of the second S3 bucket named `ccsedemobucket-2`, type the following command:

```
aws s3api get-bucket-acl --bucket
ccsedemobucket-2 --no-sign-request
```

71. You will get **FULL_CONTROL** as the permission output. This indicates **ccsedemobucket-2** is misconfigured and the information is publicly accessible.

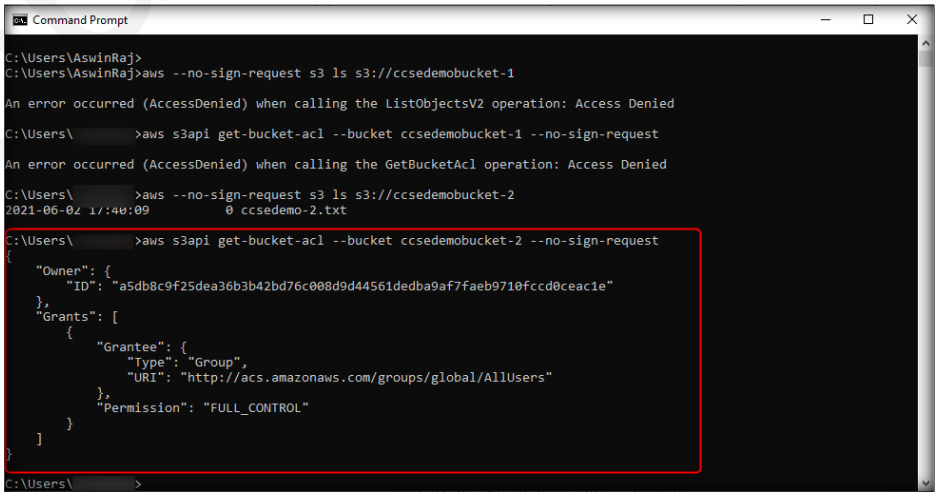


FIGURE 6.1.64: Getting ACL

72. Next, to prevent all users from writing to the **eccdemobucket-2** bucket, go back to **Edit access control list (ACL)** of **ccsedemobucket-2 (**where you were at **Step 45)** and uncheck the **Bucket ACL Write** permissions for **Everyone (public access)**.

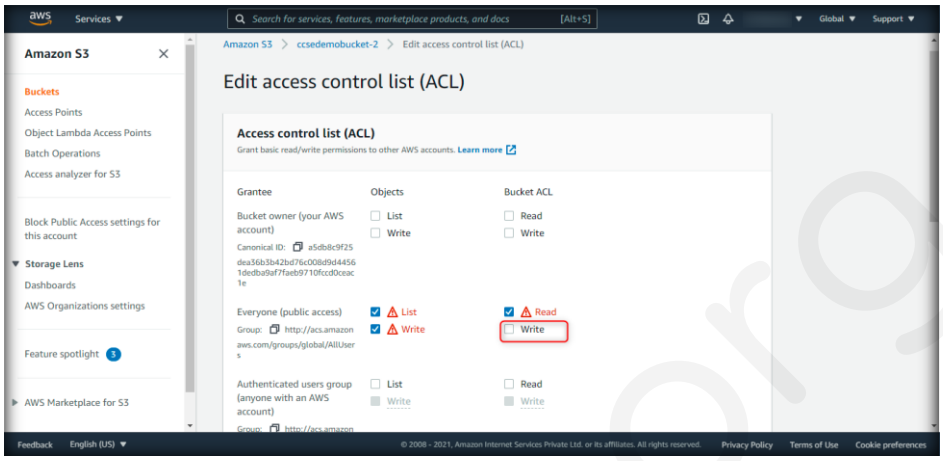


FIGURE 6.1.65: Disable Write Permissions for Bucket ACL

73. Scroll down and select the acknowledgement checkbox. Click on **Save changes**.

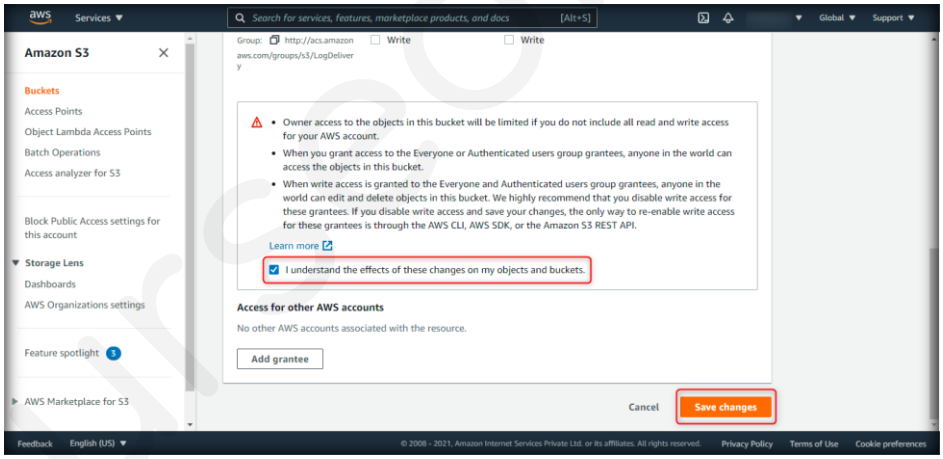


FIGURE 6.1.66: Disable Write Permissions for Bucket ACL

74. Now, go back to the **Command Prompt** window and type the following command again to check the **ACL**. Then, press **Enter**.

    ```
    aws s3api get-bucket-acl --bucket
    ccsedemobucket-2 --no-sign-request
    ```

    You will see that the permissions have changed from **Full Control** to `READ`, `WRITE`, and `READ_ACP`.

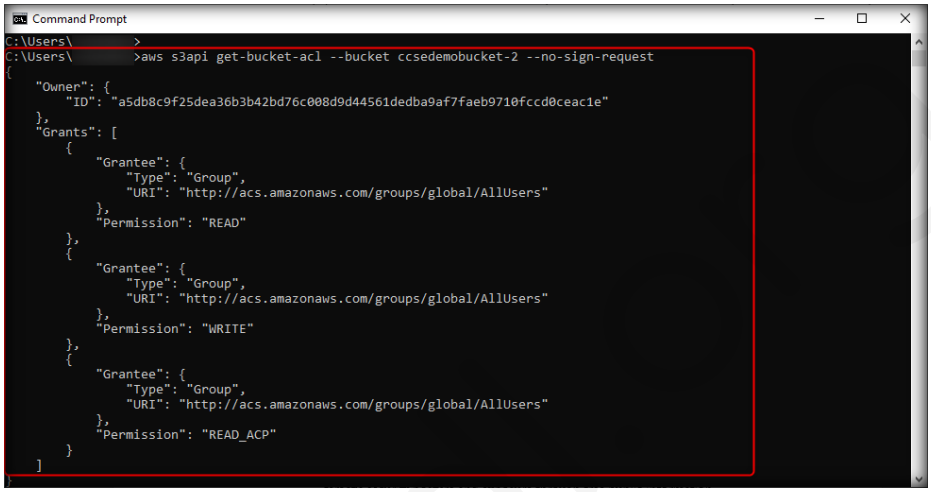


FIGURE 6.1.67: Checking ACL Permissions

75. As described above, a cloud security engineer can perform penetration testing on AWS S3 buckets to identify misconfigured buckets and take necessary actions to secure them.

Caution: Ensure you delete, shut down, or terminate all resources created and used in this lab to prevent their billing.

76. Now delete the instances created in this lab. Navigate to the **Buckets** under Amazon S3 and select the checkbox of the bucket created in this lab (here. **ccse-demobucket-1**). Click on the **Empty** button to empty the bucket before deleting it. In the pop up window, enter permanently delete and click on **Empty**.

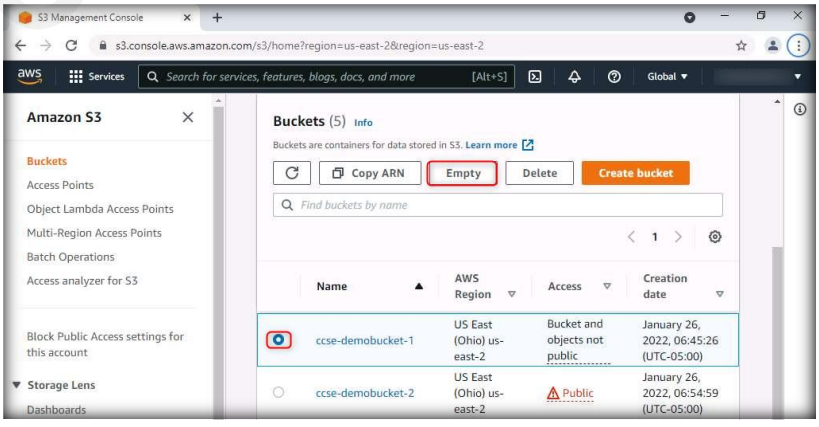


FIGURE 6.1.68: Emptying bucket

77. Navigate to the Buckets under S3 and select the checkbox of the bucket created in this lab (here. **ccse-demo bucket-1**). Click on the **Delete** button. In the pop-up window, enter the bucket's name and click on **Delete bucket**.

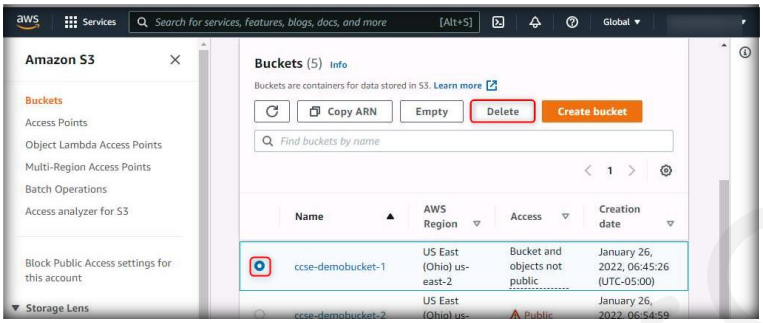


FIGURE 6.1.69: Deleting bucket

78. Navigate to the **Buckets** under Amazon S3 and select the checkbox of the bucket created in this lab (here. **ccse-demobucket-2**). Click on **Empty** button to empty the bucket before deleting. In the pop up window, enter permanently delete and click on **Empty**.

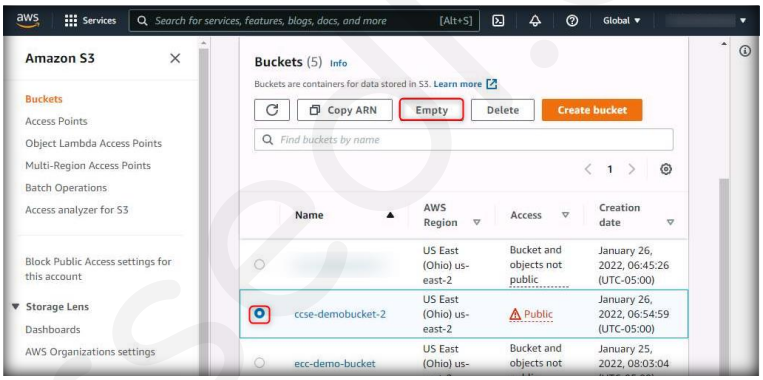


FIGURE 6.1.70: Emptying bucket

79. Navigate to the Buckets under S3 and select the checkbox of the bucket created in this lab (here. **ccse-demo bucket-2**). Click on the **Delete** button. In the pop-up window, enter the bucket's name and click on **Delete bucket**.

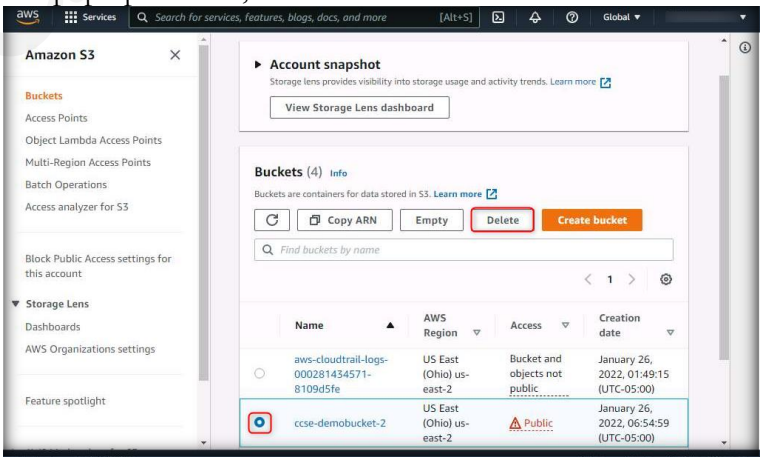


FIGURE 6.1.71: Deleting bucket

Lab Analysis

Analyze and document the results of this lab exercise. Provide your opinion on your target's security posture and exposure through free public information.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS ABOUT THIS LAB.

Lab 2

Identifying Publicly Accessible Data with Compromised AWS API Keys

AWS IAM access keys consist of an access key ID and a secret access key that act as long-term credentials for making programmatic calls to AWS CLI.

Lab Scenario

AWS identity and access management (IAM) services can be used for secure access control of resources. The access keys consist of an access key ID and a secret access key and are long-term credentials for making programmatic calls to AWS CLI or AWS API. A secret access key that is included with the API keys is only available for the time at which the cloud administrator created it. This secret access key needs to be stored securely. If the secret keys are lost or if a cloud administrator feels they are stolen, new access keys will have to be created. A cloud security engineer should gather information about the organization's AWS resources that would be easily available if the access keys fall into the hands of a turncloak, pawn, or imposter.

KEY ICON

- Valuable information
- Test your knowledge
- Web exercise
- Workbook review

Lab Objectives

This lab will demonstrate how a pentester will gather information about an organization's AWS resources that would be easily available if the access keys are compromised.

In this lab, you will learn to do the following:

- Create an AWS IAM user
- Create a user group for the IAM user
- Create a security group
- Perform penetration testing to identify compromised IAM access keys

Lab Environment

To perform this lab, you need the following:

- **Admin Machine** VM
- **Administrative**
- Registered AWS account

Lab Duration

Time: 20 minutes

Overview of IAM access keys

AWS IAM services allow a cloud security engineer to implement access control to their resources. Programmatic access to the AWS APIs can be implemented using AWS access keys to confirm user identity. The access keys consist of an access key ID and a secret access key. The secret access keys are only available at the time of their creation. Thus, the secret access keys should be stored securely. In an organization, these access keys will be shared with the users. If they are accidentally leaked from the user or go into the hands of a turncloak or an imposter, certain information on the organization's AWS resources can be easily accessed by them. Therefore, if the secret keys are lost or stolen, the corresponding user should be deleted and a new user should be created.

Lab Tasks

Note: Web applications in a cloud environment may undergo frequent updates. As we are working on a cloud-based environment for this lab (i.e., AWS), the application interface may be updated with time. Hence, in case you happen to work on an updated version of AWS, the user interface you see on the application might differ from what you see in the lab. Consequently, the steps and screenshots demonstrated in this lab might also differ.

Note: Before starting this lab, you should create an AWS account using the following link: https://portal.aws.amazon.com/billing/signup. Once the registration is complete, perform the following tasks.

Note: You can also use any existing AWS account but be aware that it may incur significant charges to your account.

TASK 1

Creating AWS IAM User

1. Log in to the **Admin Machine** VM with the user **Admin** and password **admin@123**.

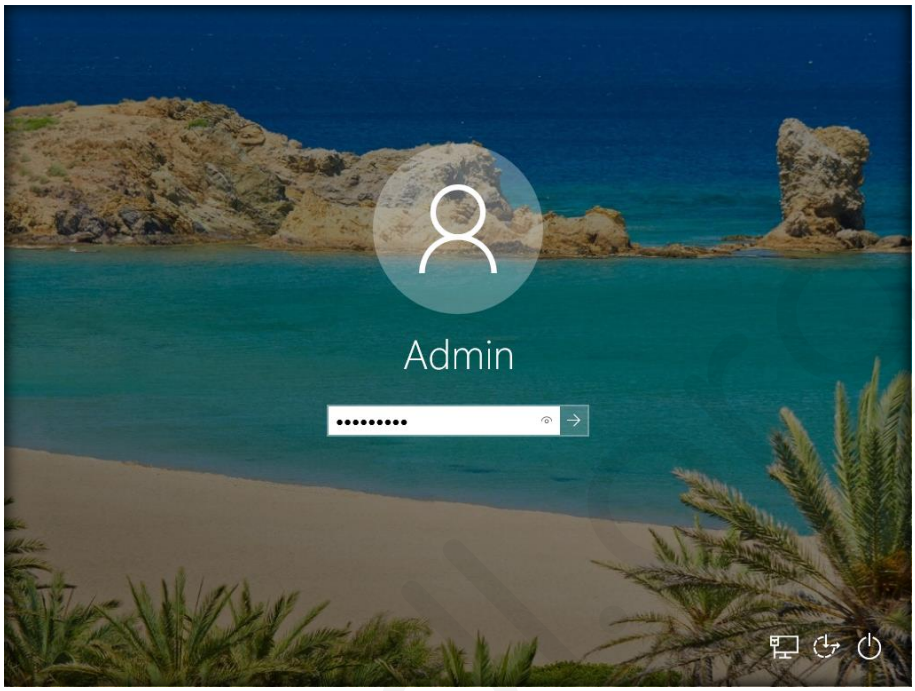


FIGURE 6.2.1: User Login

2. To open the browser, double-click on the **Google Chrome** icon on the desktop.

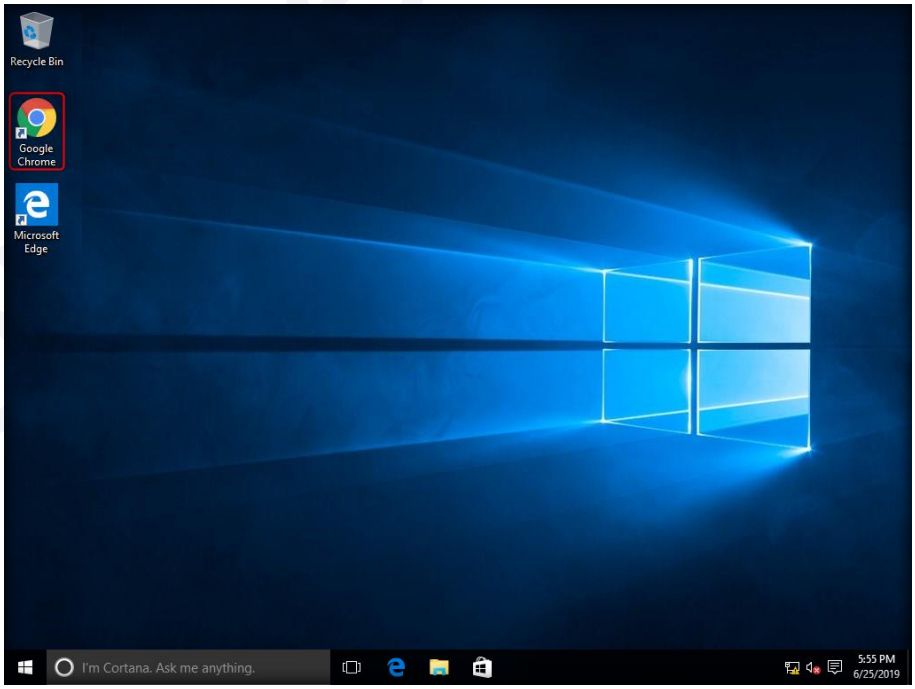


FIGURE 6.2.2: Navigating to the Chrome Browser from Taskbar

3. The **Google Chrome** browser opens. Go to the address bar, type **https://aws.amazon.com/**, and press **Enter.**

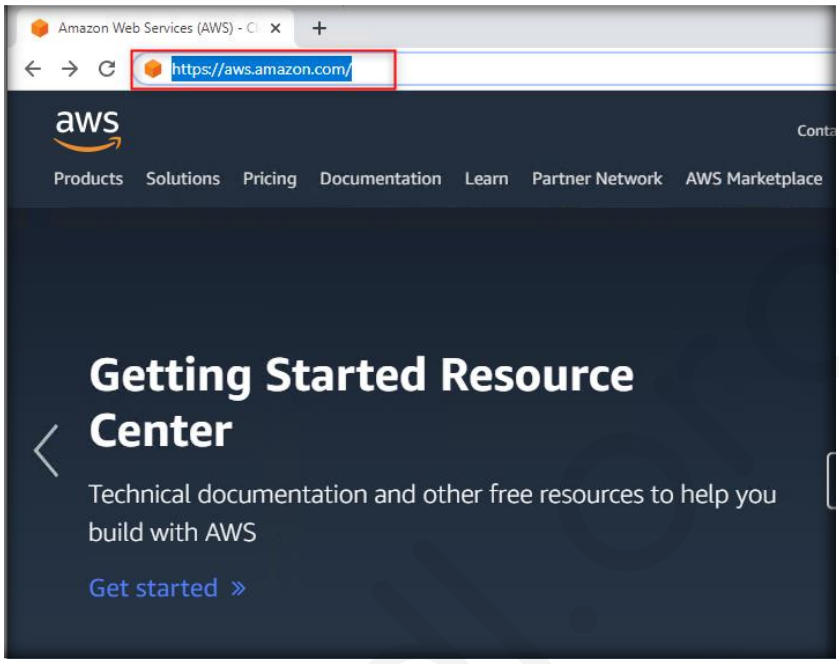


FIGURE 6.2.3: Access AWS in Browser

Note: If you are already logged in, skip the login steps.

4. The **AWS Web Services - Cloud Computing Services** page appears. Click on the **AWS Management Console** from the **My Account** dropdown, as shown in the screenshot below.

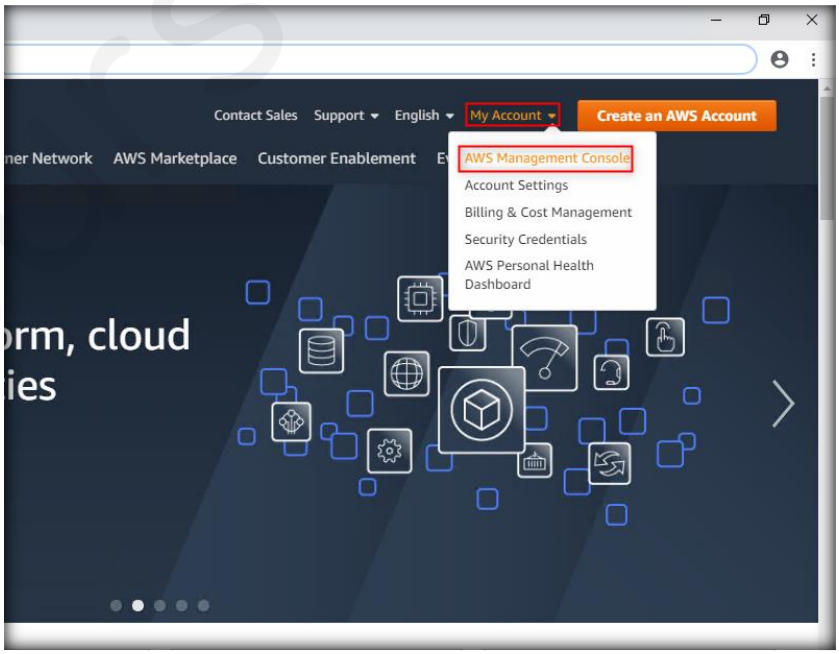


FIGURE 6.2.4: Navigate to AWS Management Console

5. The **AWS Sign-in** page appears. Choose **Root user** and type the AWS administrator account ID (Root user email address), as shown in the screenshot below, and click on **Next**.

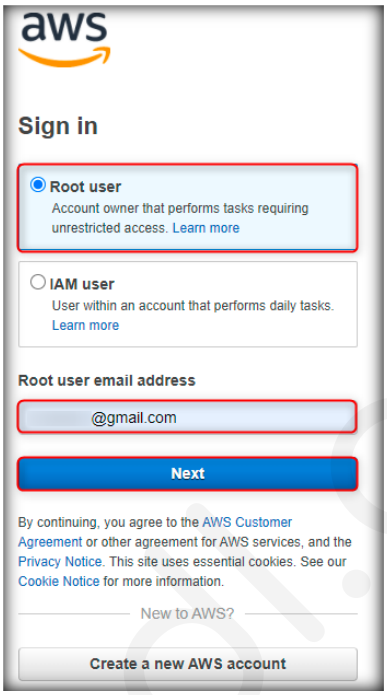


FIGURE 6.2.5: Login Credentials

6. In the security check page that appears, type the characters shown in the image and click **Submit**.

FIGURE 6.2.6: AWS Security Check page

7. In the **Password** field, type the password, and click on **Sign in**, as shown in the screenshot below.

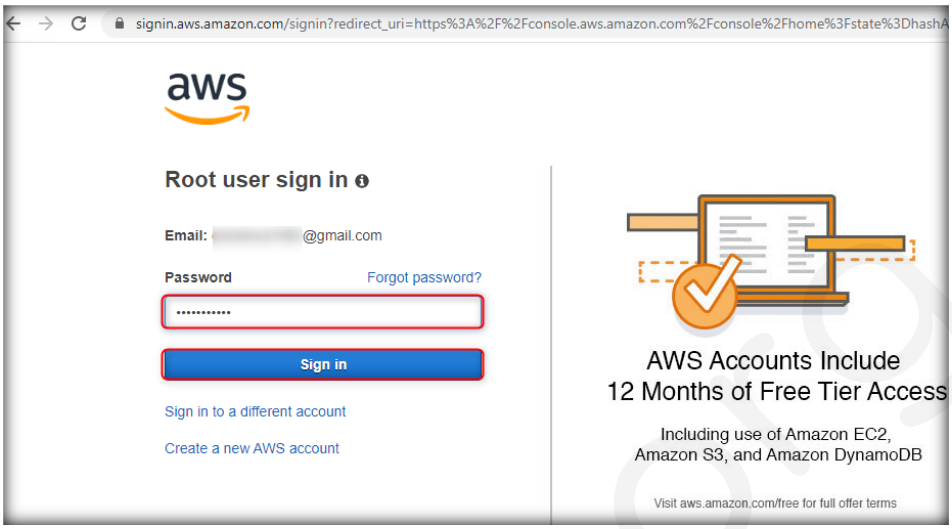


FIGURE 6.2.7: Sign-in to AWS

8. Click on **Services**. Then, scroll down and click on **IAM** under **Security, Identity, & Compliance** under **All services**.

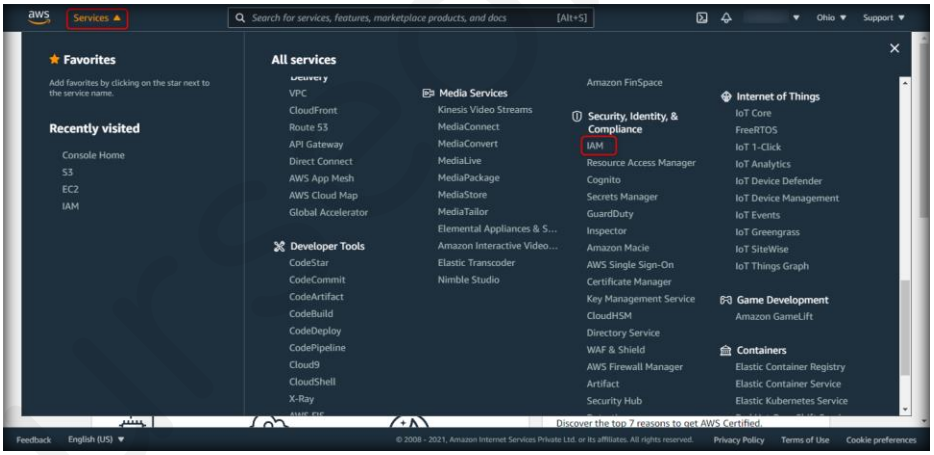


FIGURE 6.2.8: Navigating to IAM

9. In the **IAM dashboard,** click on **Users** on the left pane under **Access management**.

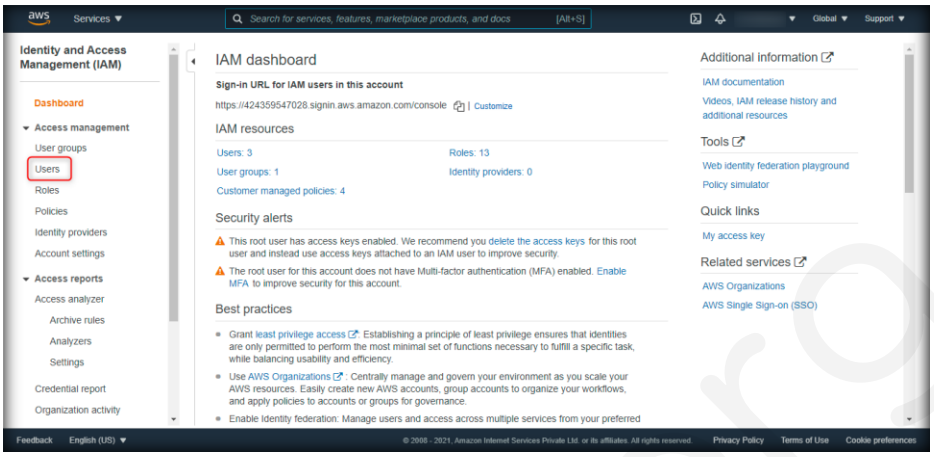


FIGURE 6.2.9: IAM Dashboard

10. Click on **Add user** at the top.

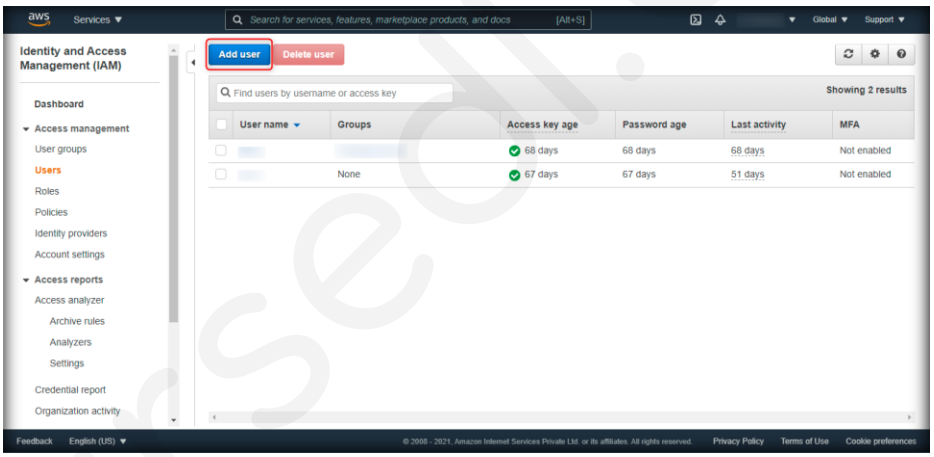


FIGURE 6.2.10: Adding User

11. In the **Add user** window that opens, configure the following, and click on **Next: Permissions** at the bottom.

    User name: **ccseuser**

    Access type: **Programmatic access**

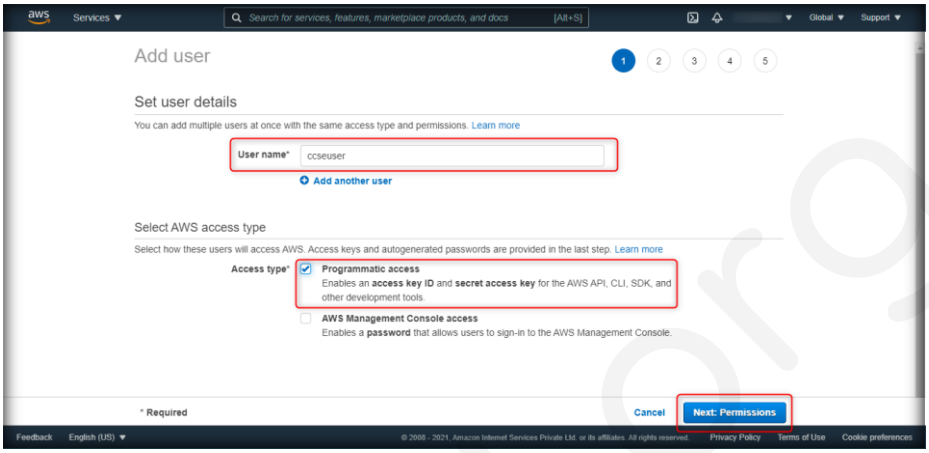


FIGURE 6.2.11: Configuring New User

12. In the next window that appears, under **Set permissions**, click on the third option: **Attach existing policies directly**.

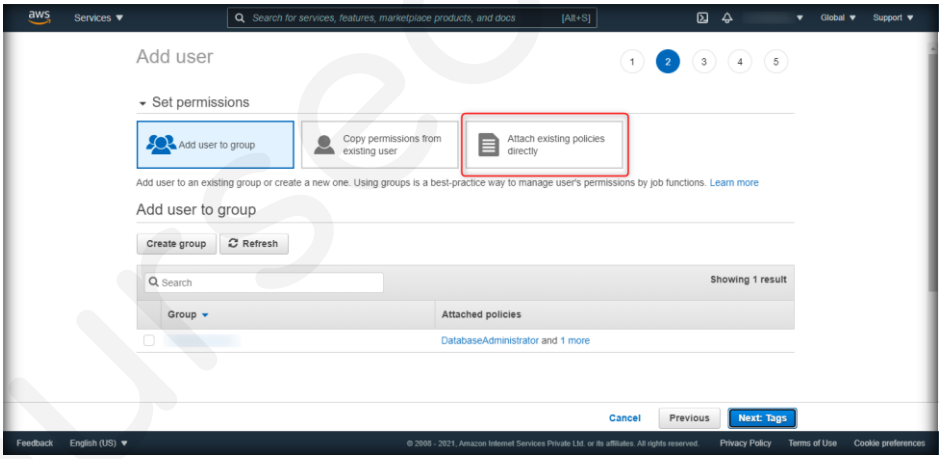


FIGURE 6.2.12: Add Labels

13. In the **Filter policies** search box, type **AmazonEC2FullAccess**. Then, select the checkbox for **AmazonEC2FullAccess**.

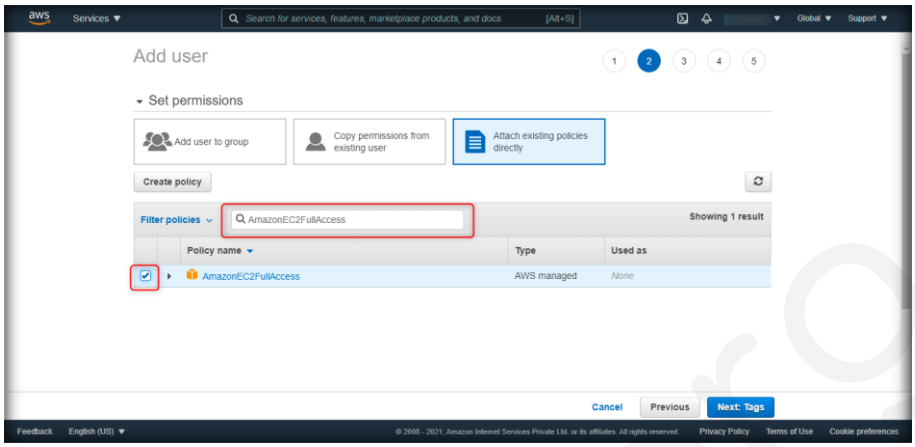


FIGURE 6.2.13: Configuring Policy

14. Click on **Next: Tags** at the bottom.

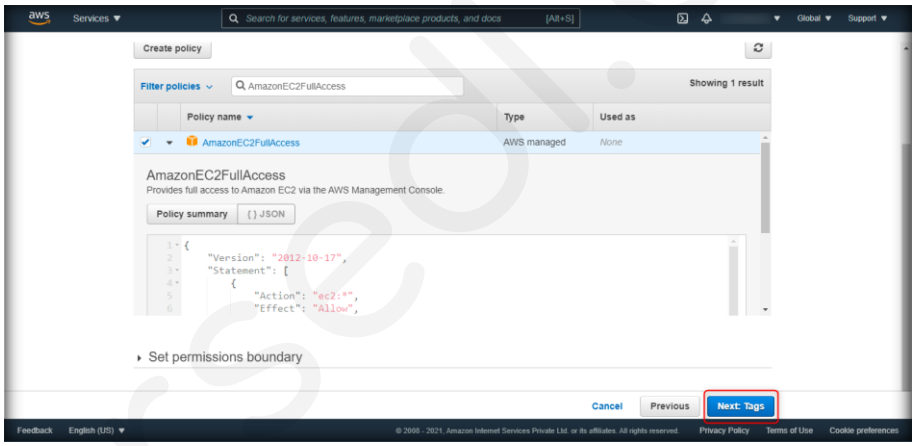


FIGURE 6.2.14: Moving Next

15. In the **Add tags** window, do not configure anything. Click on **Next: Review**.

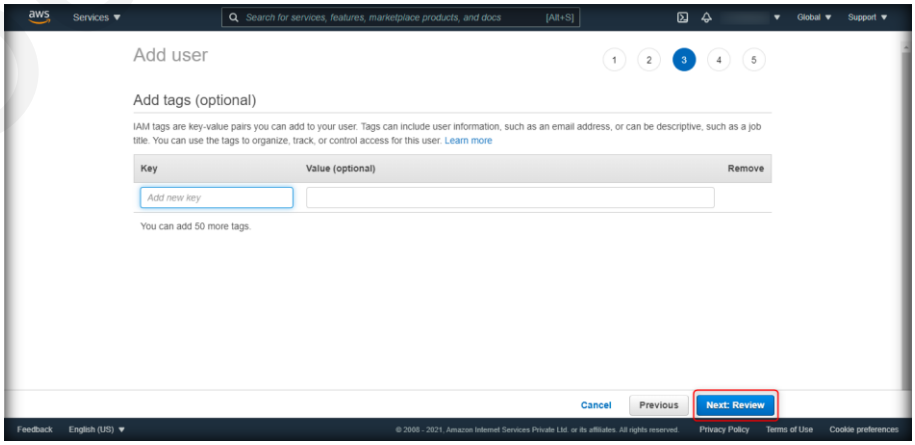


FIGURE 6.2.15: Key Policy

16. A **Review** window appears with the configured settings. Verify the configuration and click on **Create user**.

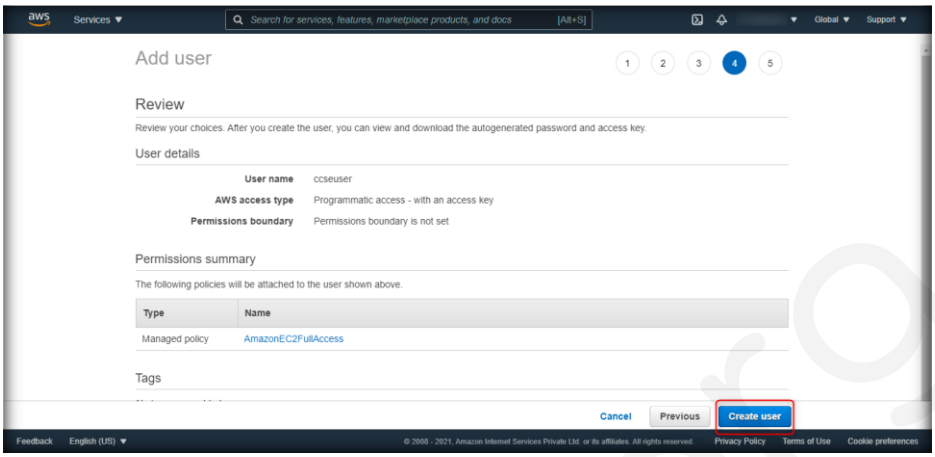


FIGURE 6.2.16: Creating the user

17. A user named **ccseuser** will be successfully created now. Note the **Access key ID** for **ccseuser**. Under **Secret access key**, click on **Show** to get the secret access key. Note the secret access key and click on **Close** at the bottom.

    Note: Do not close the **User created** window before noting the **Secret access key**, as it cannot be accessed later.

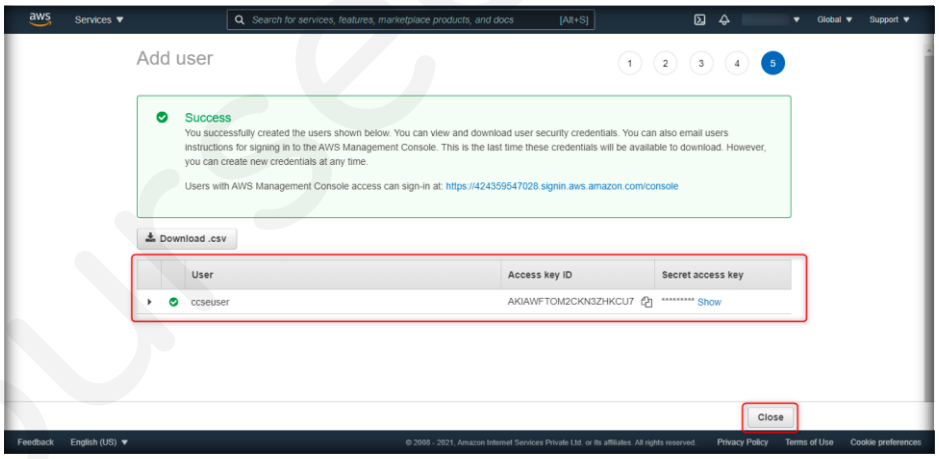


FIGURE 6.2.17: Viewing access key

TASK 2

Creating user group for the IAM user

18. Now, you can add **ccseuser** to a group. In the **IAM Dashboard** window, click on **User groups** under **Access management** in the left pane.

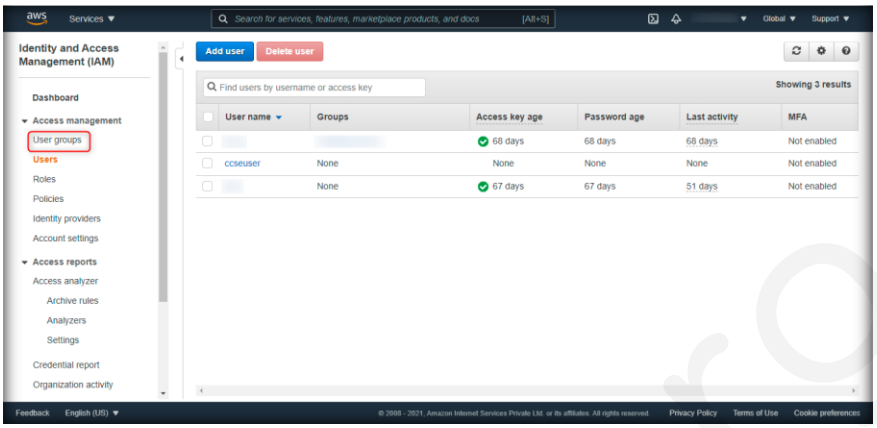


FIGURE 6.2.18: Creating User group

19. Click on **Create group** at the top right.

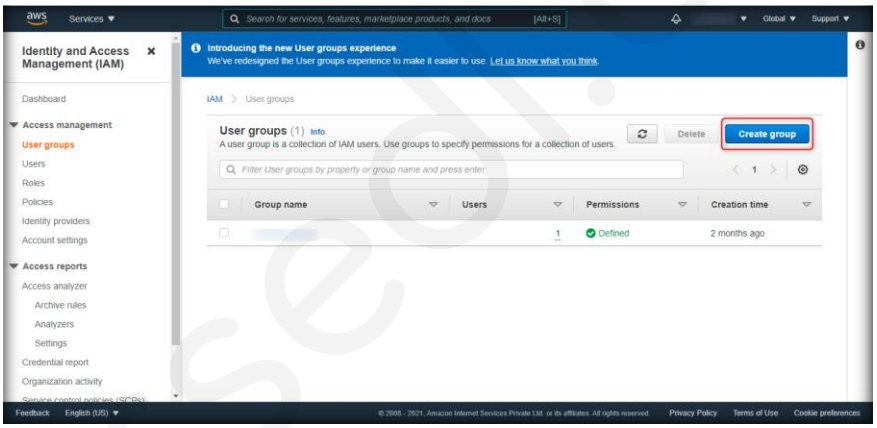


FIGURE 6.2.19: Creating Group

20. In the **User group name** field under the **Create user group** window**,** enter **CloudSecurityEngineer**.

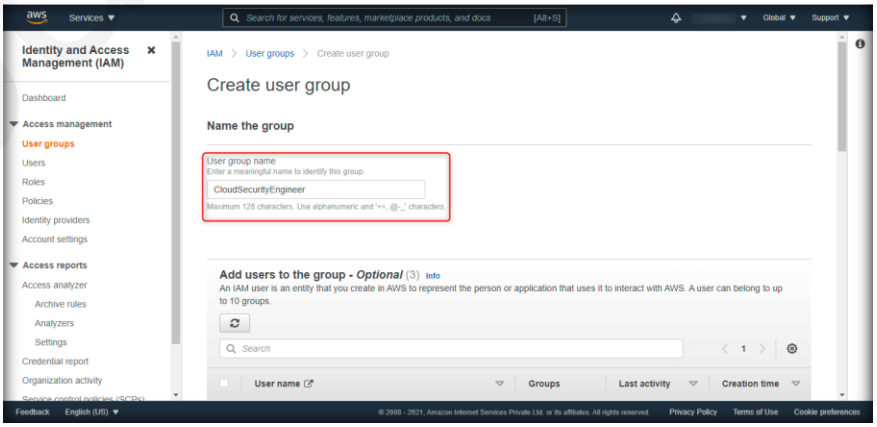


FIGURE 6.2.20: Naming User Group

21. Scroll down and select **ccseuser** under **Add users to the group**.

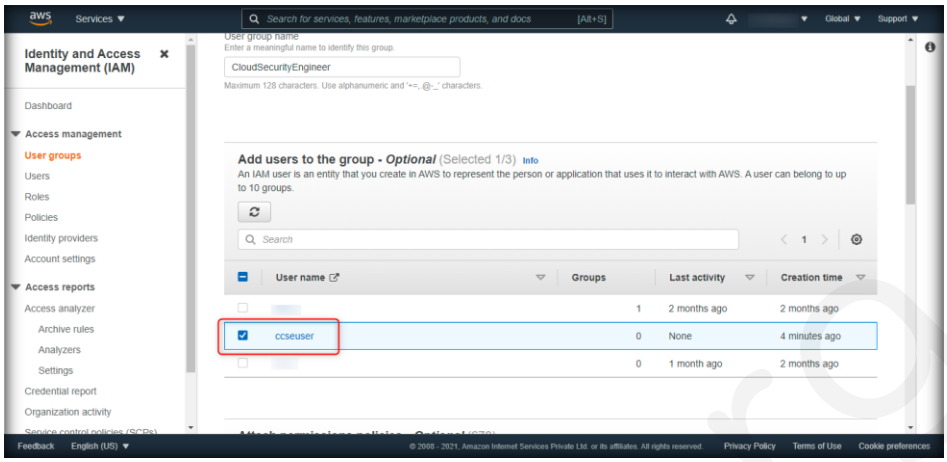


FIGURE 6.2.21: Selecting User

22. Scroll down and search for **IAMReadOnlyAccess** under **Attach permissions policies** and press **Enter**.

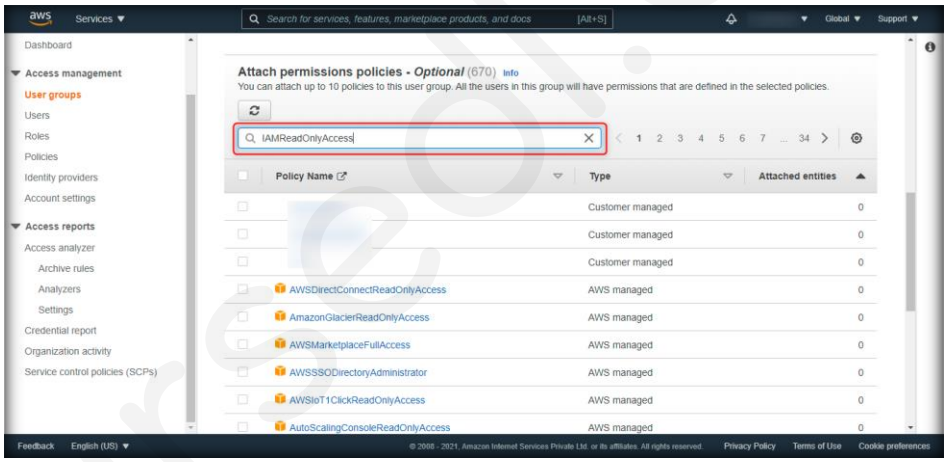


FIGURE 6.2.22: Configuring permissions

23. Select the checkbox for **IAMReadOnlyAccess** and click on **Create group**.

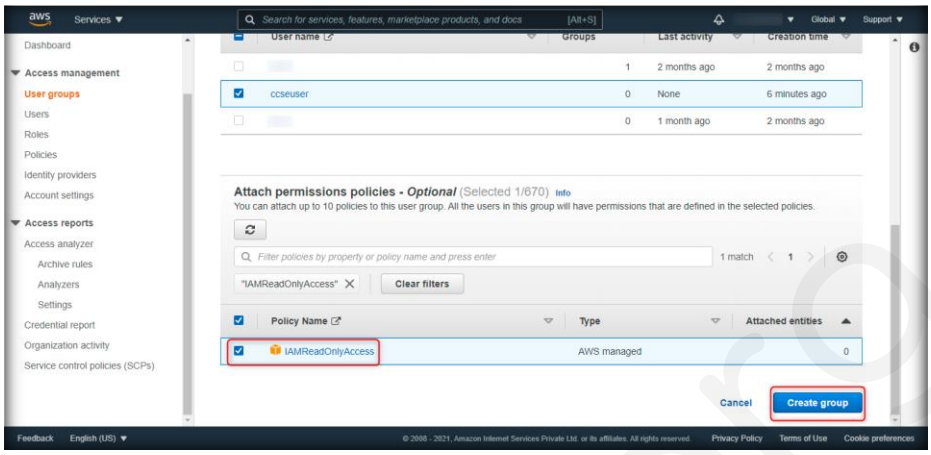


FIGURE 6.2.23: Creating Group

24. The **CloudSecurityEngineer** group will be successfully created.

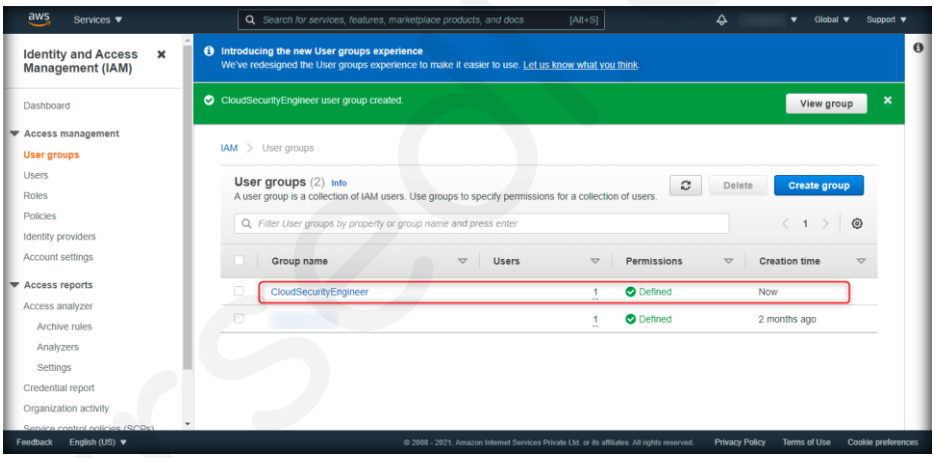


FIGURE 6.2.24: Group Created

25. Click on **Users** in the left pane.

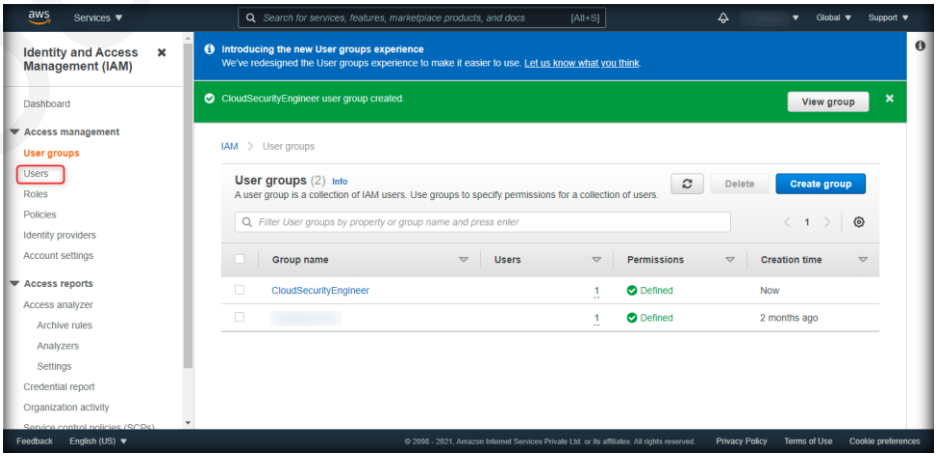


FIGURE 6.2.25: Select Users

26. You will see that **ccseuser** comes under **CloudSecurityEngineer**.

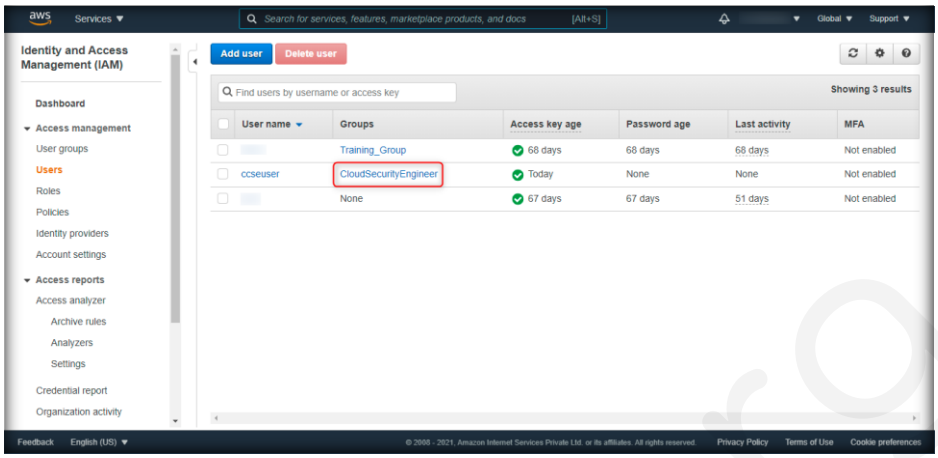


FIGURE 6.2.26: User is added to Group

TASK 3

Creating Security Groups

27. Now, you will create a **Security Group.** Click on **Services** at the top of the AWS console and select **EC2** under **Compute Engine**.

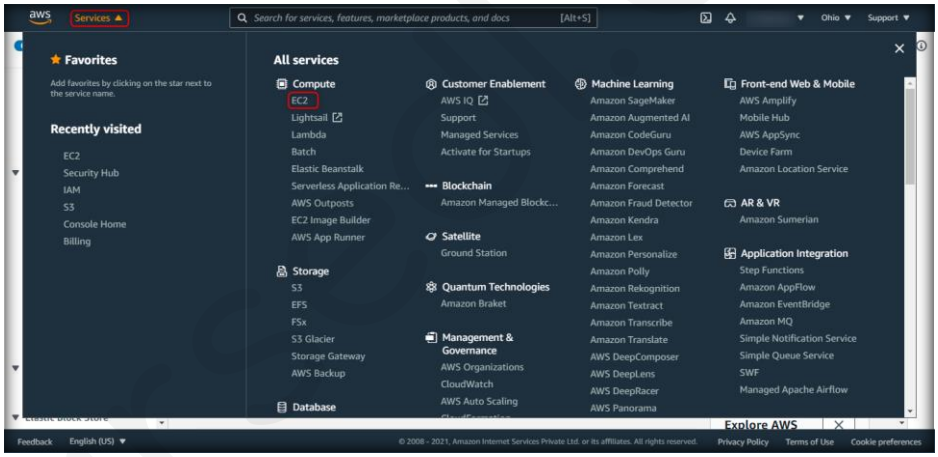


FIGURE 6.2.27: Navigate to EC2

28. In the left pane of the EC2 console, click on **Security Groups** under **Network & Security**.

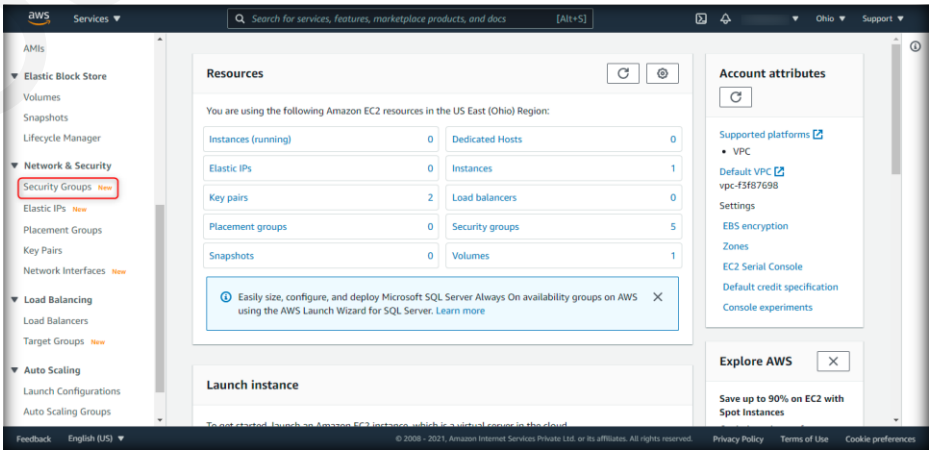


FIGURE 6.2.28: Select Security Groups

29. Click on **Create security group** at the top right of the EC2 console.

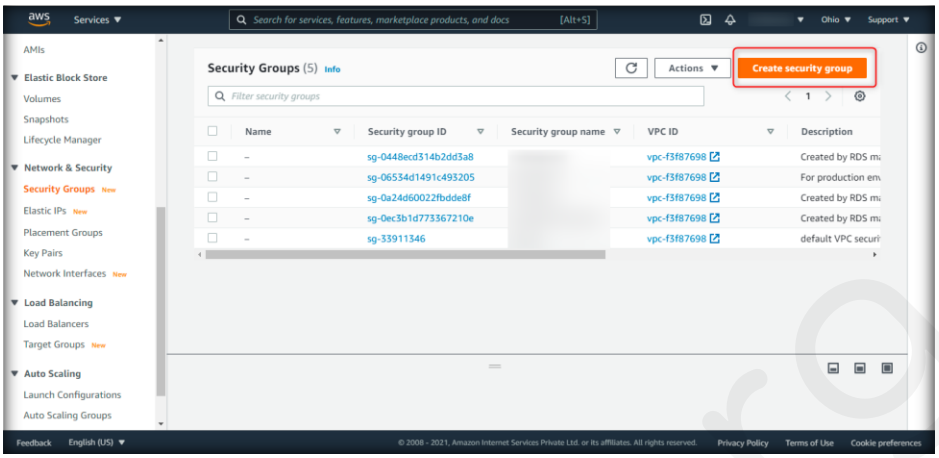


FIGURE 6.2.29: Create Security Groups

30. In the **Create security group** window that appears, configure the following under **Basic details**:

Security group name: ccsesecuritygroup

Description: Allow access to ccse users

VPC: Default value

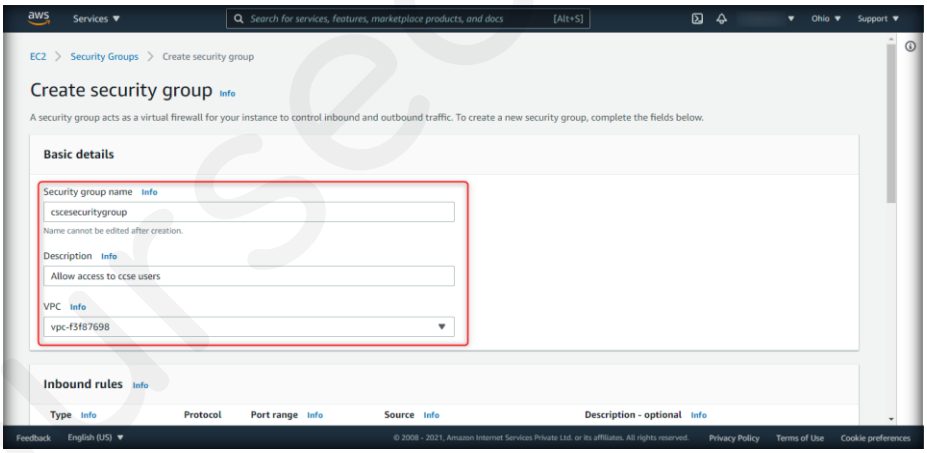


FIGURE 6.2.30: Configuring Security Groups

31. Scroll down and click on **Add rule** under **Inbound rules**.

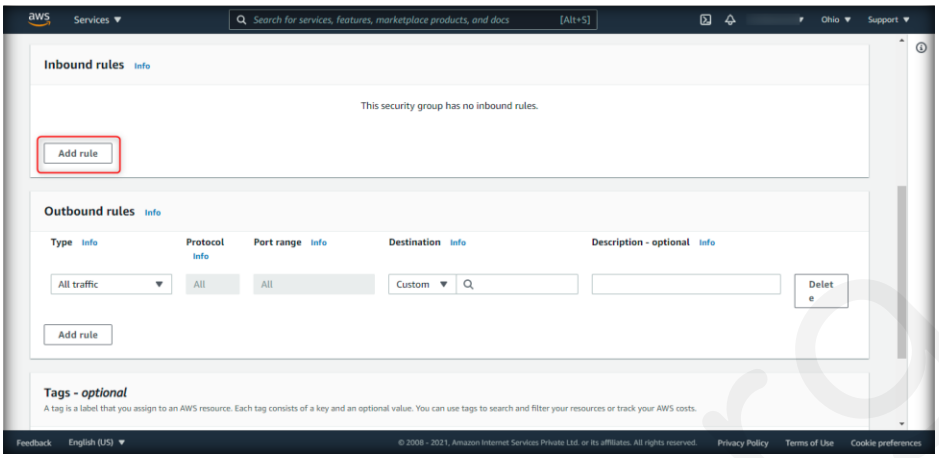


FIGURE 6.2.31: Adding Inbound rule

32. Configure the following for **Inbound rules**:

    Type: SSH

    Source: 0.0.0.0/0

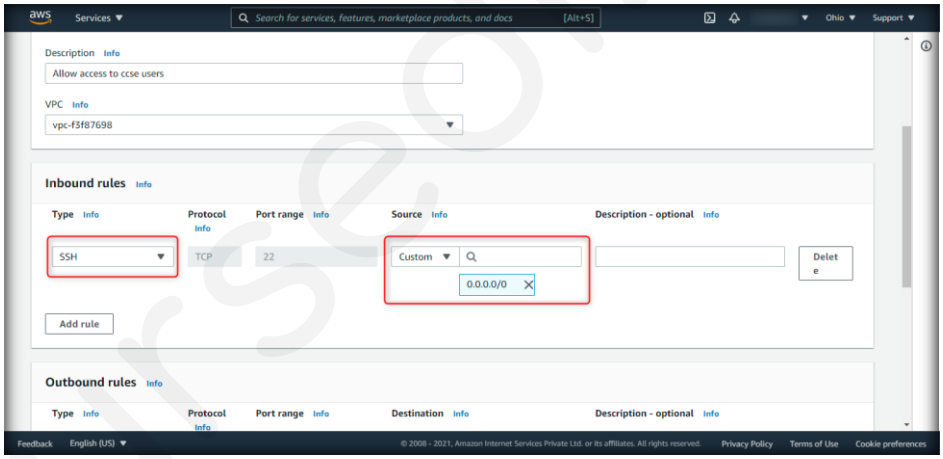


FIGURE 6.2.32: Configuring Inbound rule

33. For **Outbound rules**, configure the following:

    Type: All traffic

    Destination: 0.0.0.0/0

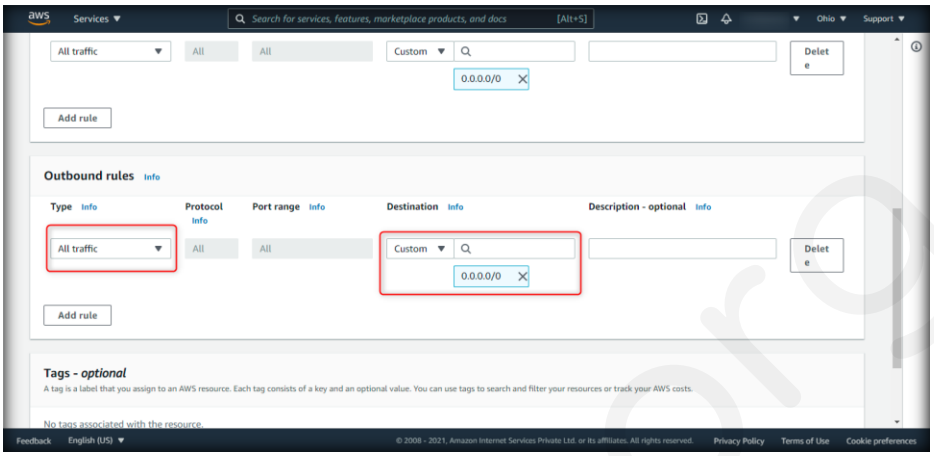


FIGURE 6.2.33: Configuring outbound rule

34. Scroll down and click on **Create security group**.

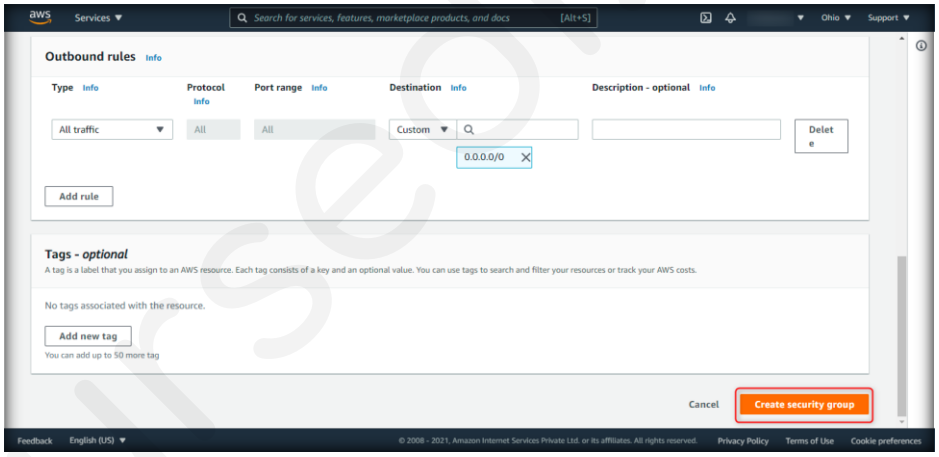


FIGURE 6.2.34: Creating Security group

35. The security group has been created successfully. Note the **Security group ID**.

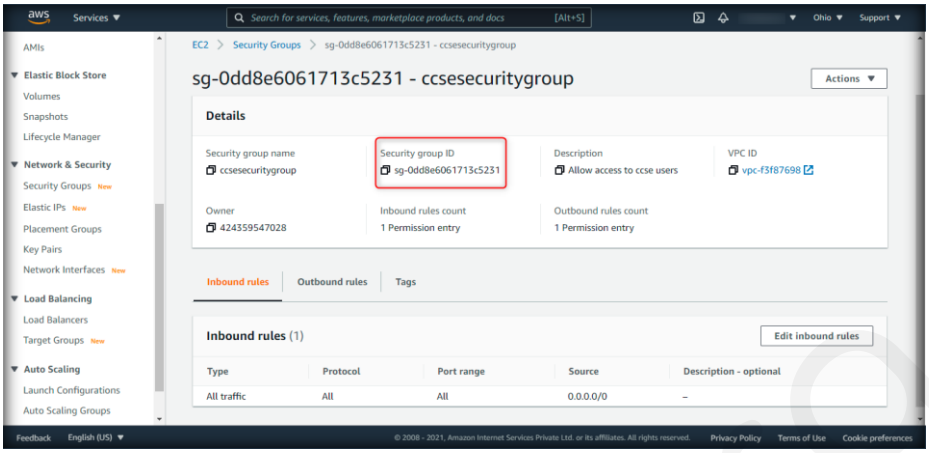


FIGURE 6.2.35: Security Group Created

TASK 4

Performing Penetration Testing with access keys

36. Now, assume the access keys are stolen or used by an insider for performing malicious activities. You, as a cloud security engineer, can collect the information about the AWS resources that are leaked when these access keys are misused.
37. To do this, you need to install AWS CLI in your Windows VM machine, which can be found in the previous lab on identifying misconfigured S3 buckets. If it is already installed, you can continue with the next step.
38. Search for **cmd** in the **Search** section at the bottom task bar of the Windows VM and press **Enter**.

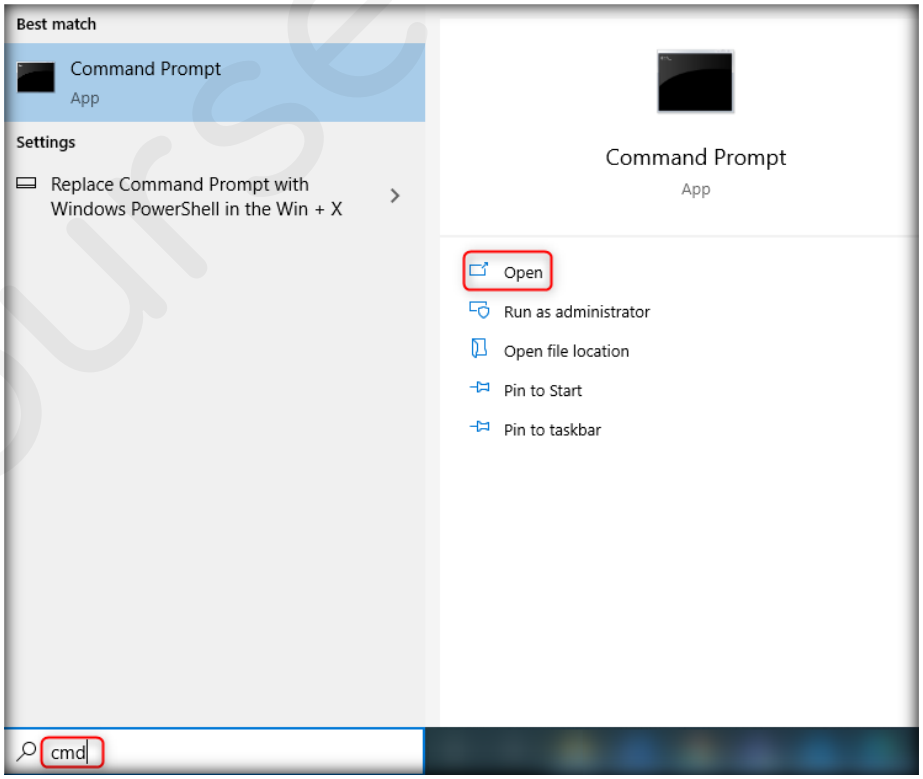


FIGURE 6.2.36: Opening cmd

39. In the command prompt window, type the following command to ensure AWS CLI is installed successfully. Then, press **Enter**.

    ```
    aws --version
    ```

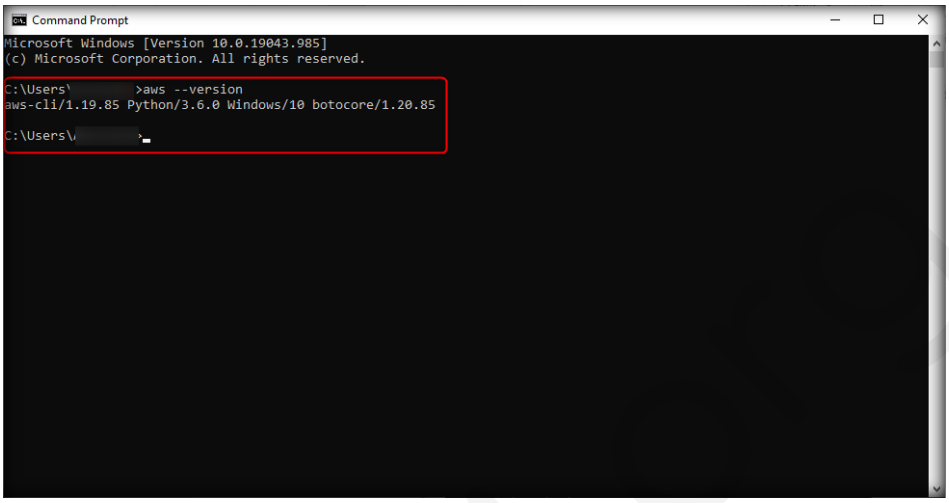


FIGURE 6.2.37: Verifying AWS CLI is installed

Note: IF AWS CLI is not installed, follow the **Steps 47–52** of Lab1 in this module to install AWS CLI before proceeding to the next step.

40. Now, type the following command and press **Enter** to add compromised user credentials to AWS CLI. This stores the compromised access key credentials in the profile named `ccsetest`.

    ```
    aws configure --profile ccsetest
    ```

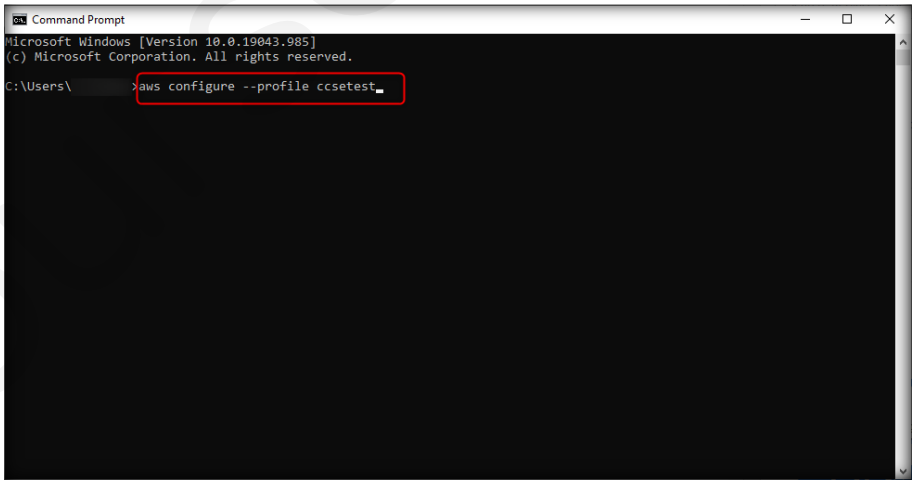


FIGURE 6.2.38: Storing credentials

41. When running this command, you will have to enter certain values. For each value, configure the following and press **Enter**.

    ```
    AWS Access Key ID [None]: <access_key_ID>
    ```

    ```
    AWS Secret Access Key [None]:
    <secret_access_key>
    ```

    ```
    Default region name [None]: <default_region>
    ```

    ```
    Default output format [None]: json
    ```

    (Here, replace `<access_key_ID>` with the access key ID of ccseuser you noted after creating the user, `<secret_access_key>` with the secret access key of ccseuser you noted, and `<default_region>` with the name of your default region.)

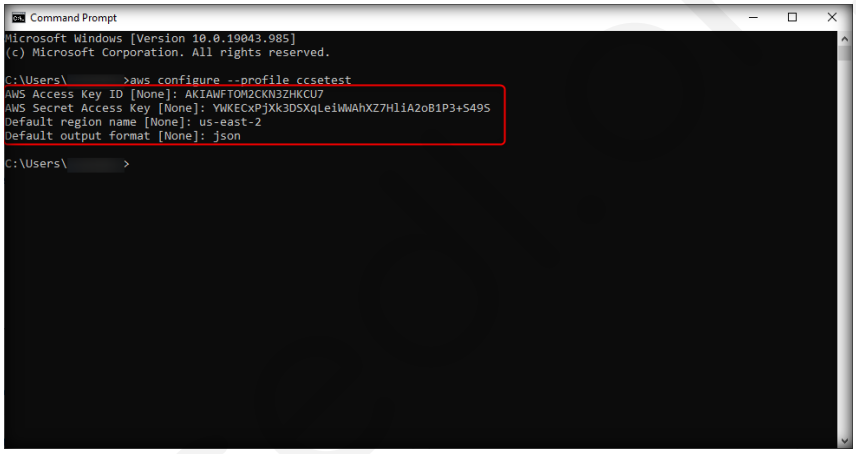


FIGURE 6.2.39: Creating ccsetest profile

42. Now, type the following **Security Token Service (STS)** command to gather information about the leaked keys and press **Enter**.

    ```
    aws sts get-caller-identity --profile ccsetest
    ```

43. The **STS: Get Caller Identity** command gathers the user's **User ID**, **Account ID,** and **Amazon Resource Name (ARN)**. Here, the user is named **ccseuser**.

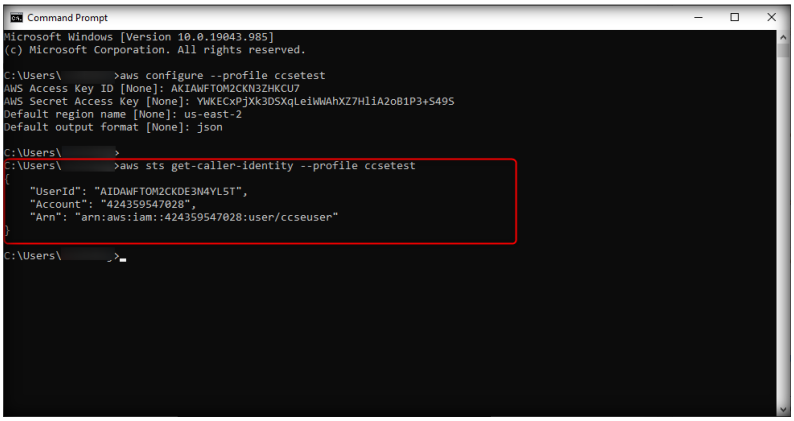


FIGURE 6.2.40: Running STS command

44. Type the following command to list the instances in the region (here, the region is **us-east-2**) and press **Enter**.

    ```
    aws ec2 describe-security-groups --region us-east-2 --profile ccsetest
    ```

45. You will get the instances configured in the region **us-east-2**. Here, you have an instance that was previously created for this lab. This information is available because **ccseuser** has the **AmazonEC2FullAccess policy**.

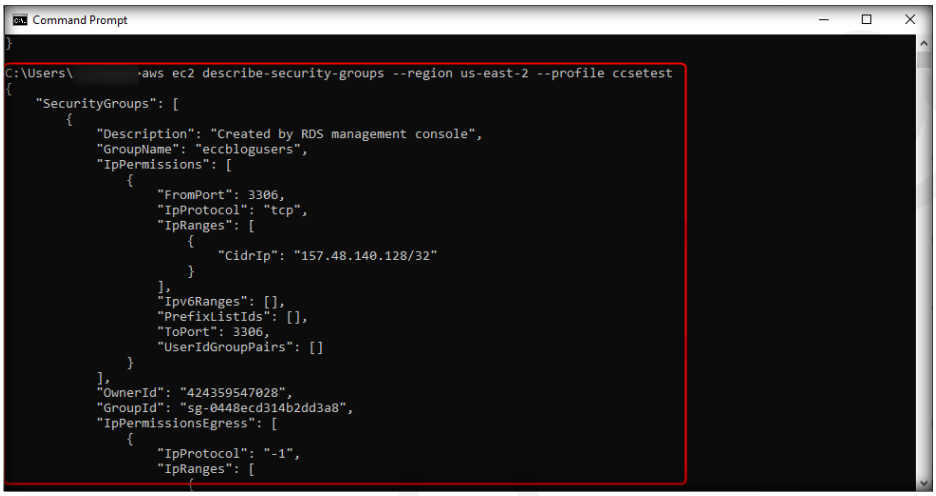


FIGURE 6.2.41: Listing Instances

46. Now, if you type the following command to list the instances of a region that has no instances, you will get an output that follows the command. Here, there is no instance in **us-east-1**.

    ```
    aws ec2 describe-instances --region us-east-1 --profile ccsetest
    ```

    The output will be

    ```
    {
    "Reservations": []
    }
    ```

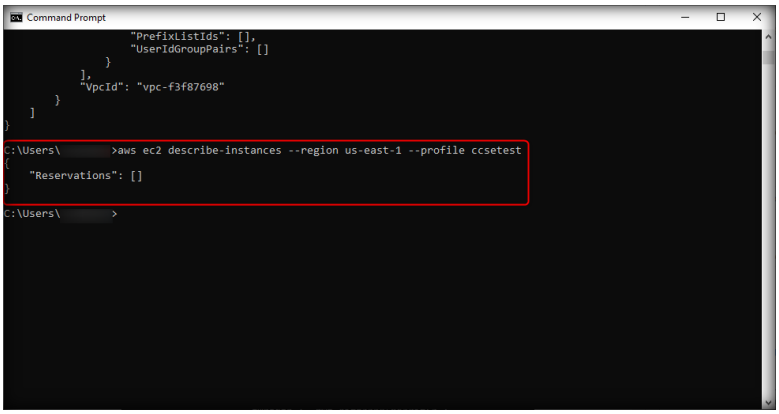


FIGURE 6.2.42: Listing Instances of regions with no instances configured

47. Now, type the following command to get information about the security group and press **Enter**.

    ```
    aws ec2 describe-security-groups --region us-east-2 --profile ccsetest
    ```

48. You will get the information regarding the configured security groups.

    Note: The security groups listed in this screenshot have been created for another lab.

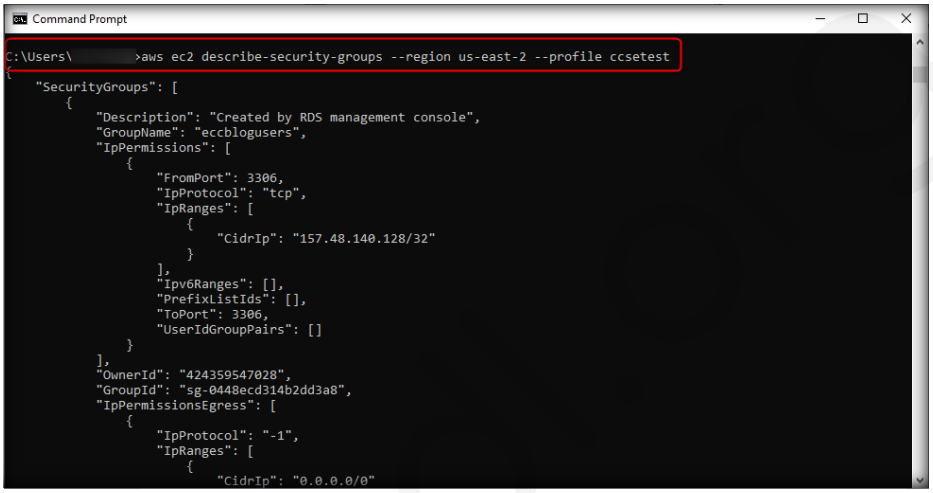


FIGURE 6.2.43: Gathering Security Groups Information

49. If you scroll down for the security groups information, you will get the details of **ccsesecuritygroup,** including the firewall rules.

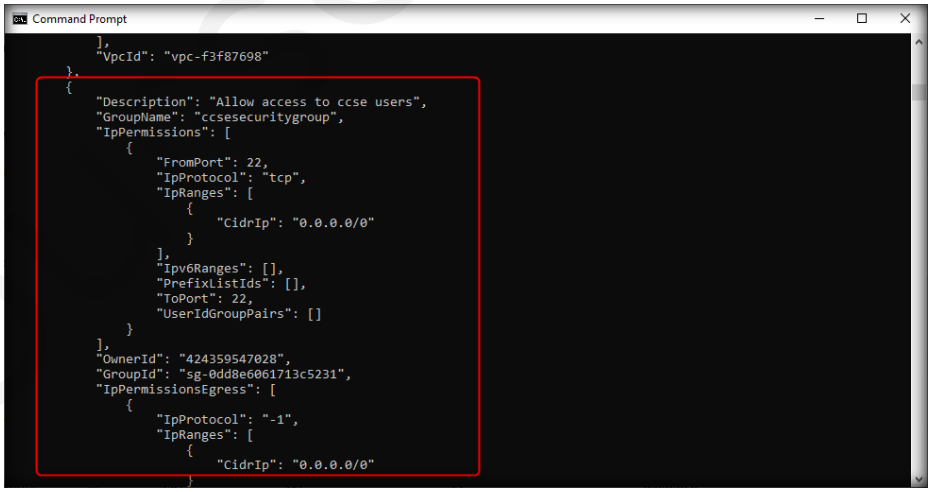


FIGURE 6.2.44: Information Regarding Security Group

50. Thus, a cloud security engineer can test for AWS resource data that gets leaked because of stolen or leaked access keys and take appropriate security measures to secure their environment.

Caution: Ensure you delete, shut down, or terminate all resources created and used in this lab to prevent their billing.

51. Now, to delete the instances created in this lab, navigate to the **Security Groups** under EC2 and select the checkbox for the security group you want to delete (here, **ccsesecuritygroup**). Select the **Actions** dropdown and click on the **Delete security groups** option. In the pop-up window, click on the **Delete** button.

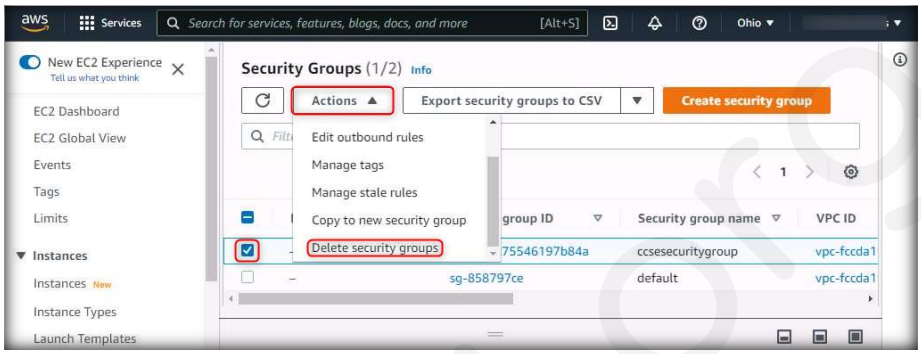


FIGURE 6.2.45: Deleting Security Group

52. Navigate to **User groups** under IAM console and select the checkbox of the instance created in this lab (here, **CloudSecurityEngineer**). Click on the **Delete** button. In the pop-up window, enter the user group's name and click on **Delete**.

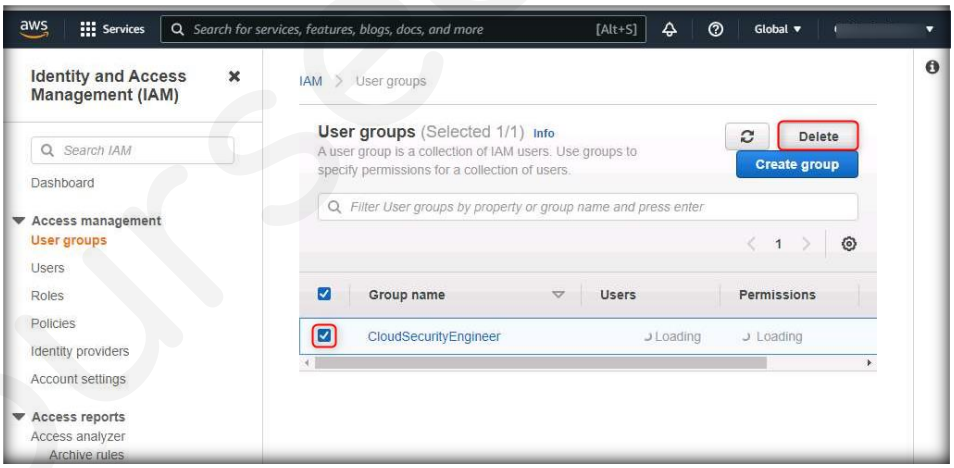


FIGURE 6.2.46: Deleting User Group

53. Navigate to **Users** under IAM console and select the checkbox of the instance created in this lab (here, **ccseuser**). Click on the **Delete** button. In the pop-up window, enter the user's name and click on the **Delete** button.

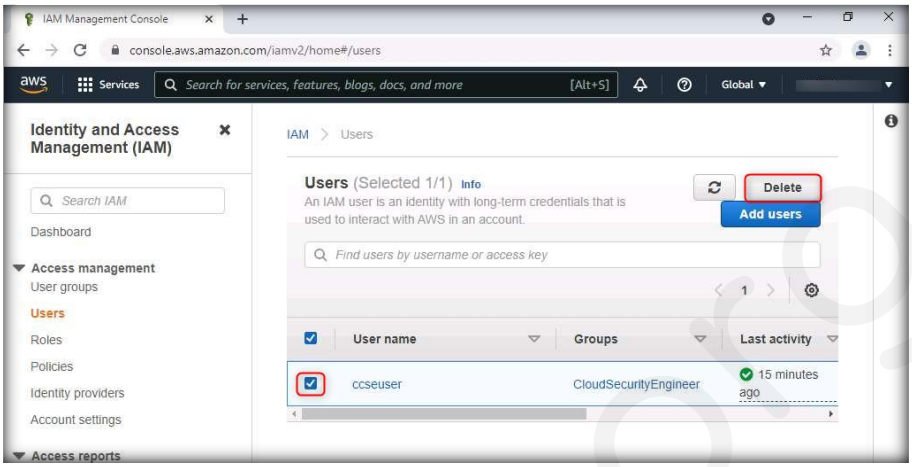


FIGURE 6.2.47: Deleting User

Lab Analysis

Analyze and document the results of this lab exercise. Provide your opinion on your target's security posture and exposure through free public information.

PLEASE TALK TO YOUR INSTRUCTOR IF YOU HAVE QUESTIONS ABOUT THIS LAB.

This page is intentionally left blank.