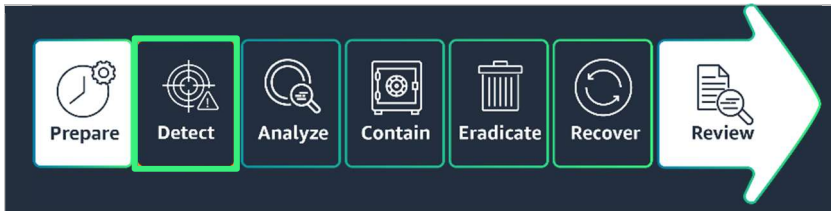
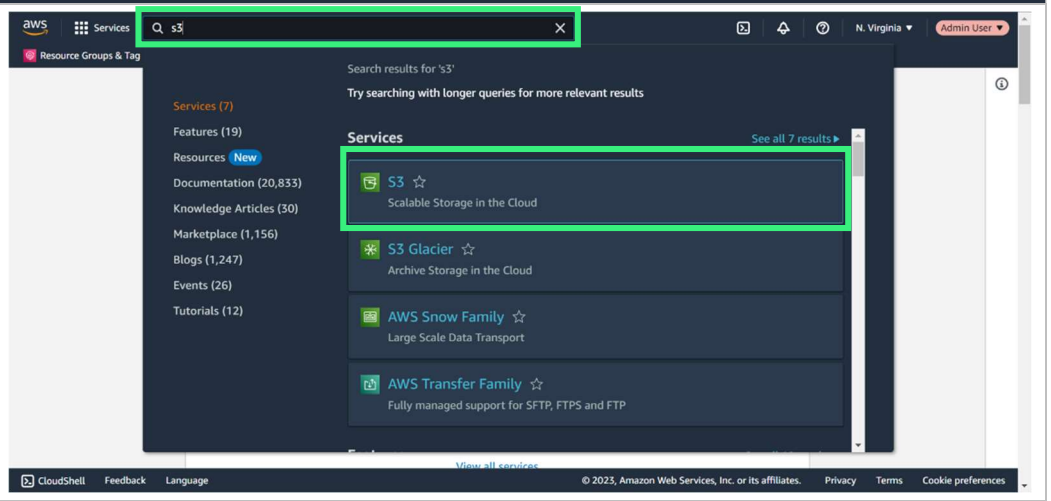
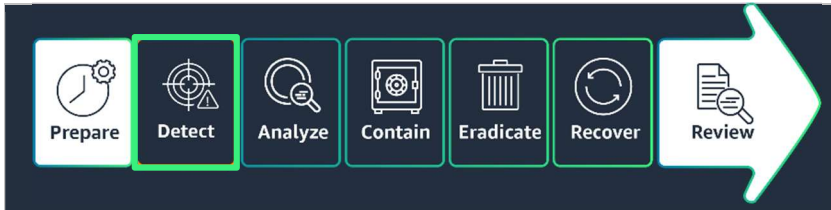
AWS Security Incident Response - Ransomware Job Aid



Detect

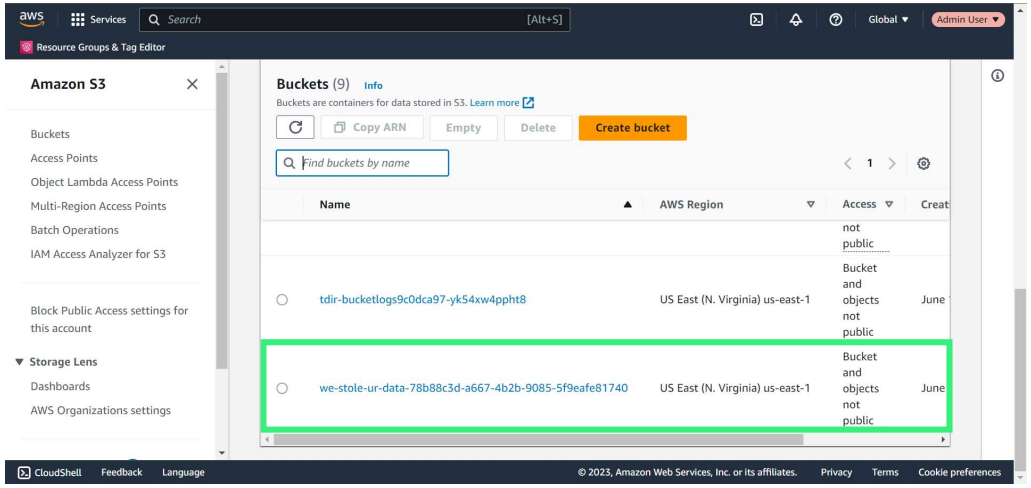
Use the S3 Console to identify the suspicious bucket.

Steps	Content	Visual
Step 1:	Let's navigate to our S3 buckets and see if we can find the suspicious bucket. From the AWS Management Console: 1. In the field search field enter **S3.** 2. Under Services, choose **S3**.	



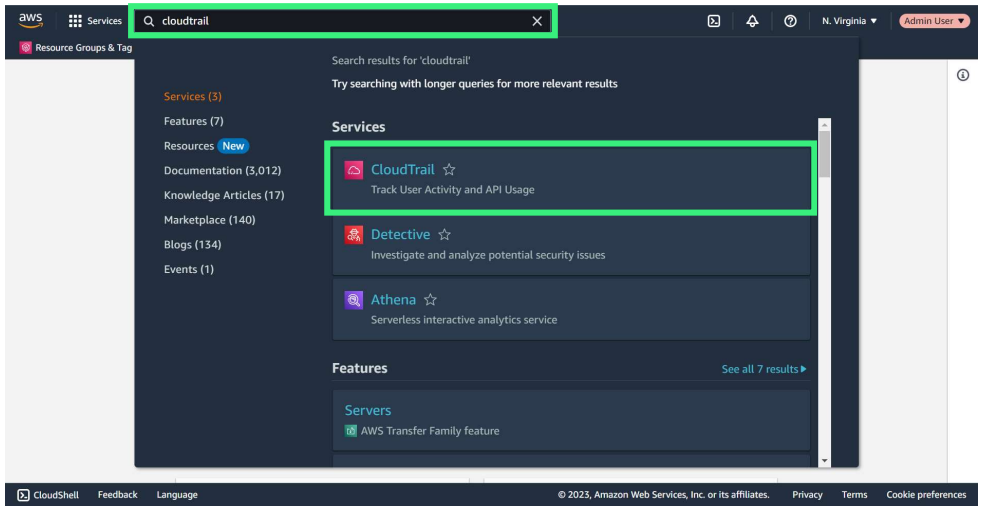
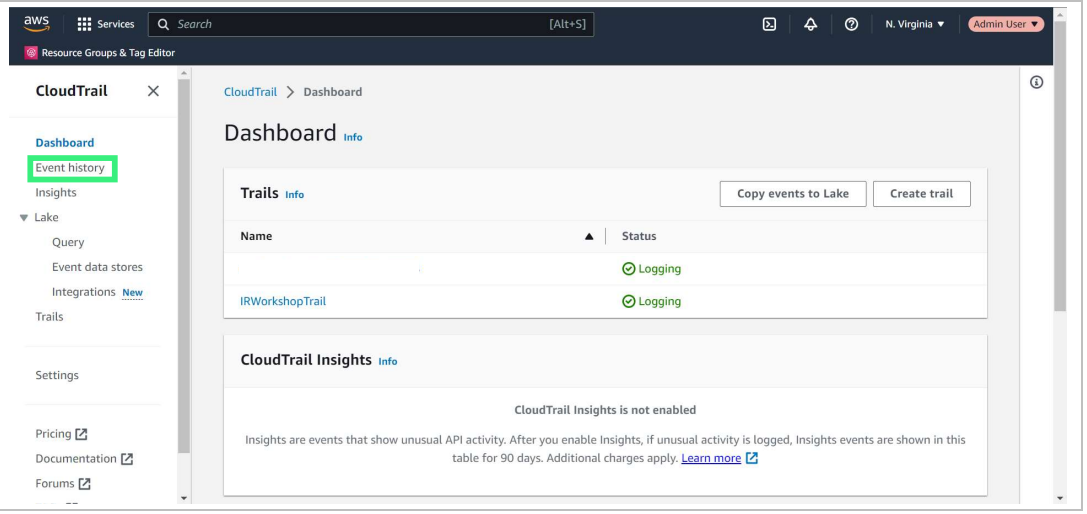
Detect

Use the S3 Console to identify the suspicious bucket.

Steps	Content	Visual
Step 2:	The Amazon S3 page displays. Scroll down to the Buckets section, and locate the bucket, we-stole-ur-data-*. This confirms the bucket exists and we should investigate further. Next, we will use AWS CloudTrail to determine who created the suspicious bucket.	

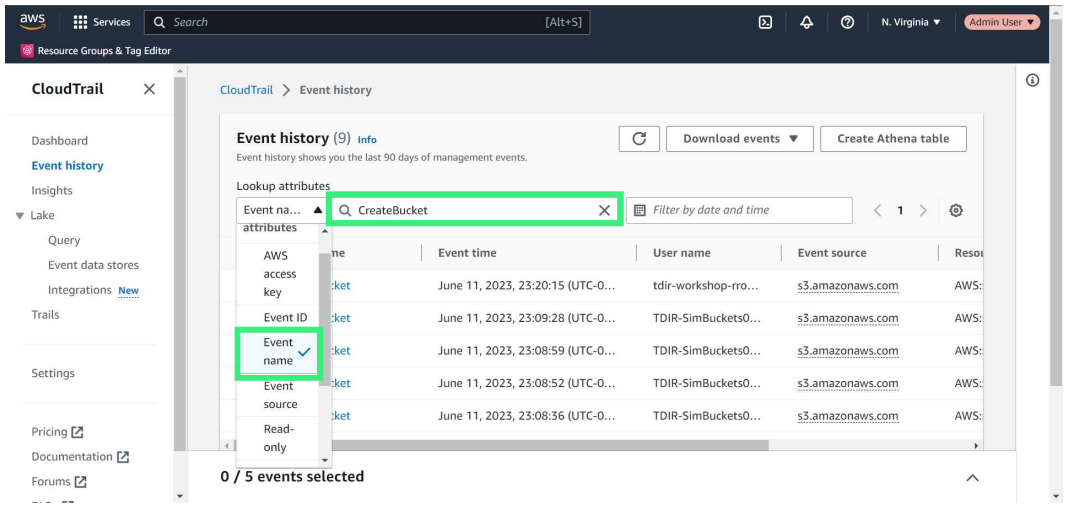
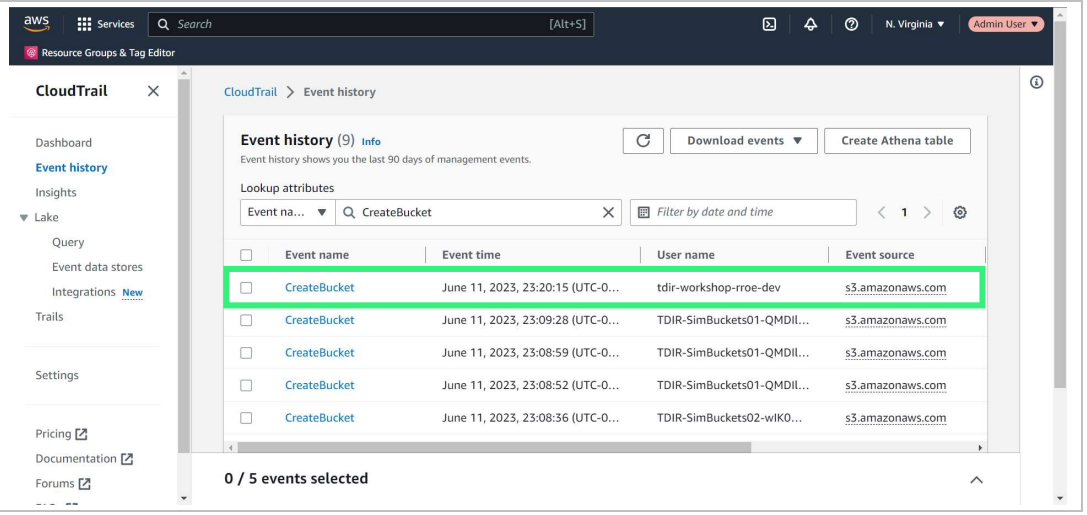
Analyze

Review the event history in CloudTrail to determine who created the suspicious bucket.

Steps	Content	Visual
Step 1:	Let's navigate to CloudTrail to determine who created the suspicious bucket. From the AWS Management Console: 1. In the search field, enter **CloudTrail**. 2. Under Services, choose **CloudTrail**.	
Step 2:	The CloudTrail dashboard displays. To view the Event history in CloudTrail, choose **Event history**.	

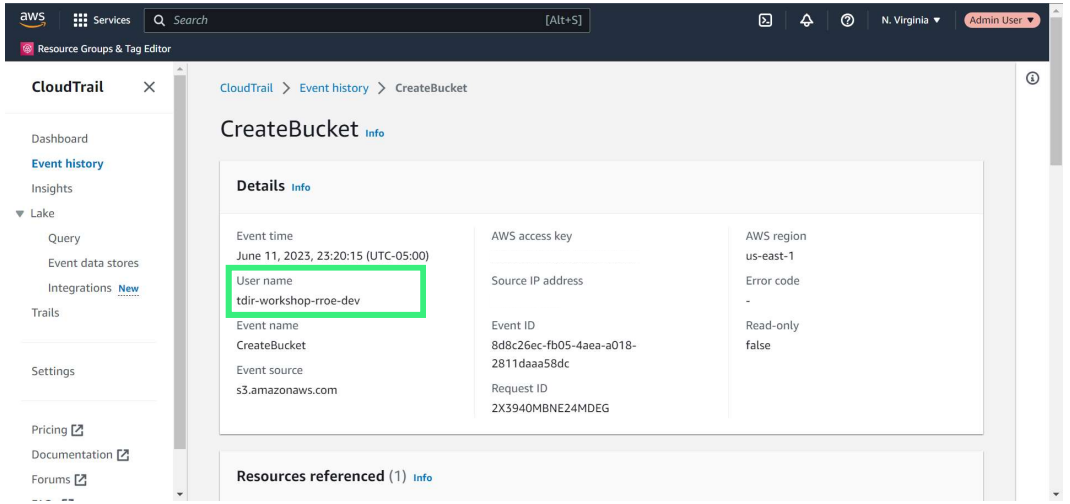
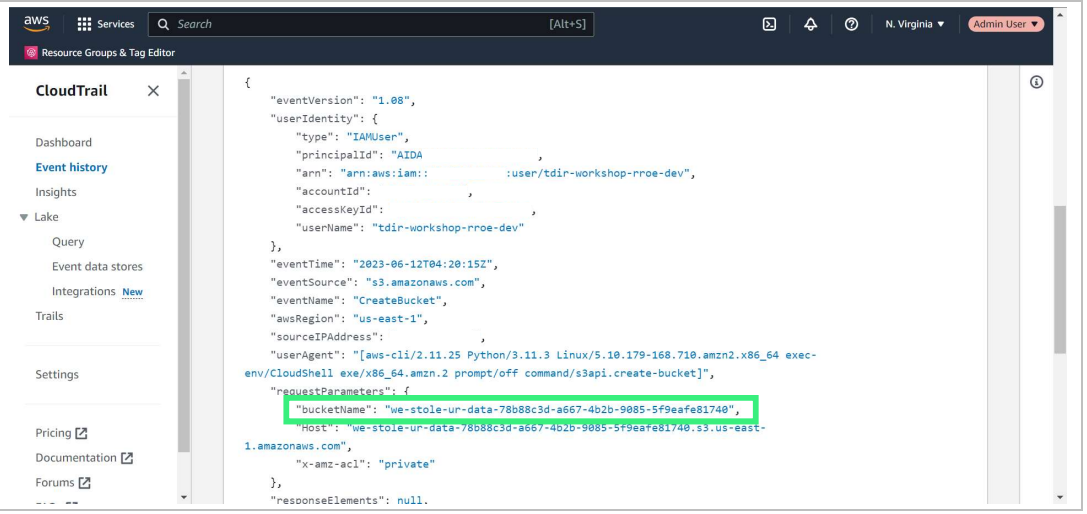
Analyze

Review the event history in CloudTrail to determine who created the suspicious bucket.

Steps	Content	Visual
Step 3:	The Event history page displays. Under Lookup attributes, go to the first drop-down menu and choose **Event name**. Then, in the second field, enter **CreateBucket**.	
Step 4:	Notice a CreateBucket event appears. Let's determine if this event was where the we-stole-ur-data bucket was created. Choose **CreateBucket**.	

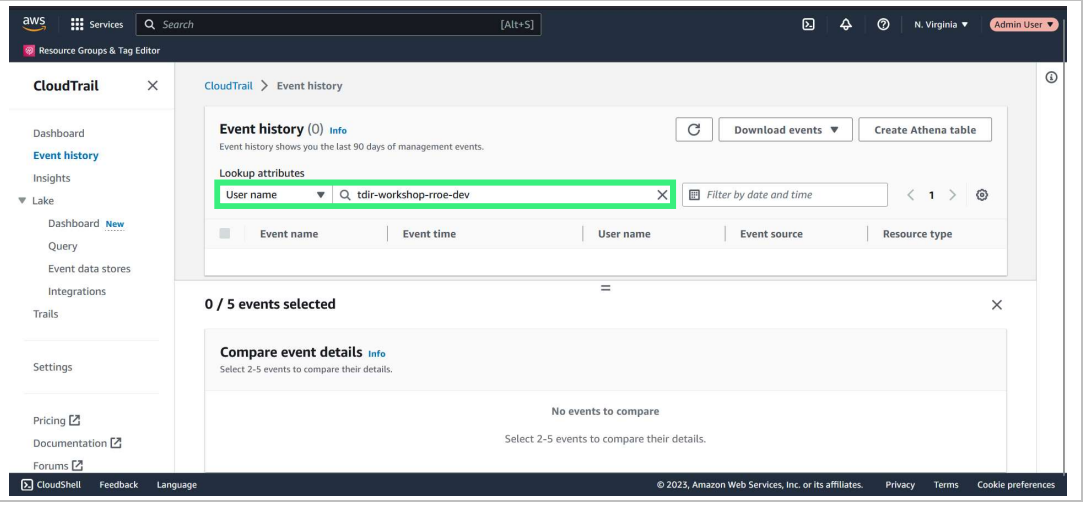
Analyze

Review the event history in CloudTrail to determine who created the suspicious bucket.

Steps	Content	Visual
Step 5:	The CreateBucket Event history page displays. In the Details section, notice the username tdir-workshop-rroe-dev.	
Step 6:	Scroll down to the Event records section, and notice the CreateBucket parameter matches the suspicious bucket. This confirms that the tdir-workshop-rroe-dev was the user who created the new bucket. We identified the suspicious bucket and determined who created it. Next, we will find out what events the user performed and who created the bucket.	

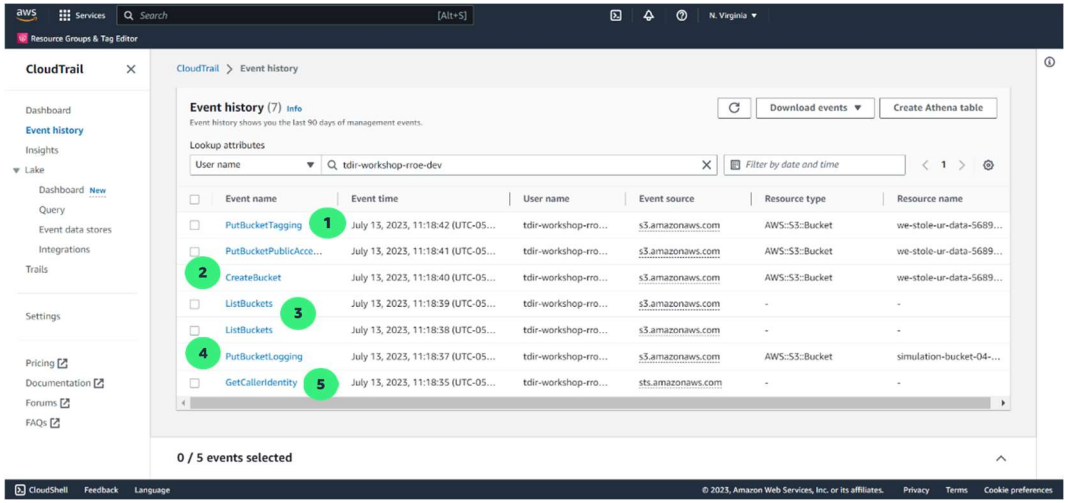
Analyze

Use CloudTrail to investigate the events performed by the user who created the suspicious bucket.

Steps	Content	Visual
Step 1:	Now that we've identified the user as tdir-workshop-rroe-dev, let's use CloudTrail to investigate any other events performed by this user. From the Event history page in CloudTrail, in the first Lookup attributes field, choose **User name.** In the second field, enter the **tdir-workshop-rroe-dev** user that we identified earlier and press enter.	

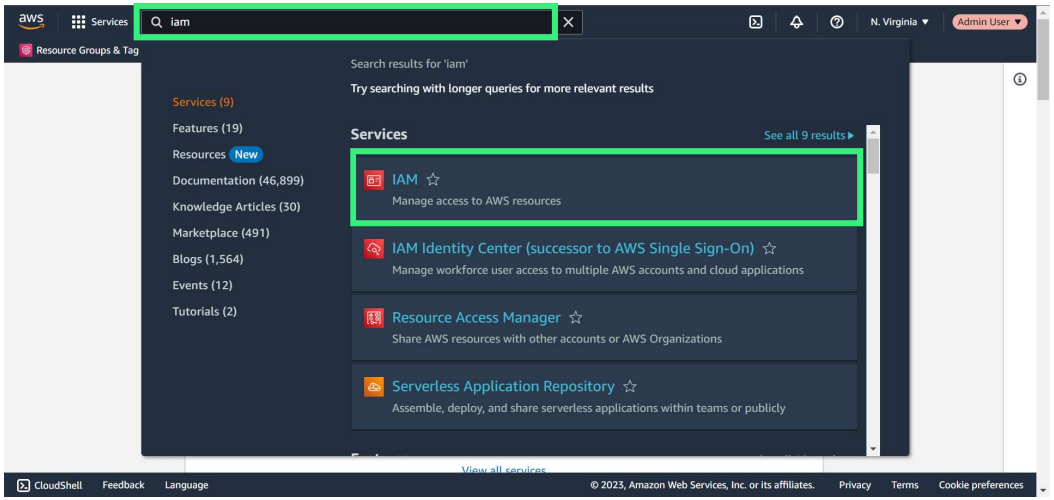
Analyze

Use CloudTrail to investigate the events performed by the user who created the suspicious bucket.

Steps	Content	Visual
Step 2:	The results of the query show activity related to the suspicious bucket. There is no additional activity to address. In the eventname column we see the following actions were taken: 1. **PutBucketTagging:** This creates a tag set for an S3 bucket. 2. **CreateBucket:** This API creates buckets in the AWS account and was the API used to create the suspicious bucket named we-stole-ur-data-* 3. **ListBuckets:** This API lists buckets in the AWS account. 4. **PutBucketLogging:** This API changes the logging status of an S3 bucket. 5. **GetCallerIdentity:** This API provides information to the caller about the credentials that are currently in use. It is similar to the whoami command found on most Unix-like operating systems. Next, let's contain the situation by preventing the user (tdir-workshop-rroe-dev) from signing in to the console and deactivating their access key.	

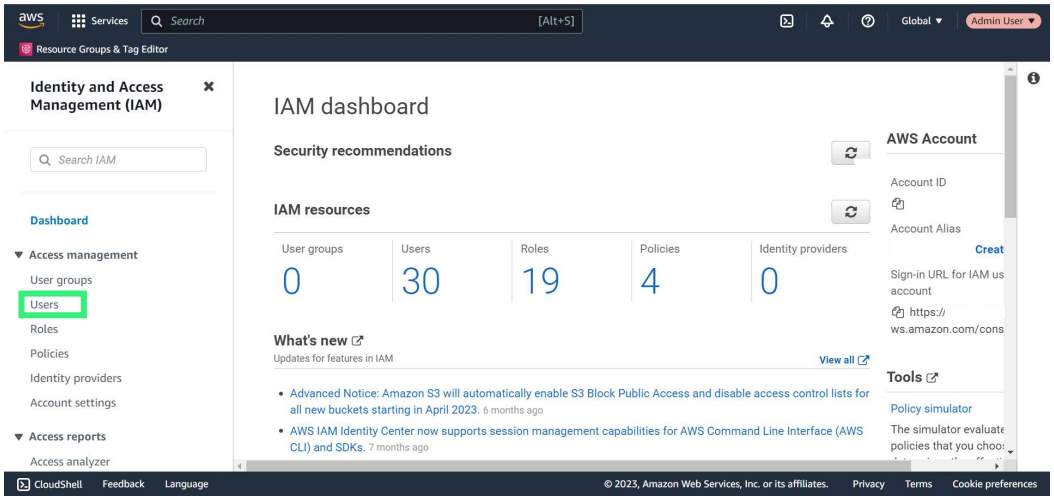
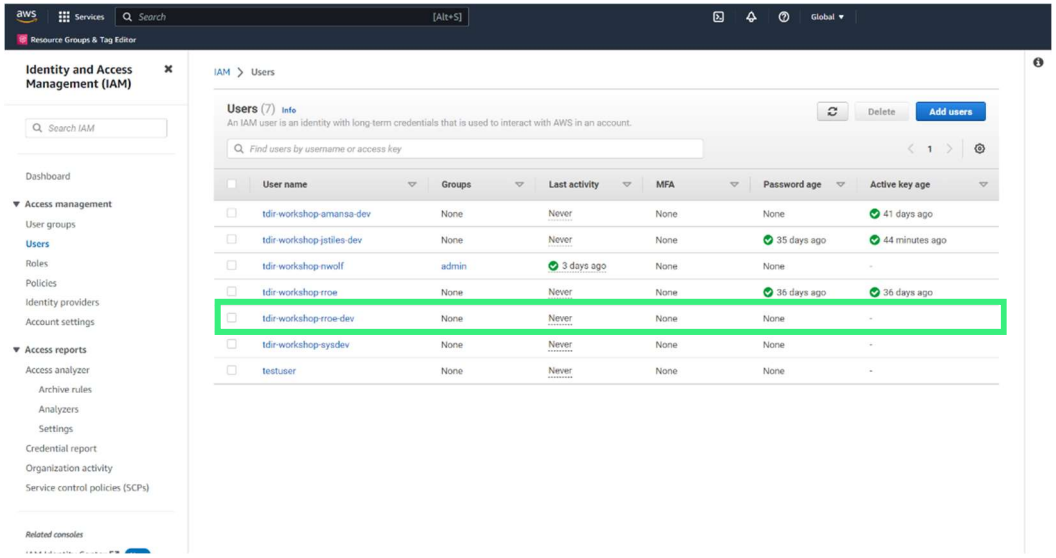
Contain

Disable the compromised user's console access and deactivate the user's access key.

Steps	Content	Visual
Step 1:	To disable the credentials for the compromised user, navigate to the IAM console: 1. In the search field, enter **IAM.** 2. Under Services, choose **IAM**.	

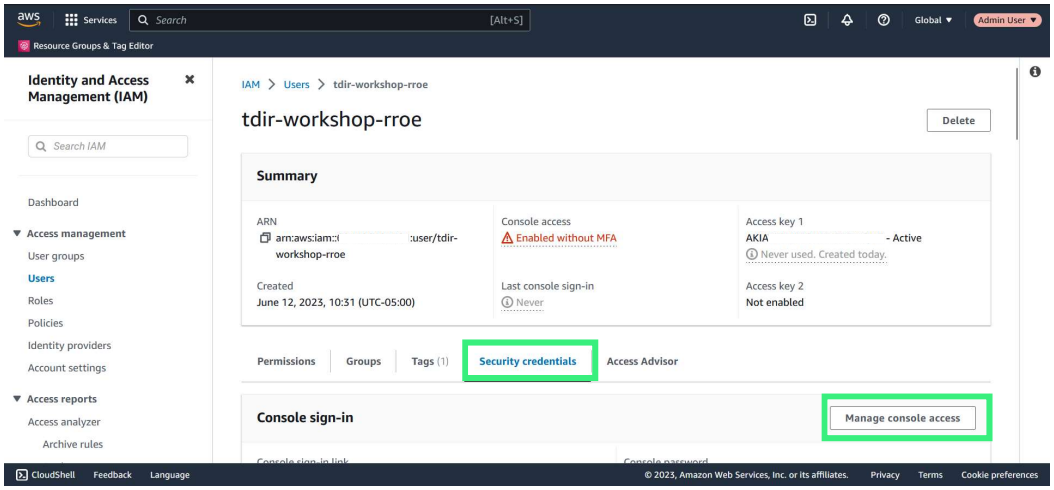
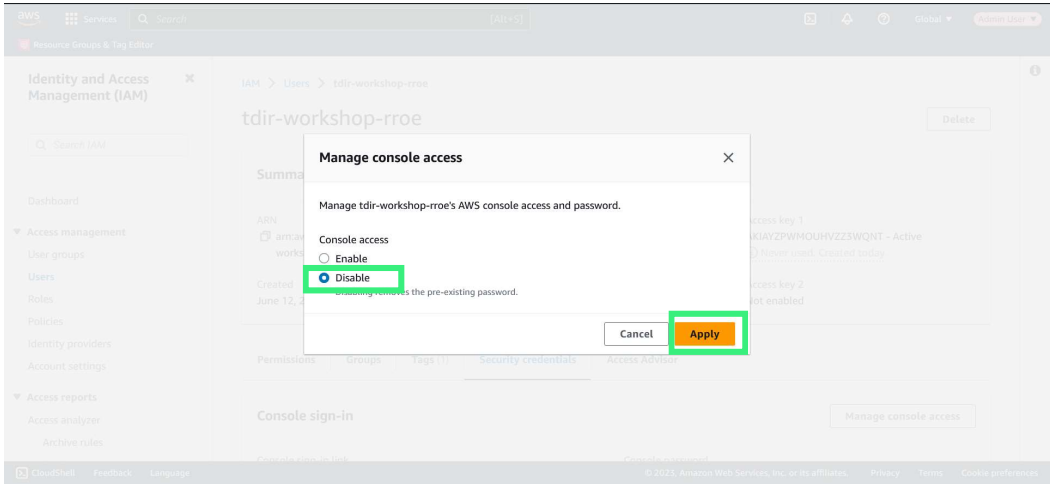
Contain

Disable the compromised user's console access and deactivate the user's access key.

Steps	Content	Visual
Step 2:	The IAM dashboard displays. To disable the user tdir-workshop-rroe-dev under Access management, choose **Users**.	
Step 3:	A list of users displays on the screen. Choose **tdir-workshop-rroe-dev.**	

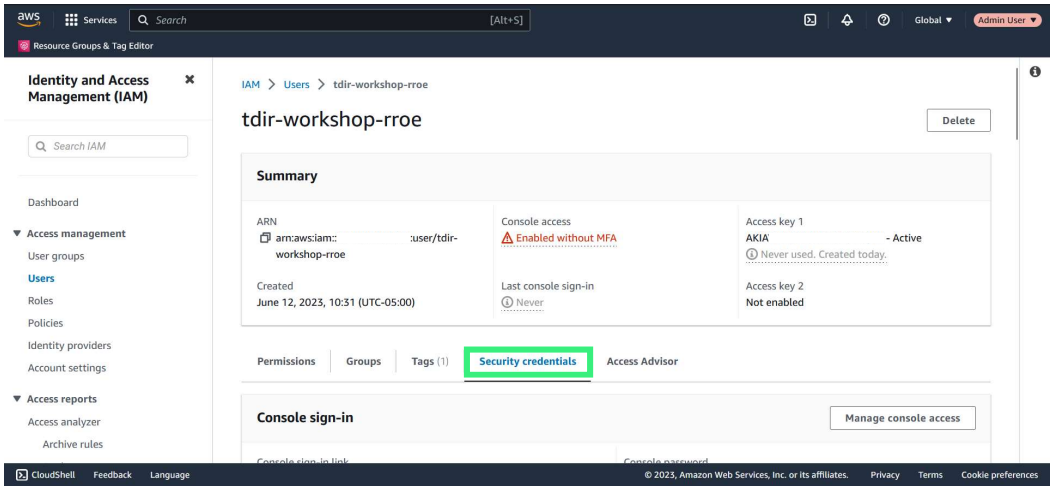
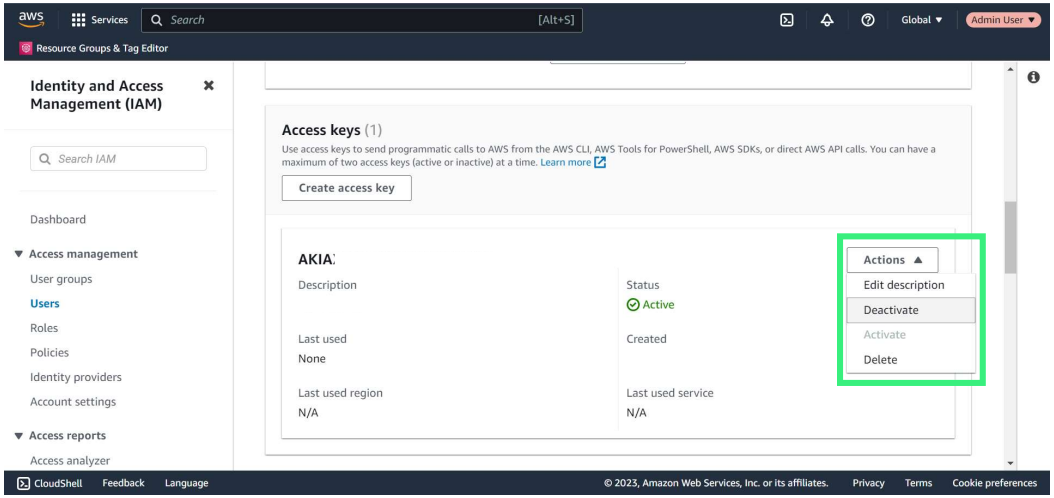
Contain

Disable the compromised user's console access and deactivate the user's access key.

Steps	Content	Visual
Step 4:	The properties for the tdir-workshop-rroe-dev user display. Choose the **Security credentials** tab and then choose **Manage console access**.	
Step 5:	The Manage console access window displays. Under Console access, select **Disable,** and then choose **Apply** to restrict the user access. We just prevented the compromised user from signing into the console. Now, we can deactivate the Access key.	

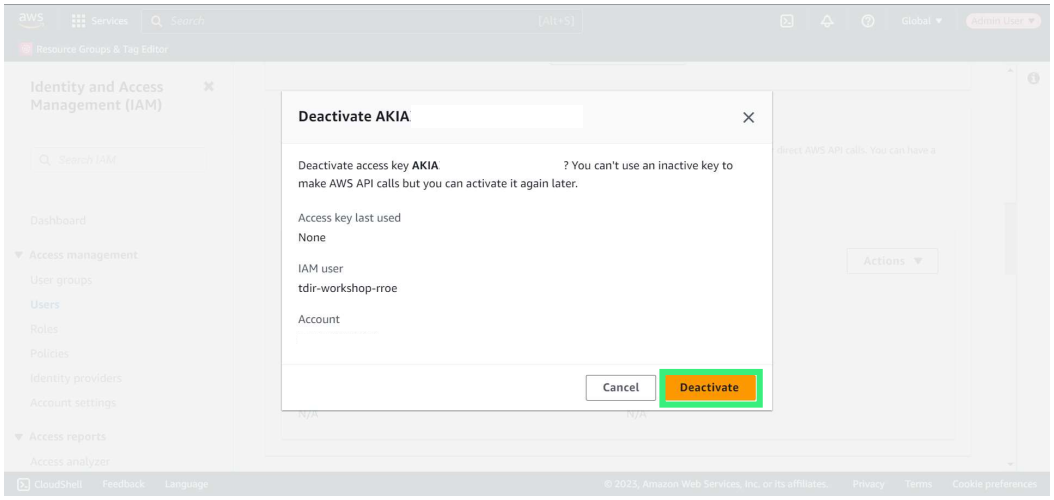
Contain

Disable the compromised user's console access and deactivate the user's access key.

Steps	Content	Visual
Step 6:	To disable the Access key, choose the **Security credentials** tab.	
Step 7:	Scroll down to the Access keys section. Choose the **Actions** drop-down menu and choose **Deactivate**.	

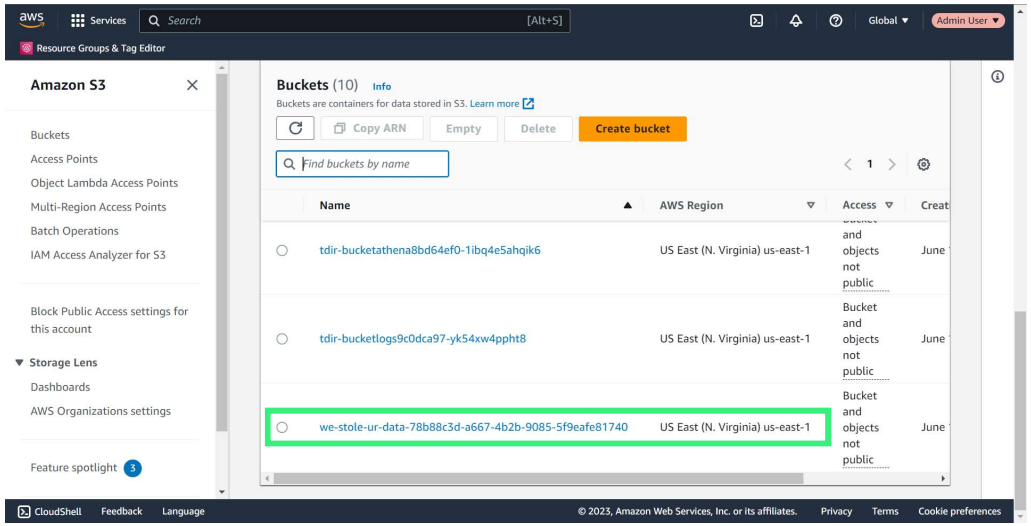
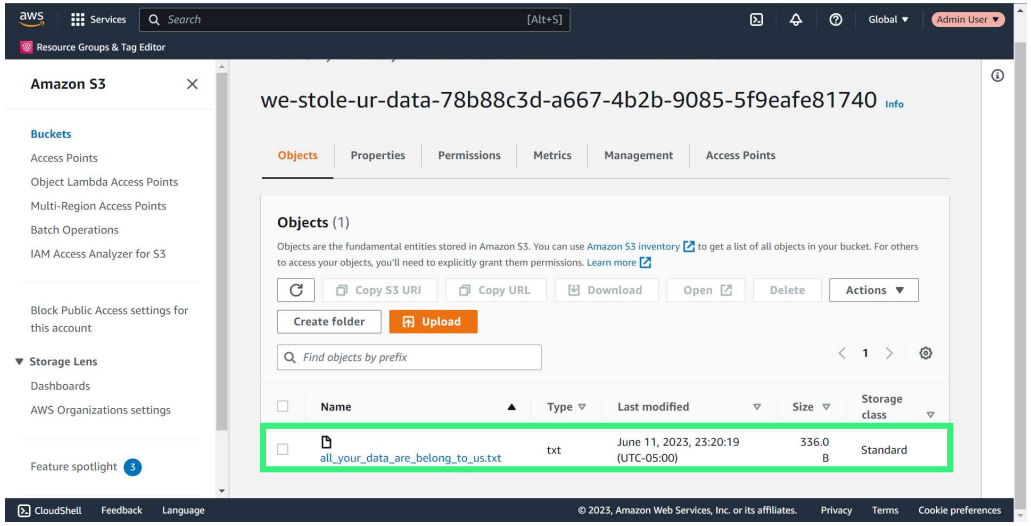
Contain

Disable the compromised user's console access and deactivate the user's access key.

Steps	Content	Visual
Step 8:	A confirmation window displays. Choose **Deactivate**. Now that we've contained the situation by disabling the user, let's check the suspicious bucket and see what we find.	

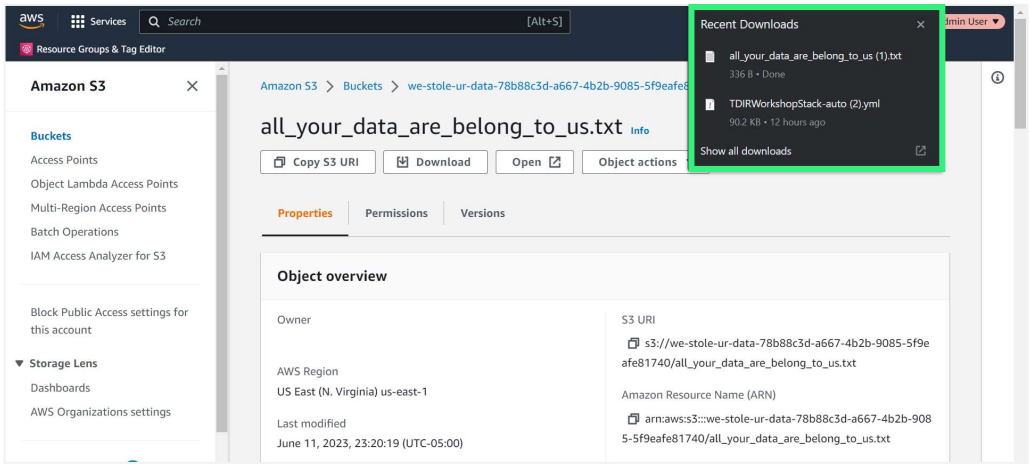
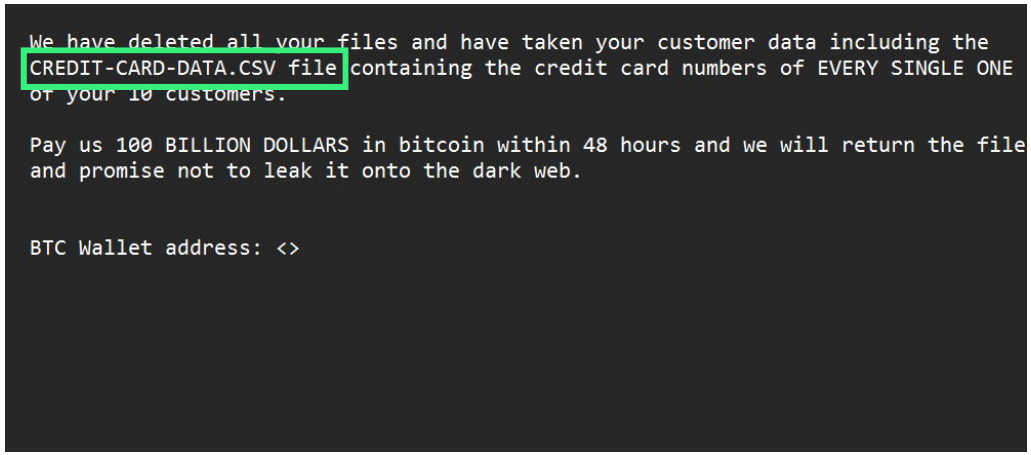
Analyze

Use the S3 console to review objects in the suspicious bucket.

Steps	Content	Visual
Step 1:	From the Amazon S3 Buckets page, navigate to the suspicious bucket titled we-stole-ur-data-*. To see the objects in the bucket, choose **we-stole-ur-data-*.**	
Step 2:	The objects in the bucket displays on the Objects tab. Choose the file name **all_your_data_are_belong_to_us.txt**.	

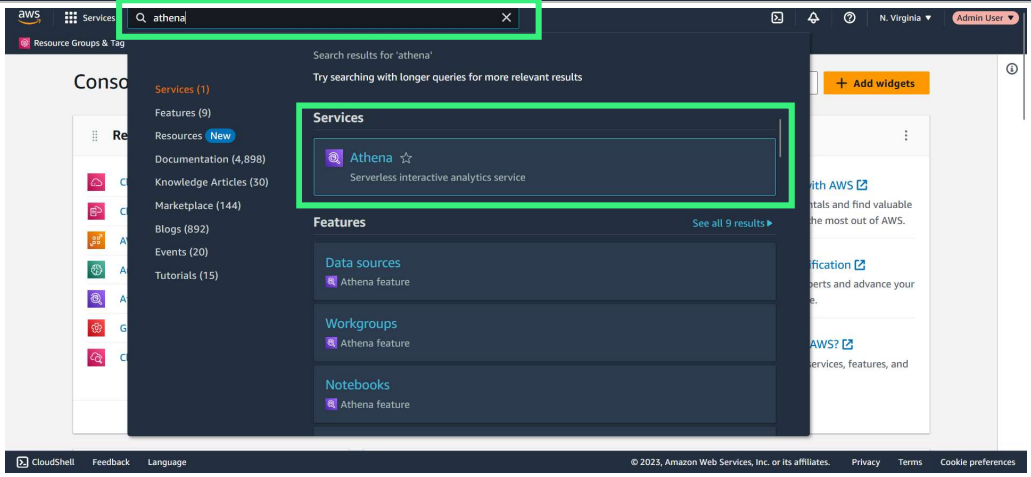
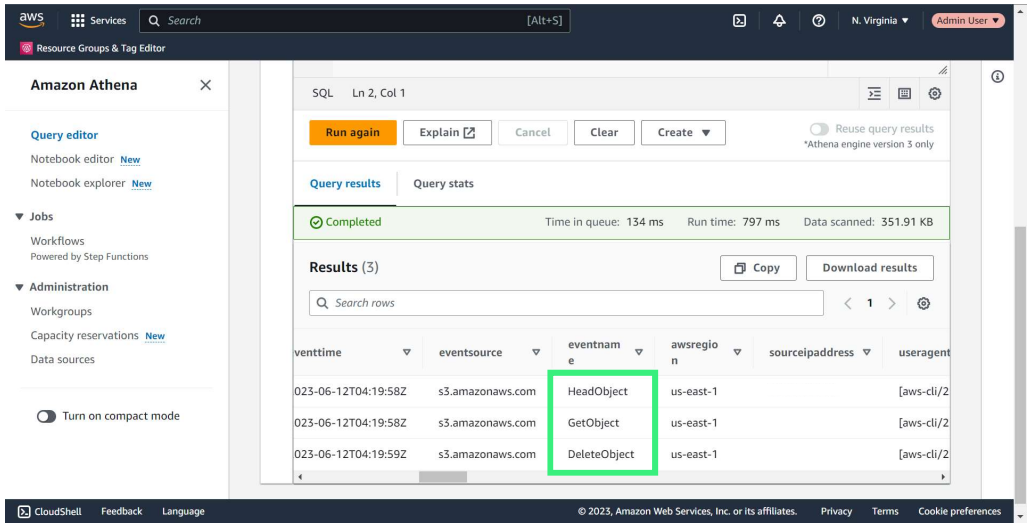
Analyze

Use the S3 console to review objects in the suspicious bucket.

Steps	Content	Visual
Step 3:	To examine the contents of the file, from the download menu, choose **all_your_data_are_belong_to_us.txt**. **Note:** We will download this file for training purposes; in a real-world scenario, we would want to do this in a secure environment or isolated Amazon Elastic Compute (Cloud Amazon EC2) instance.	
Step 4:	Once the file downloads, open the **all_your_data_are_belong_to_us.txt** file in Notepad. The file reveals that the ransomware group has taken customer data in the form of a sensitive file named credit-card-data.csv. Write down the name of this file for future reference. **Note:** While we're investigating, be aware that any information a threat actor provides may be incorrect or be an attempt to evade defenses. Next, we will use Amazon Athena to locate the credit-card-data.csv file.	

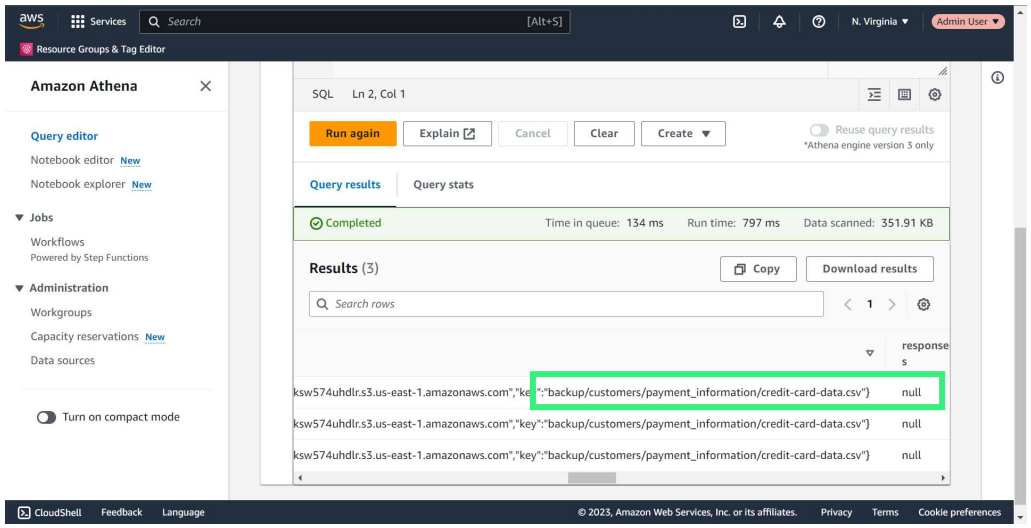
Analyze

Use Athena to determine the location of the sensitive file that the ransomware group claimed they took.

Steps	Content	Visual
Step 1:	Use Athena to query AWS CloudTrail, to determine the location of the sensitive file. From the AWS Management Console: 1. In the search field, enter **Athena**. 2. Under Services, choose **Athena**.	
Step 2:	Based on the results of the query, we see that the API calls in the eventname column are GetObject, HeadObject, and DeleteObject. This tells us that the object metadata and the object itself were obtained (HeadObject and GetObject) and that the object was deleted (DeleteObject).	

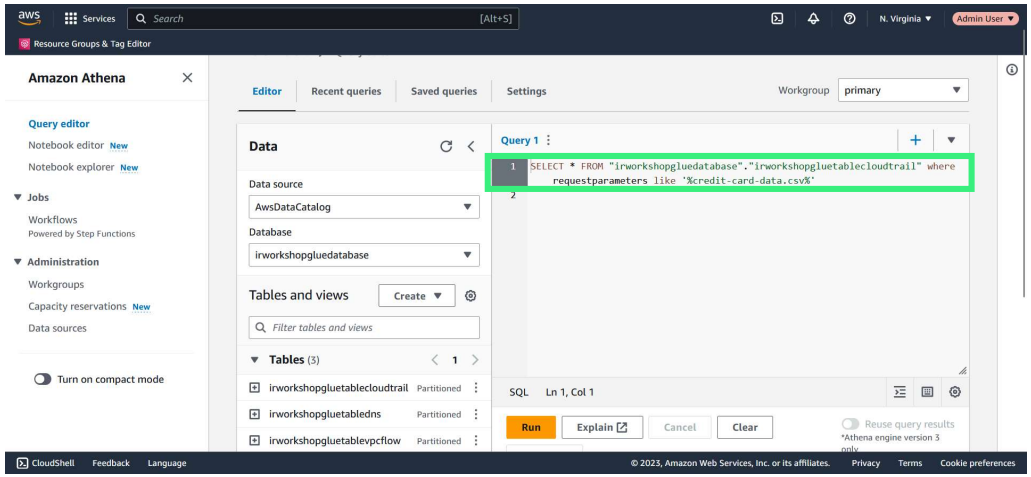
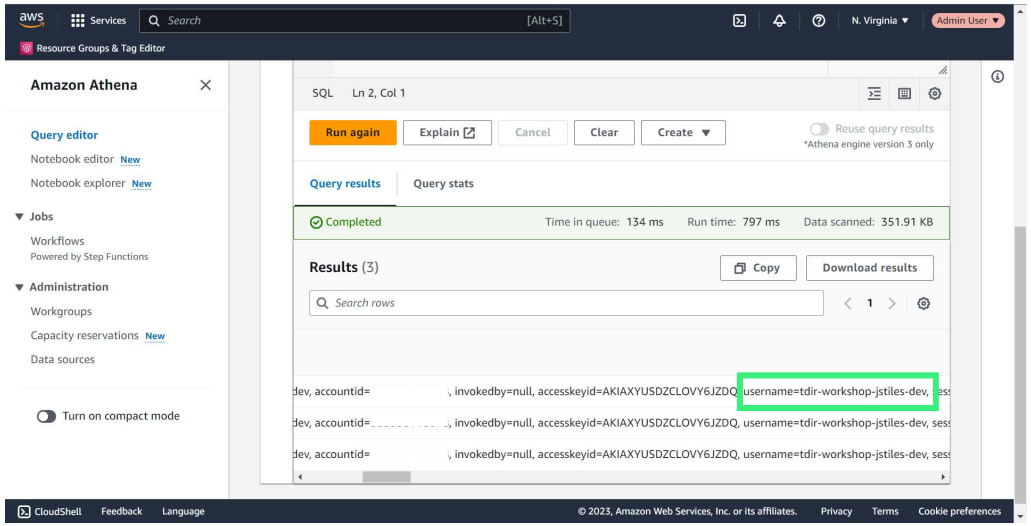
Analyze

Use Athena to determine the location of the sensitive file that the ransomware group claimed they took.

Steps	Content	Visual
Step 3:	Scroll horizontally to the requestparameters column to see exactly which object the API calls were made for. This information confirms that the **credit-card-data.csv** object was taken and deleted, and that the S3 bucket from which it was taken was simulation-bucket-03-8lonc4g9dwxzrbim with a prefix/folder location of backup/customers/payment_information/. Because the CSV file was deleted, we will need to determine the name of the IAM user that retrieved and deleted the CSV object.	

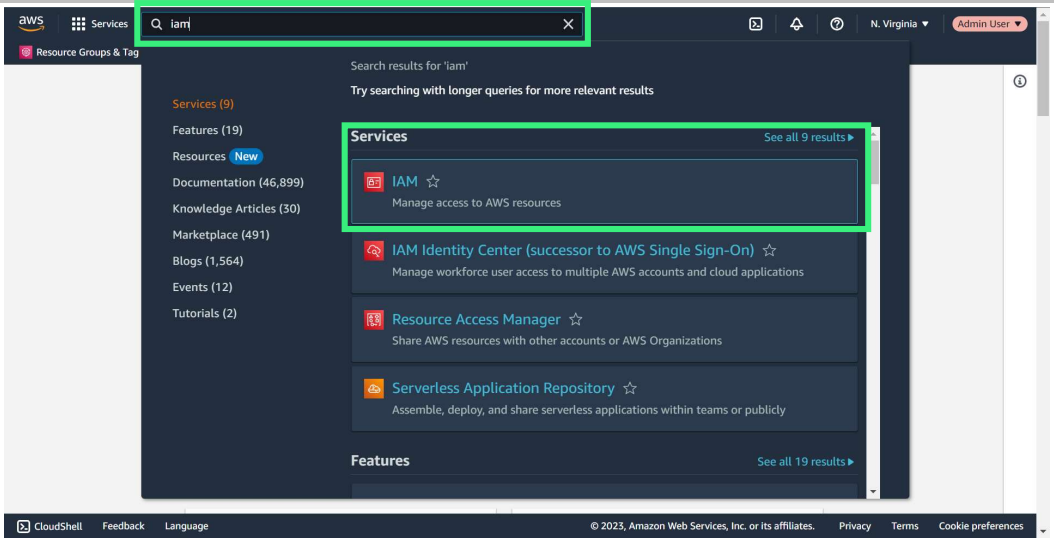
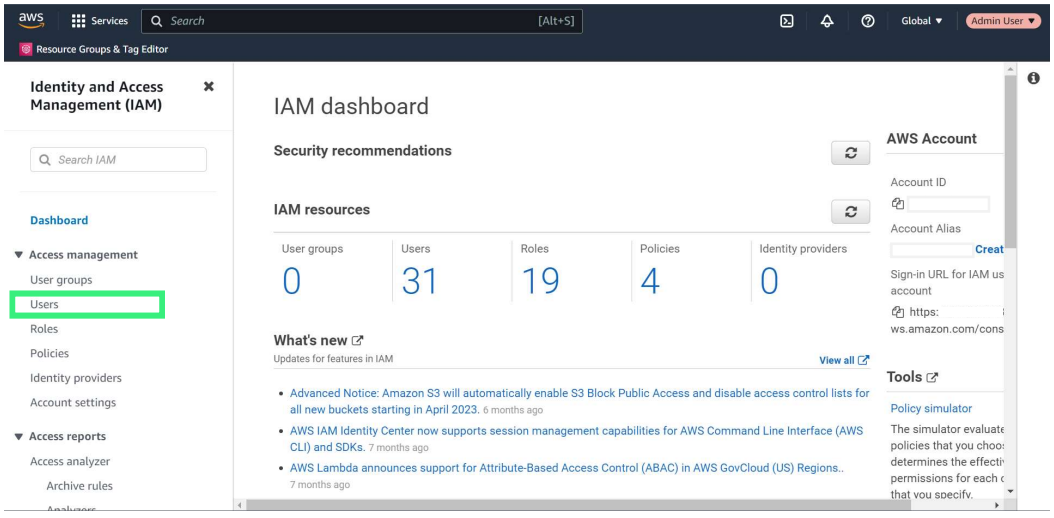
Analyze

Use Athena to determine the IAM user that deleted the sensitive file.

Steps	Content	Visual
Step 1:	Now that we've confirmed that the file was deleted, we can use this query to see what IAM user performed the action.	
Step 2:	Scroll to the useridentity column to see that the username parameter is tdir-workshop-jstiles-dev. This is the name of the IAM user that was used to retrieve and delete the credit-card-data.csv object. Because this tdir-workshop-jstiles-dev user was involved in the new activity, let's prevent the compromised user from signing in to the console by disabling their access and deactivating their access key.	

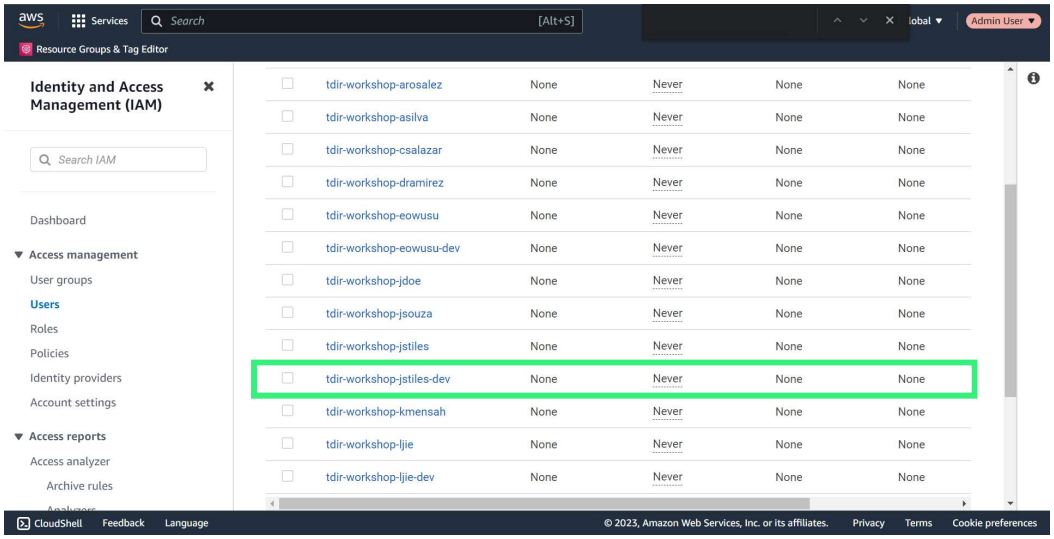
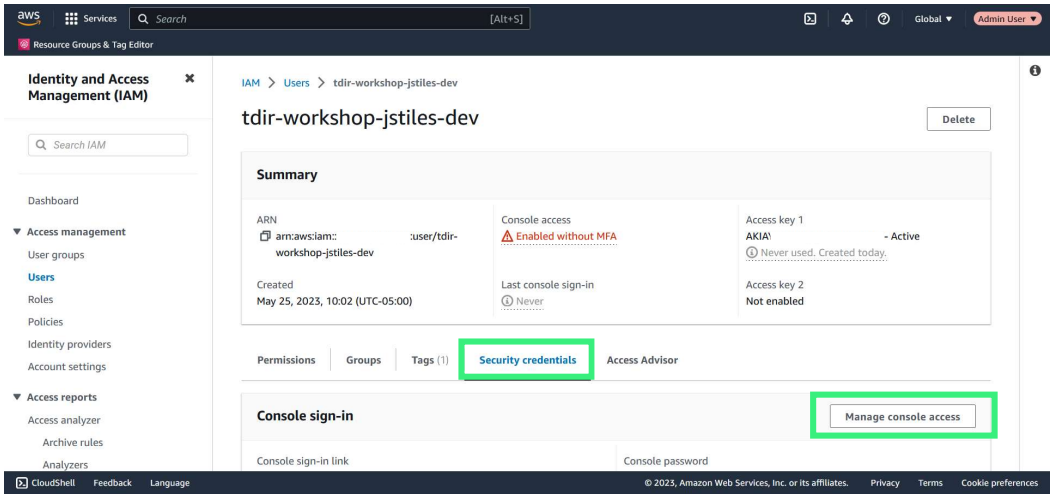
Contain

Disable the user's console access, and deactivate the user's access key.

Steps	Content	Visual
Step 1:	To navigate to the IAM console: 1. In the search field, enter **IAM.** 2. Under Services, choose **IAM**.	
Step 2:	The IAM dashboard displays. To disable the user tdir-workshop-jstiles-dev under Access management, choose **Users**.	

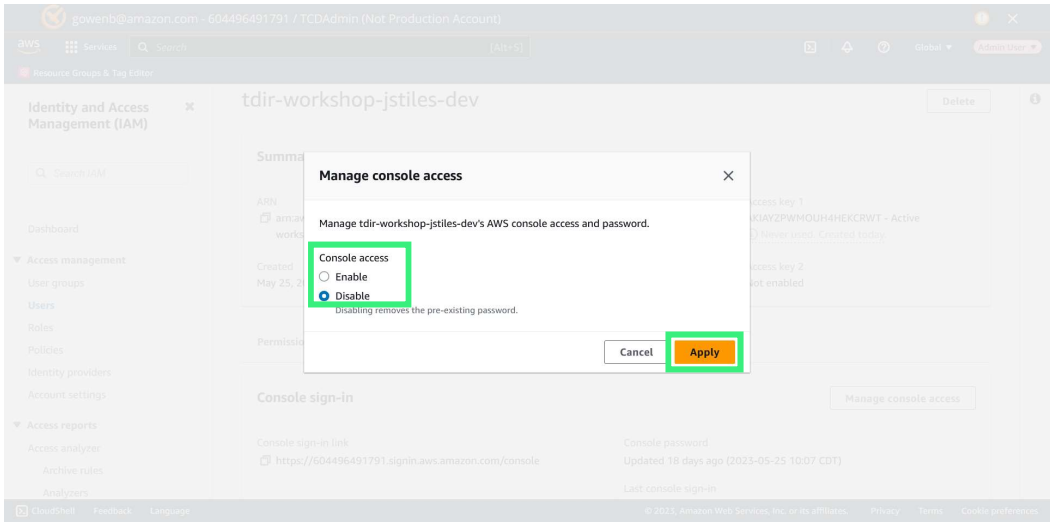
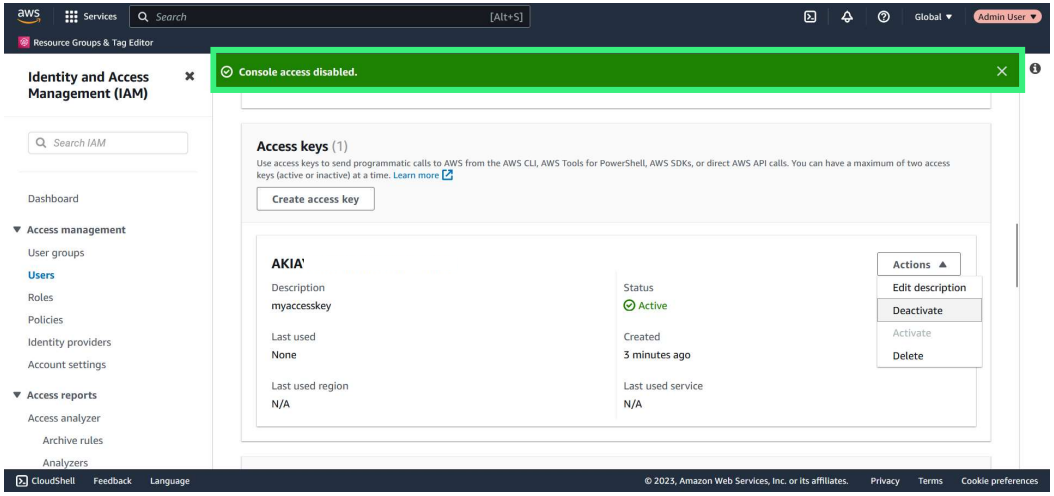
Contain

Disable the user's console access, and deactivate the user's access key.

Steps	Content	Visual
Step 3:	A list of users displays on the screen. Choose **tdir-workshop-jstiles-dev.**	
Step 4:	The properties for the tdir-workshop-jstiles-dev user display. Choose the **Security credentials** tab, and then choose **Manage console access**.	

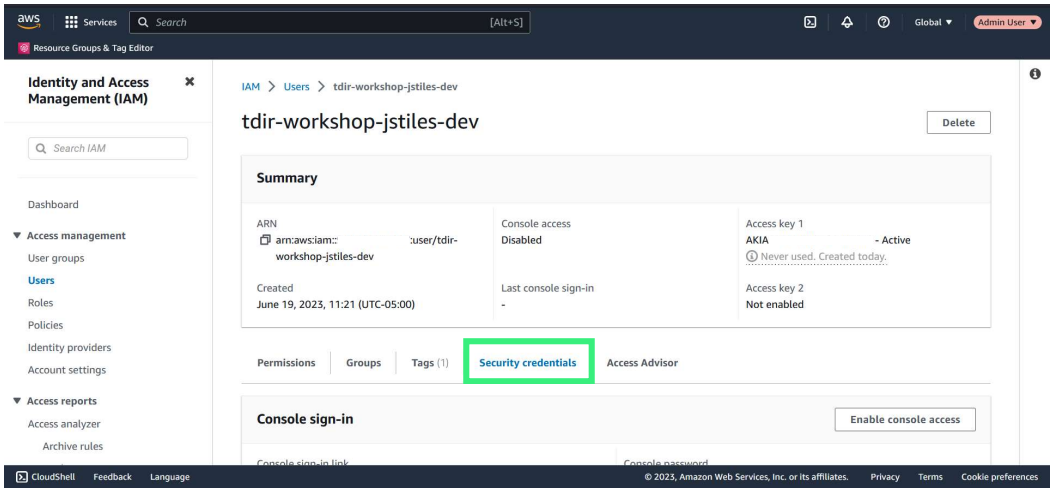
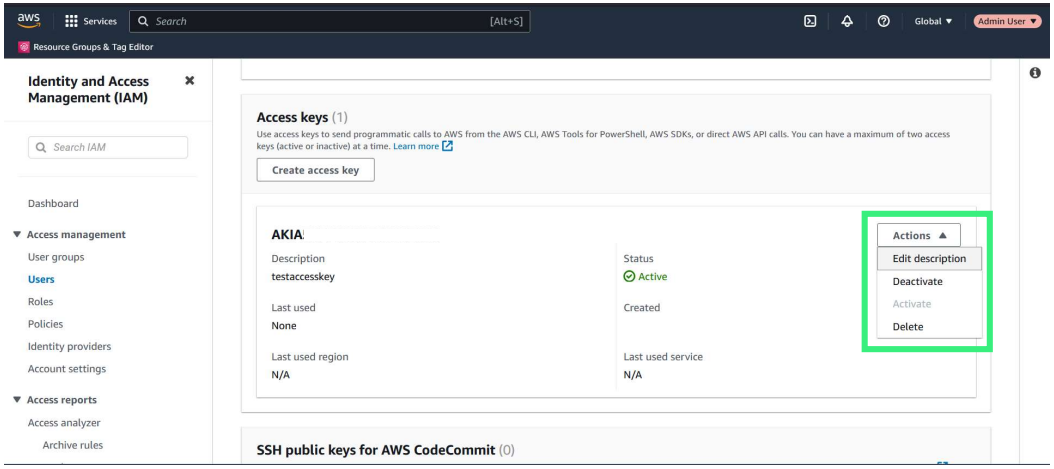
Contain

Disable the user's console access, and deactivate the user's access key.

Steps	Content	Visual
Step 5:	The Manage console access window displays. For console access, select **Disable**, and then choose **Apply** to restrict the user's access. We just prevented the compromised user from signing in to the console.	
	The message, "Console access disabled," displays.	

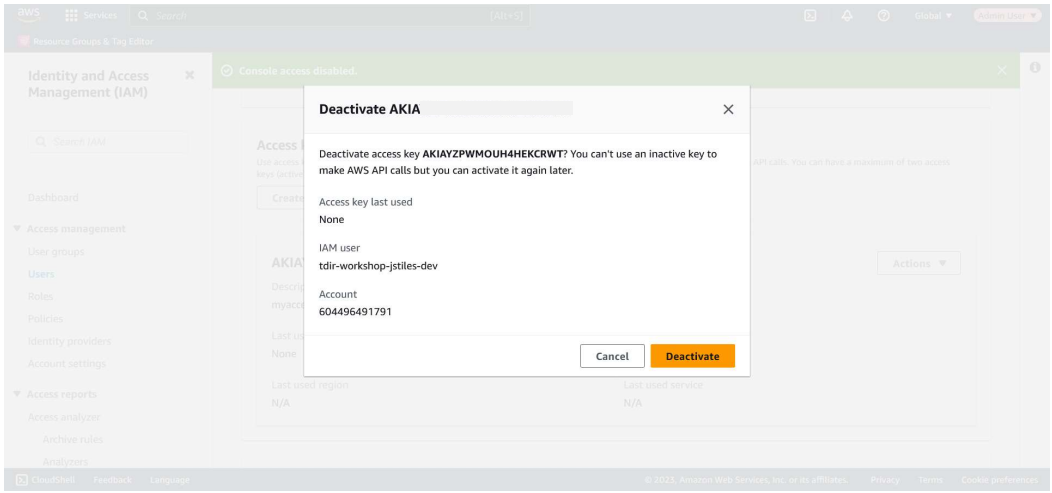
Contain

Disable the user's console access, and deactivate the user's access key.

Steps	Content	Visual
Step 6:	To deactivate the access key. Choose the **Security credentials** tab.	
Step 7:	Scroll down to the Access keys section. Choose the **Actions** drop-down menu, and then choose **Deactivate**.	

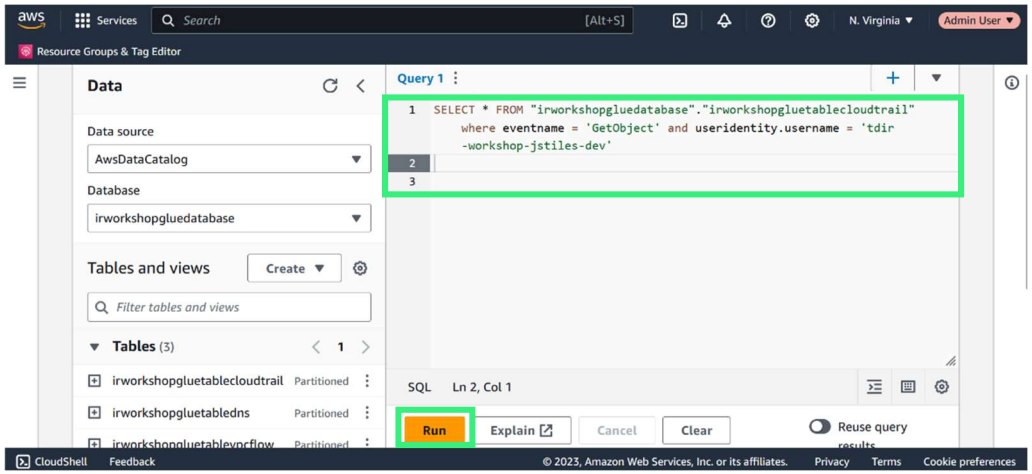
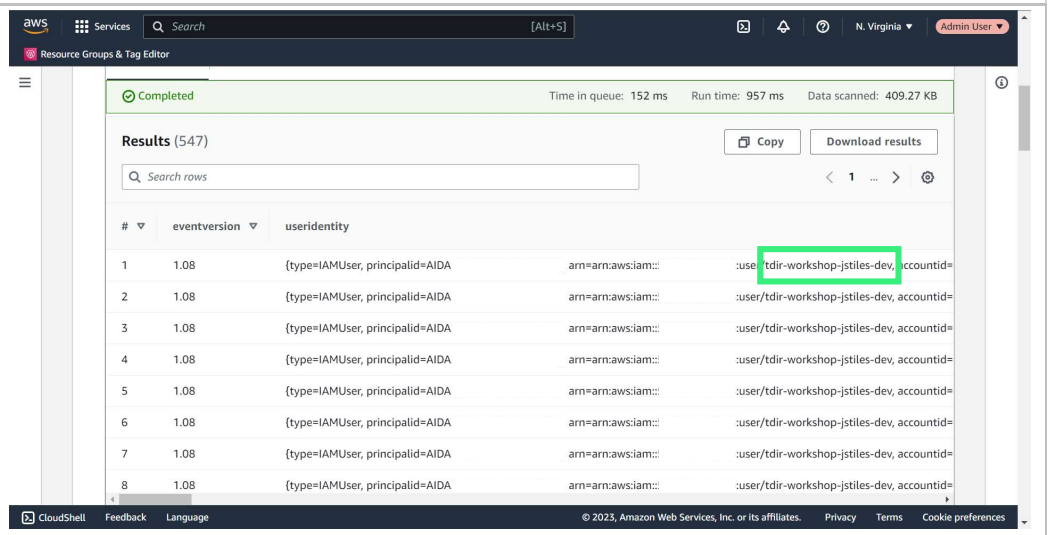
Contain

Disable the user's console access, and deactivate the user's access key.

Steps	Content	Visual
Step 8:	A confirmation pop-up displays. Choose **Deactivate**. Now that we've contained the situation by restricting permissions of the user, let's check the new S3 bucket and see what we find.	

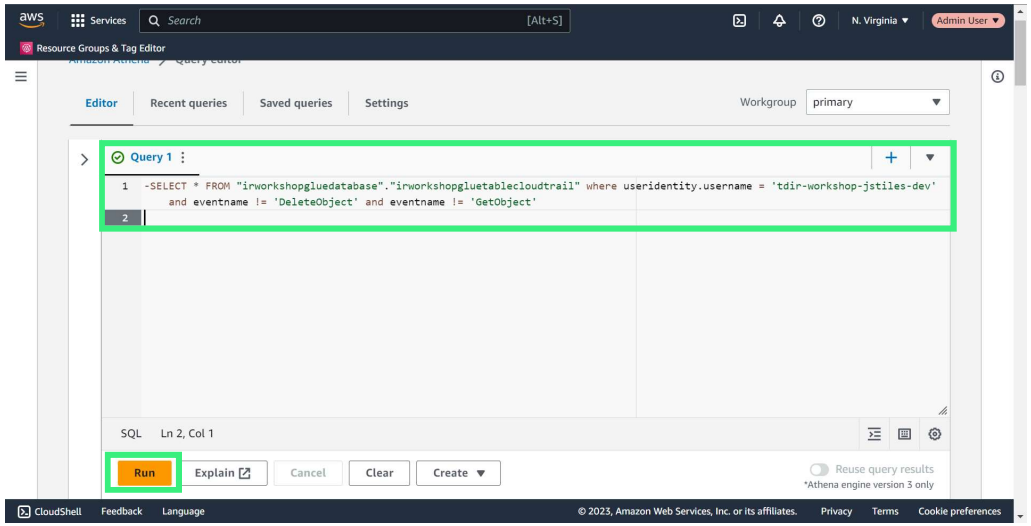
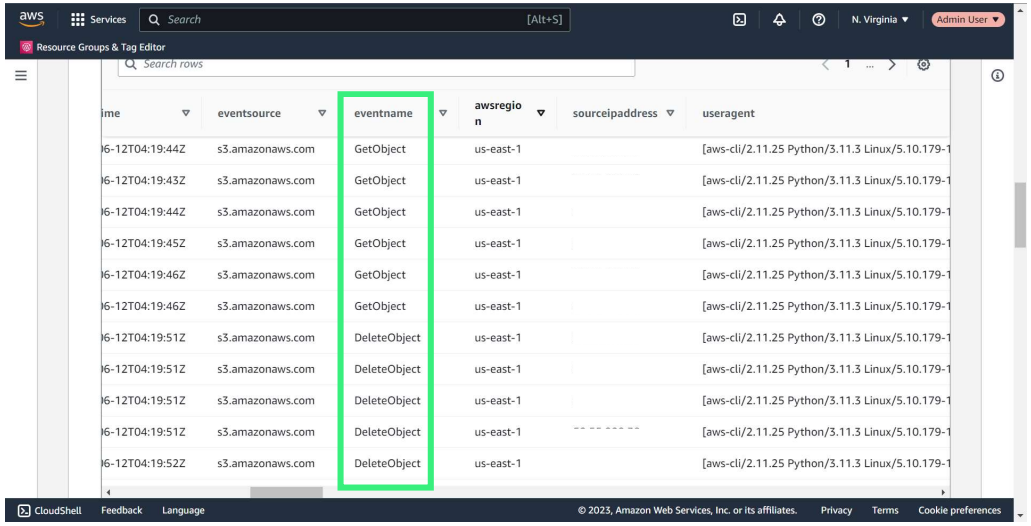
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 1:	To determine if tdir-workshop-jstiles-dev took any other objects, navigate to Athena. On the Editor tab, for Query 1, enter this query and then choose **Run**.	
Step 2:	The query that we searched for, GetObject, shows a large number of results. This indicates the GetObject API was called, and the user (tdir-workshop-jstiles-dev) downloaded other objects.	

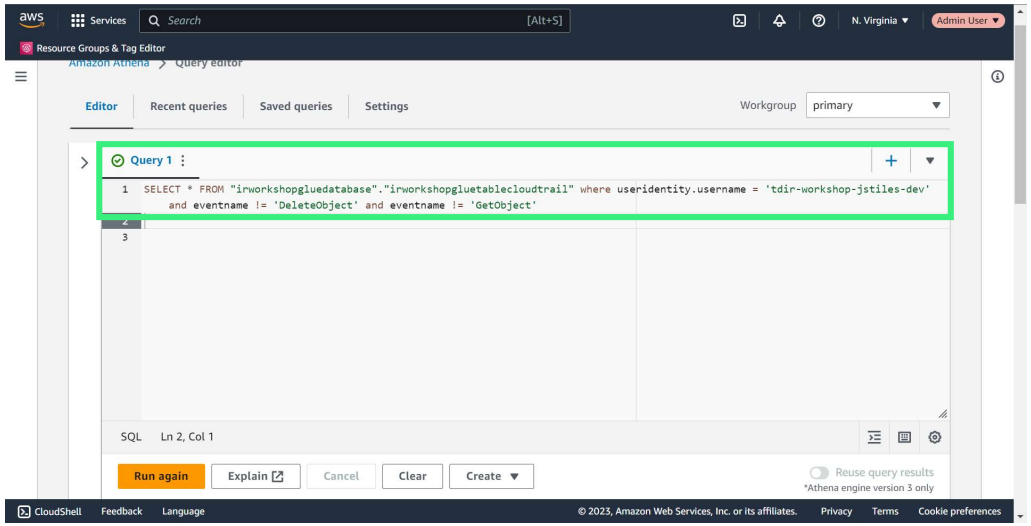
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 3:	Let's see what other activities the compromised IAM user tdir-workshop-jstiles-dev conducted. In Athena, on the Editor tab, enter this query and then choose **Run**. Use the same query in the previous step removing the where clause specifying the GetObject API. The query results will include all the username tdir-workshop-jstiles-dev activities.	
Step 4:	The query returns a large number of results. Reviewing the results shows that the DeleteObject and the GetObject API calls were used multiple times. This proves that the IAM user tdir-workshop-jstiles-dev was used to retrieve and delete numerous objects.	

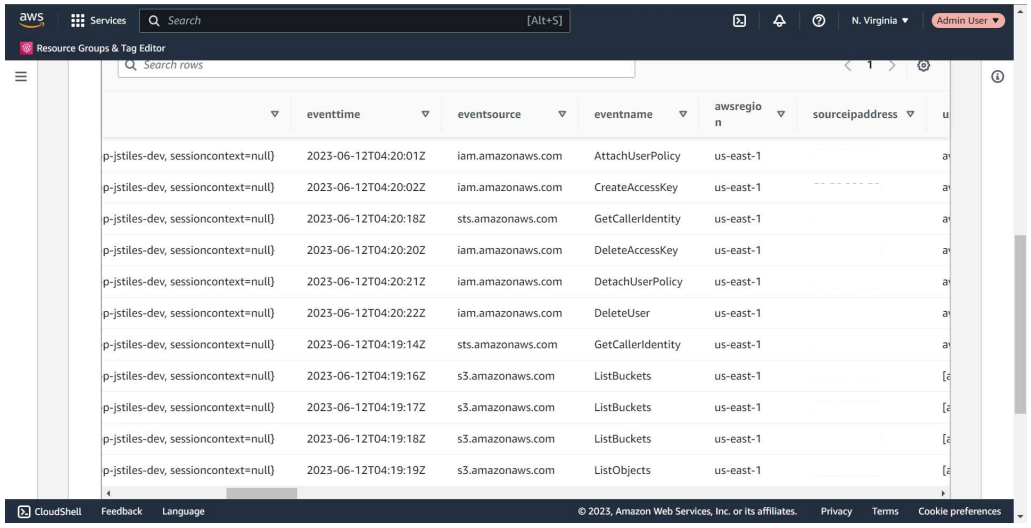
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 5:	We can reduce the number of results by eliminating these two API calls from our query. In Athena, on the Editor tab, enter **SELECT, * FROM "irworkingshopgluedatabase"."irworkshopgluetablecloudtrail" where useridentity.username = 'tdir-workshop-jstile-dev' and eventname != 'DeleteObject' and eventname != 'GetObject'** and then choose **Run again**. **Note:** We used the != operator for this query. We are reducing the number of results to not include delete object and get object by using !=.	

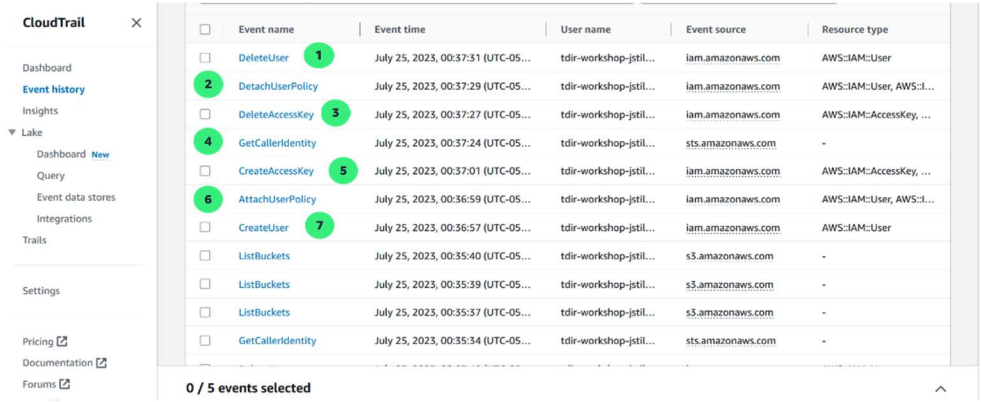
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 6:	This query provides us with 21 results—much more manageable. These results provide more detail about the scope of the unauthorized activity. Based on the data in the eventname column, the following additional activities were performed (don't forget we already filtered out DeleteObject and GetObject from the query results). The last seven API calls are interesting, and we will look further into these.	

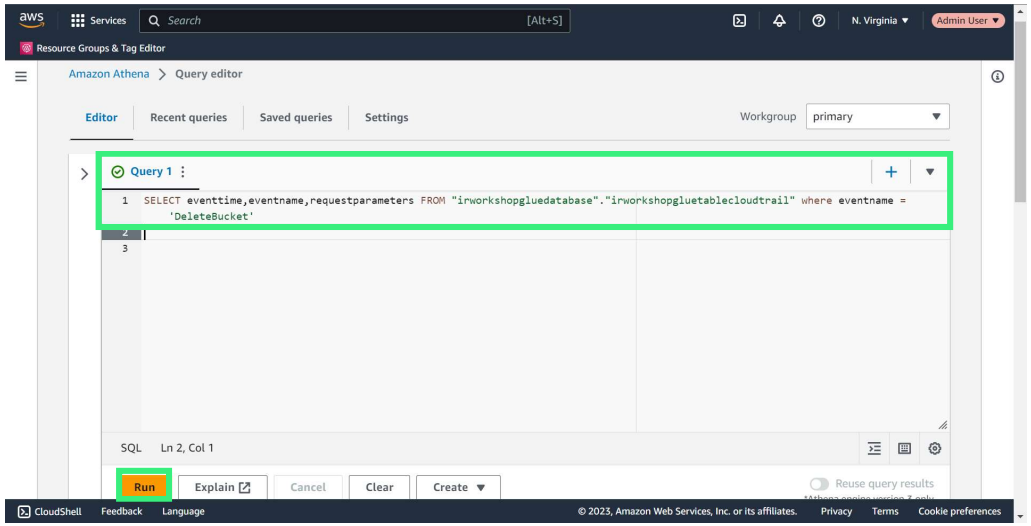
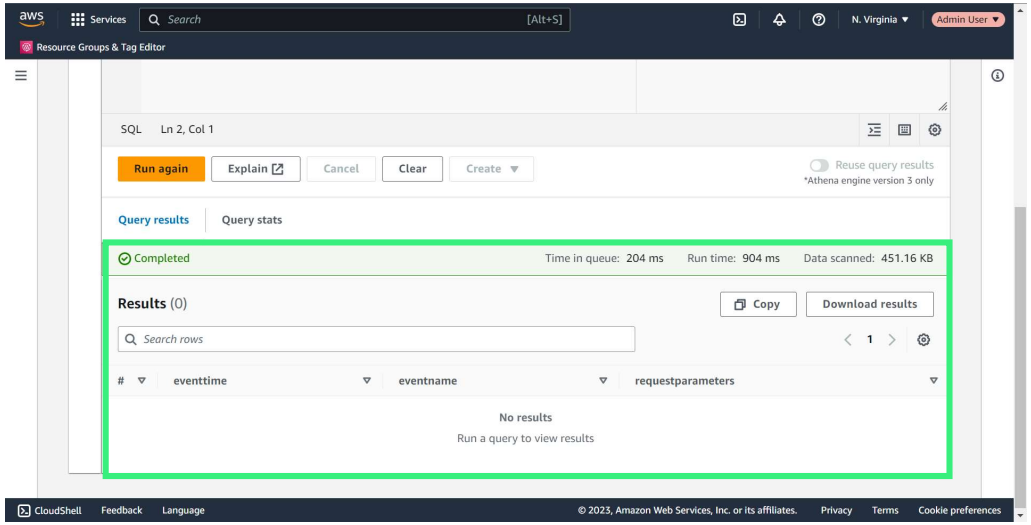
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 7:	Let's analyze the last seven API calls and focus on the time they occurred. **Note:** The dates and times displayed in your account will be different. Use these only as an example. 1. **DeleteUser**: The tdir-workshop-rroe-dev user was deleted. 2. **DetachUserPolicy**: The AdministratorAccess policy was detached from the tdir-workshop-rroe-dev IAM user. 3. **DeleteAccessKey**: An access key for tdir-workshop-rroe-dev was deleted. 4. **GetCallerIdentity**: The GetCallerIdentity API was called by the tdir-workshop-jstiles-dev IAM user. 5. **CreateAccessKey**: An access key was created for tdir-workshop-rroe-dev. 6. **AttachUserPolicy**: The AdministratorAccess policy was attached to tdir-workshop-rroe-dev. 7. **CreateUser:** The user tdir-workshop-rroe-dev was created. Based on the last seven API calls, the tdir-workshop-jstiles-dev created the IAM user tdir-workshop-rroe-dev. This gives them administrative privileges just before the IAM user tdir-workshop-rroe-dev was used to create the new bucket in the AWS account.	

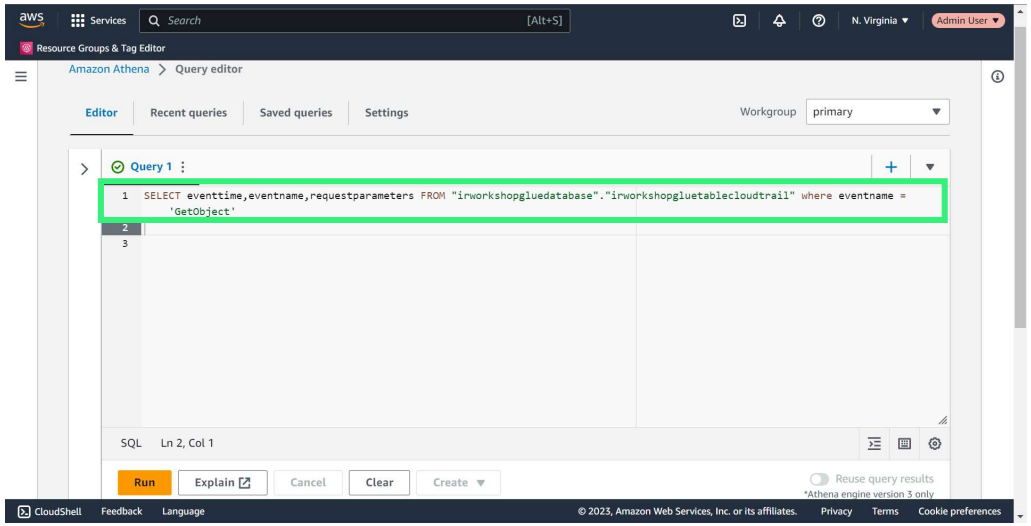
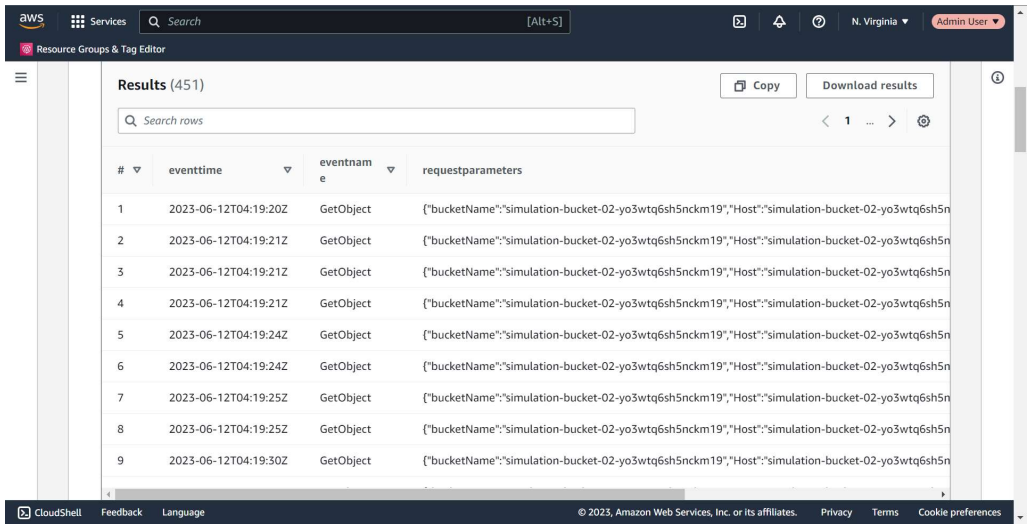
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 1:	Now that we've narrowed the scope of the incident and identified which users were involved, let's search for other buckets that might have been deleted. In Athena, on the Editor tab, enter **SELECT eventtime, eventname, requestparameters FROM"irworkingshopgluedatabase"."irworkshopgluetablecloudtrail" where eventname = 'DeleteBucket'**, and then choose **Run**.	
Step 2:	There are no results for bucket deletions. Let's see now if there were objects download from an S3 bucket.	

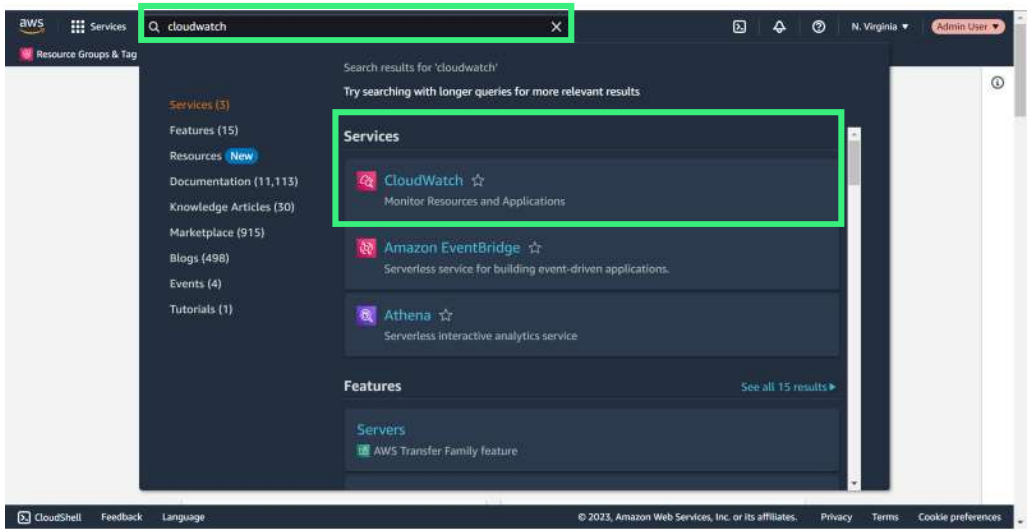
Analyze

Use Athena to review other events performed by the second IAM user we discovered.

Steps	Content	Visual
Step 3:	We can search for any objects that were download from S3 buckets by using this query in Athena.	
Step 4:	The results confirm that multiple objects were retrieved from the simulation-bucket-02-* bucket. Next, we will search for the number the of bytes downloaded from the simulation-bucket-02-* using CloudWatch.	

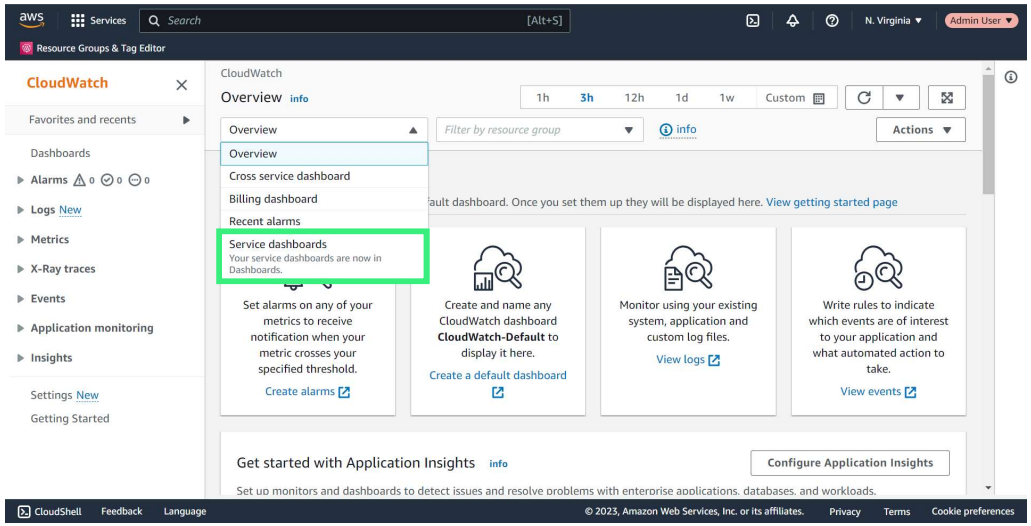
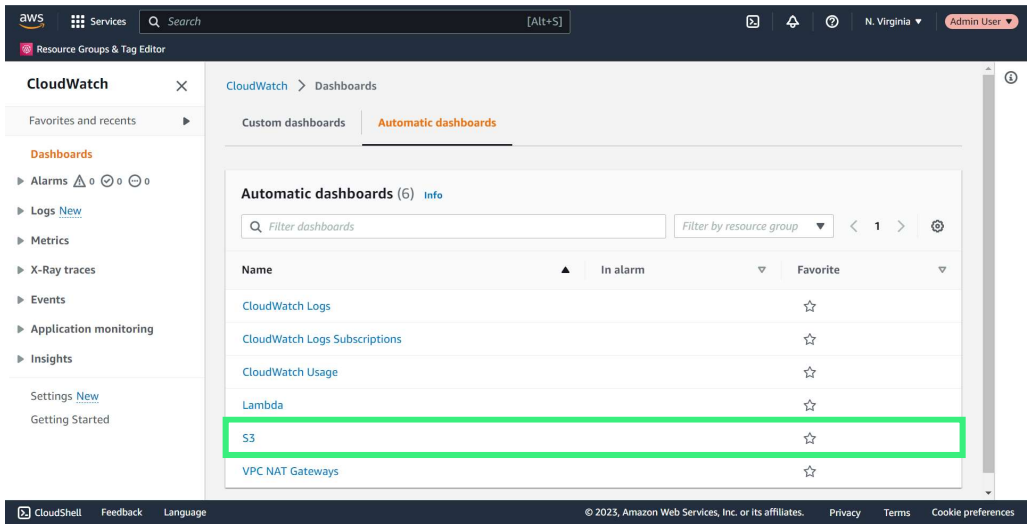
Analyze

Use CloudWatch to search for bytes downloaded from S3 buckets.

Steps	Content	Visual
Step 1:	To access CloudWatch to search for the number of bytes downloaded from the simulation-bucket-02-* bucket: From the AWS Management Console: 1. In the search field, enter **CloudWatch**. 2. From the Services list, choose **CloudWatch**.	

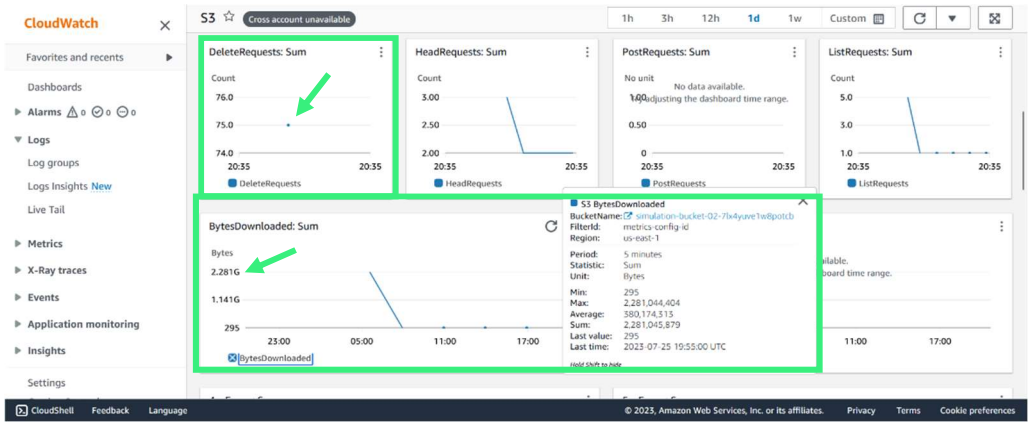
Analyze

Use CloudWatch to search for bytes downloaded from S3 buckets.

Steps	Content	Visual
Step 2:	The CloudWatch service page displays. From the Overview menu, select **Service dashboards.**	
Step 3:	The Automatic dashboards page displays. To review the simulation-bucket-02-* S3 bucket, choose **S3**.	

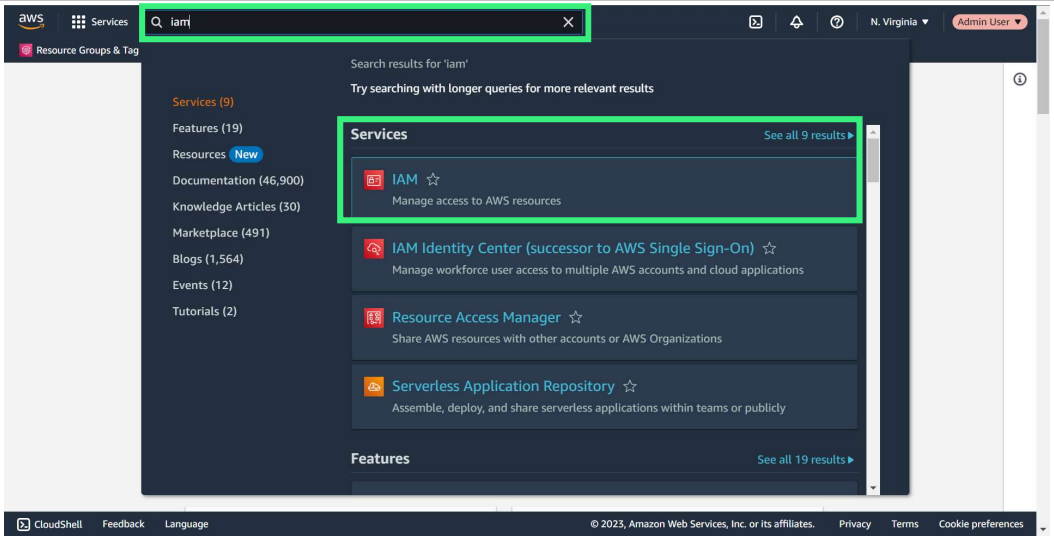
Analyze

Use CloudWatch to search for bytes downloaded from S3 buckets.

Steps	Content	Visual
Step 4:	The Amazon S3 dashboard displays. Scroll down to the S3 DeleteRequests: Sum for the simulation- bucket-02-*. This shows 75 delete requests. The BytesDownloaded: Sum section shows that 2.28GB of data was downloaded from the bucket. Now that we've identified the scope of the incident from the amount of data downloaded, let's check on any other sensitive files.	

Contain

Use the IAM console to rotate the compromised user's credentials.

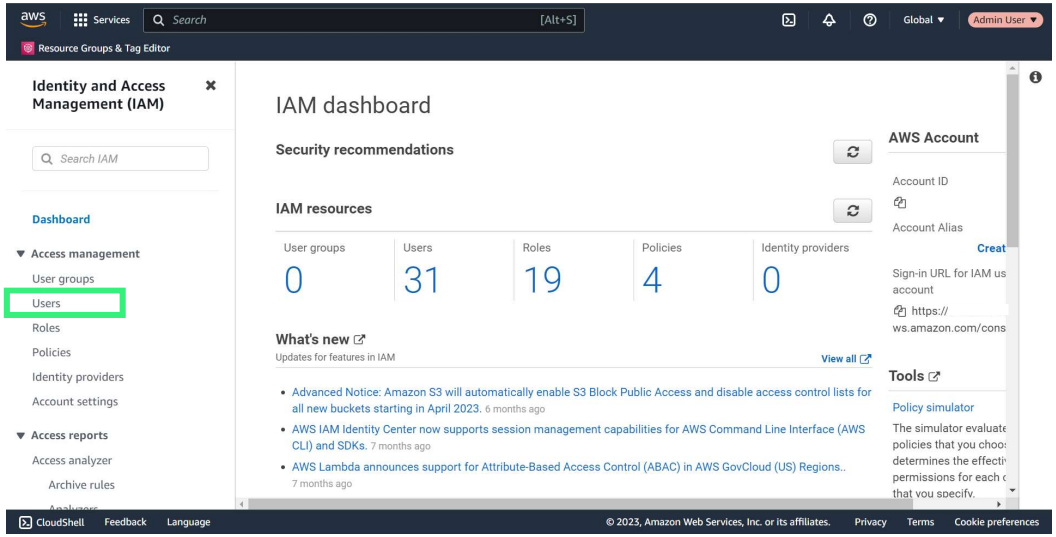
Steps	Content	Visual
Step 1:	To navigate to the IAM console: 1. In the search field, enter **IAM.** 2. Under Services, choose **IAM**.	

Contain

Use the IAM console to rotate the compromised user's credentials.

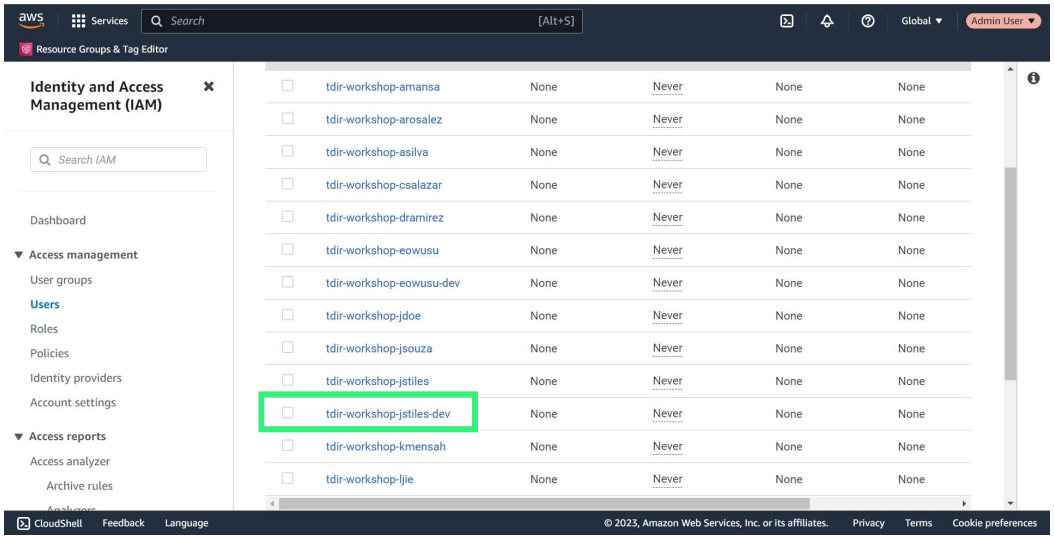
Step 2:

The IAM dashboard displays. Under Access management, choose **Users**.



Step 3:

A list of users displays on the screen. Choose **tdir-workshop-jstiles-dev.**

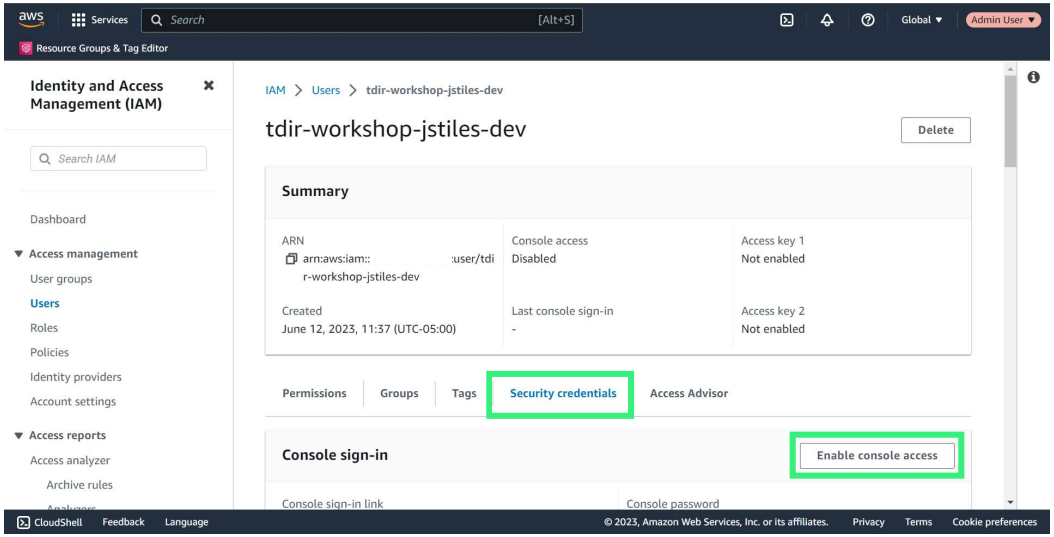


Contain

Use the IAM console to rotate the compromised user's credentials.

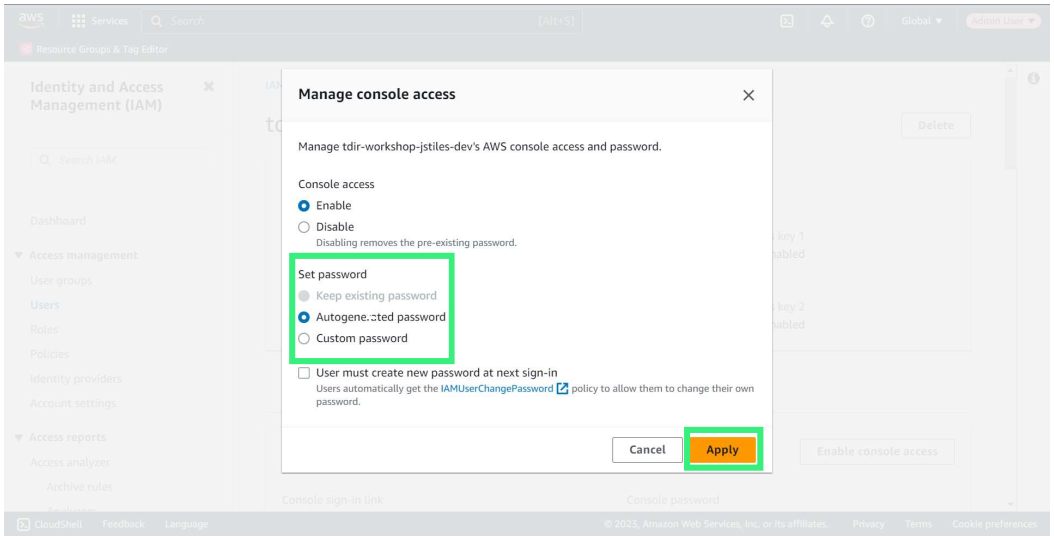
Step 4:

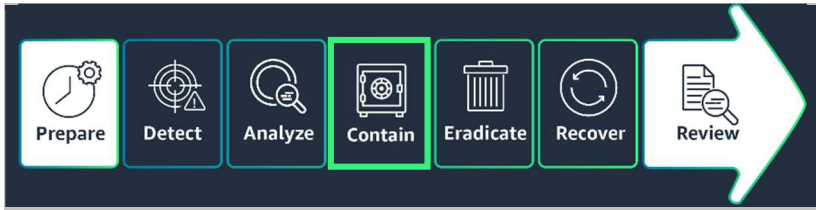
The user details page displays. Choose **Security credentials**, and then choose **Enable console access**.



Step 5:

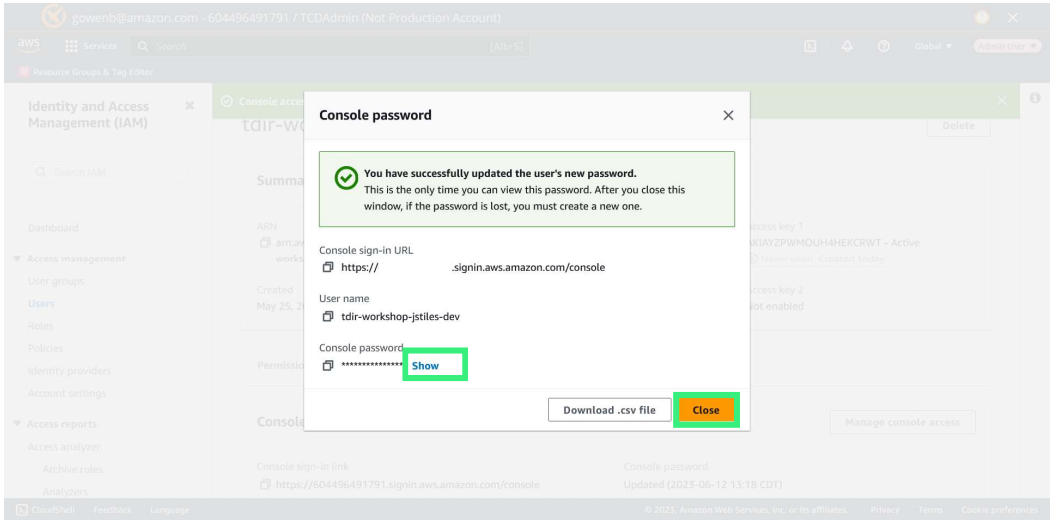
The manage console access window displays. Under Set password, choose **Autogenerated password**, and then choose **Apply**.

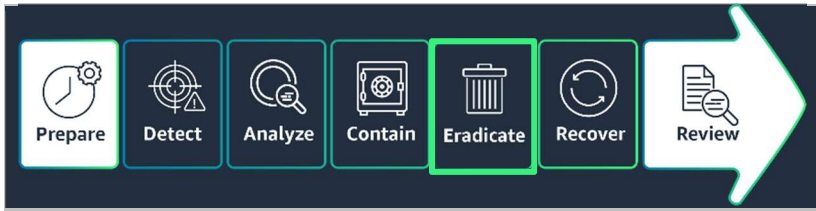




Contain

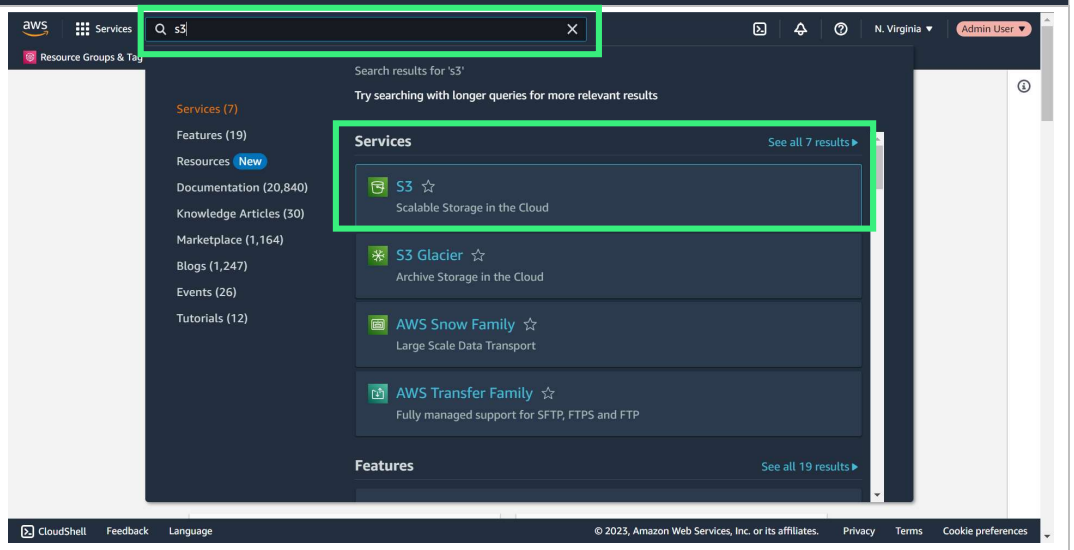
Use the IAM console to rotate the compromised user's credentials.

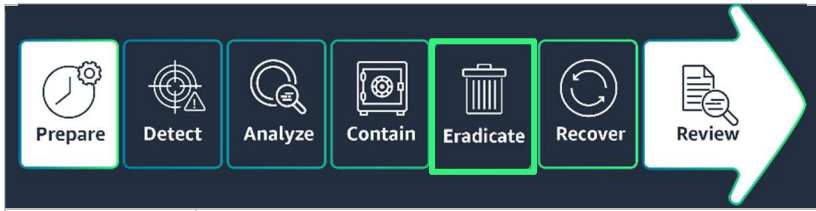
Step 6:	The console password window displays, indicating that we've successfully updated the user's new password. To view the console password, choose **Show**. Otherwise, choose **Close** to close the window. **Note:** Because the tdir-workshop-jstiles-dev user had previously deleted the tdir-workshop-rroe-dev user, we don't need to take any actions on that user.	



Eradicate

Use Amazon S3 console to remove the ransom note and delete the new bucket.

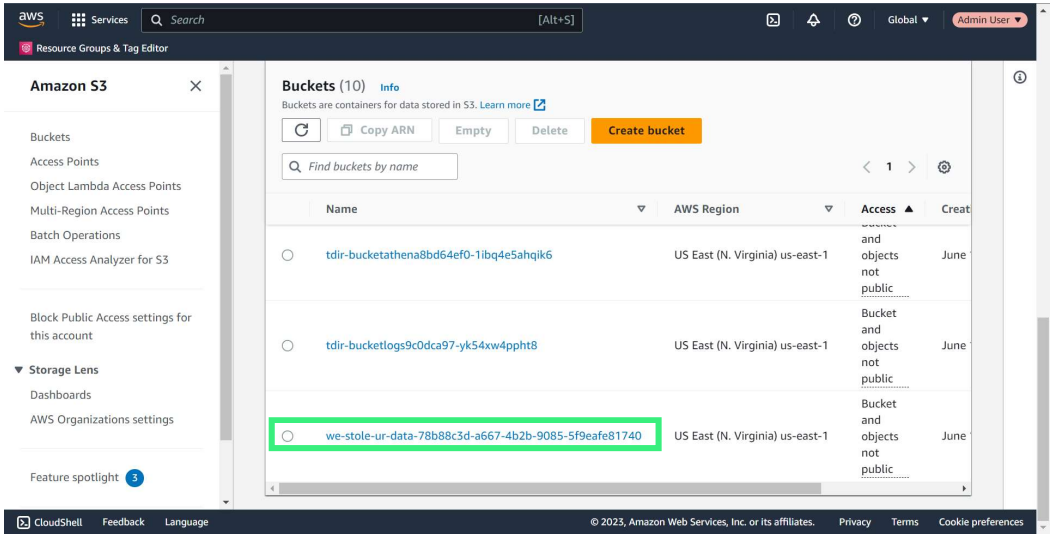
Steps	Content	Visual
Step 1:	From the AWS Management Console: 1. In the search field, enter **S3**. 2. Under the Services list, choose **S3.**	



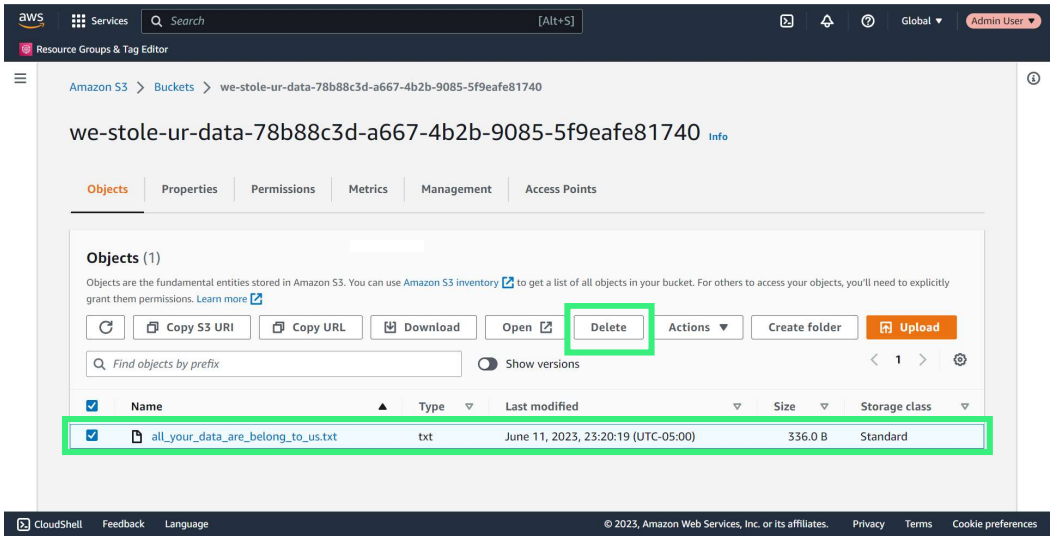
Eradicate

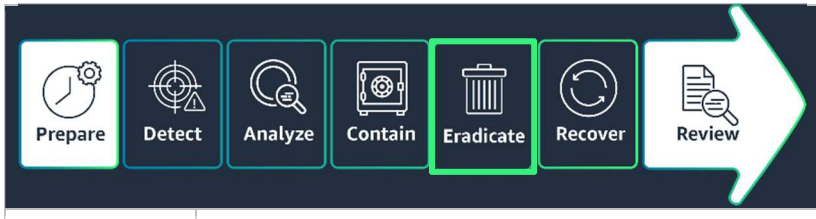
Use Amazon S3 console to remove the ransom note and delete the new bucket.

Step 2: Scroll down to the Buckets section, and choose the **we-stole-ur-data-*** bucket.



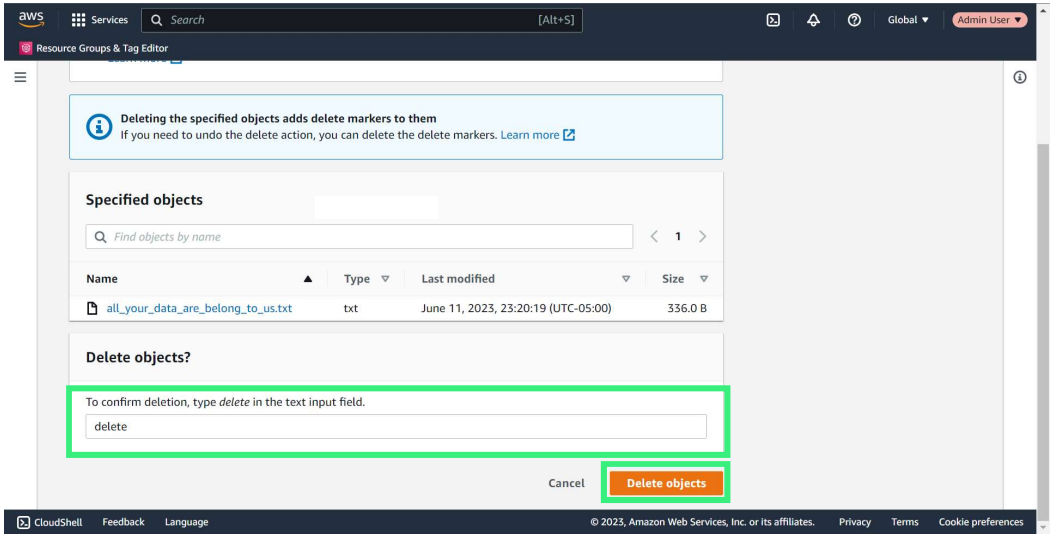
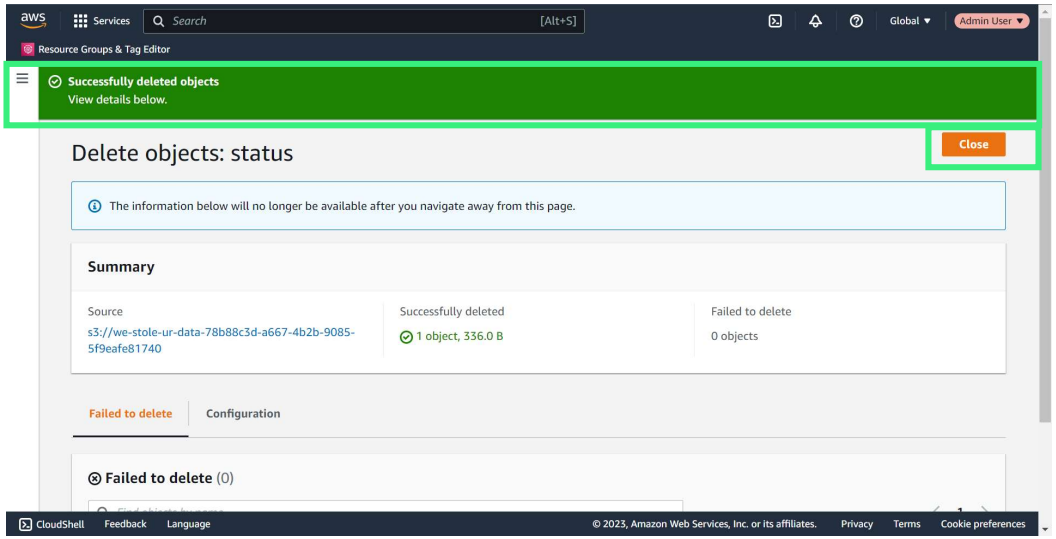
Step 3: The bucket Objects page displays. Select **all_your_data_are_belong_to_us.txt** and then choose **Delete.**

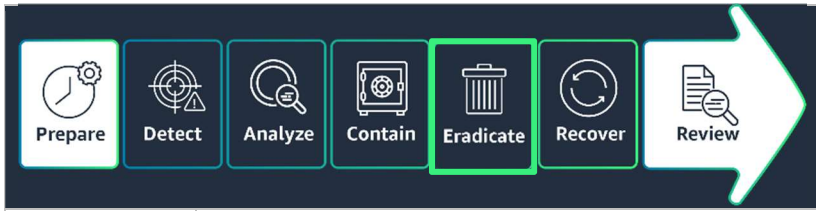




Eradicate

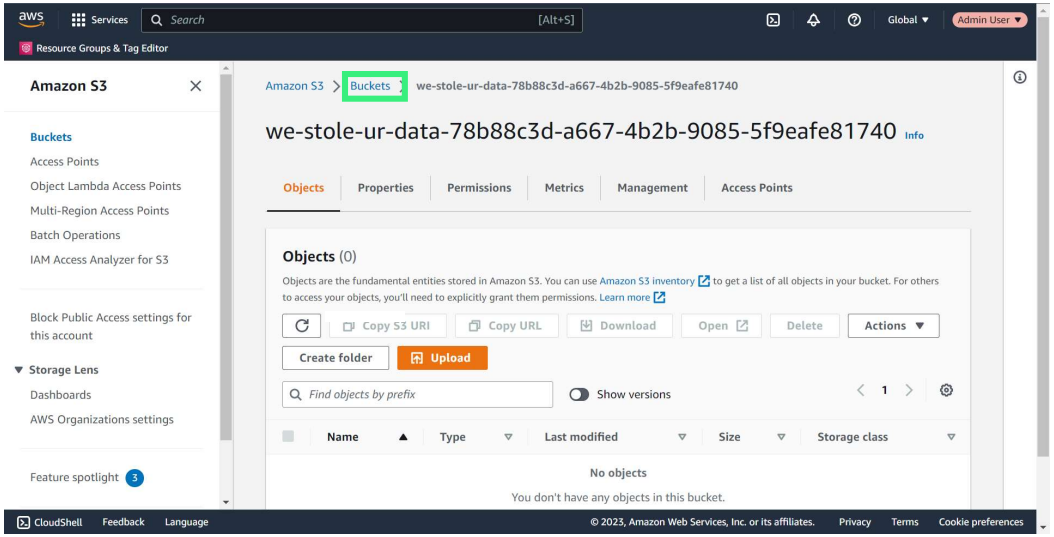
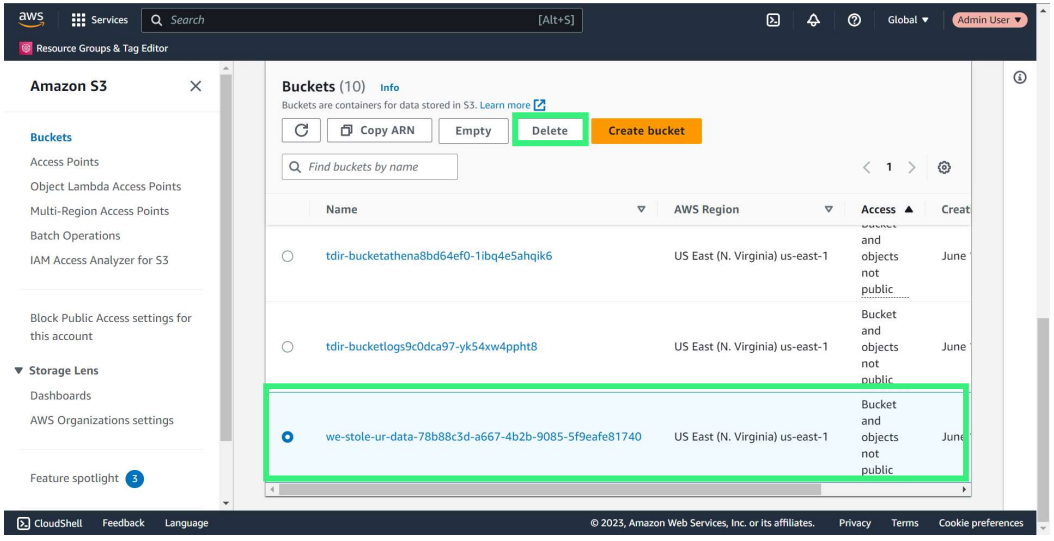
Use Amazon S3 console to remove the ransom note and delete the new bucket.

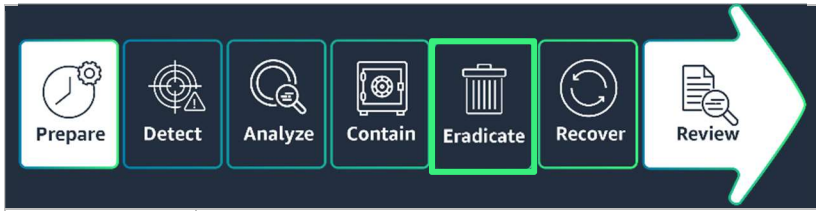
Step 4:	The Delete objects page displays. Scroll to the **Delete objects?** section. In the text field, enter **delete**, and then choose **Delete objects**.	
Step 5:	The Delete objects: status page displays with a status message indicating that the object has been deleted. Choose **Close**.	



Eradicate

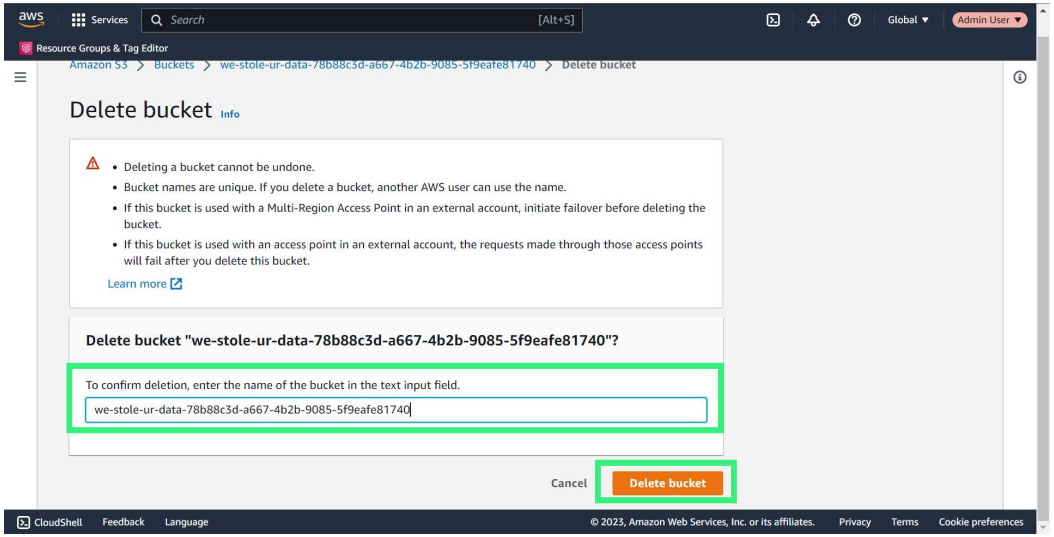
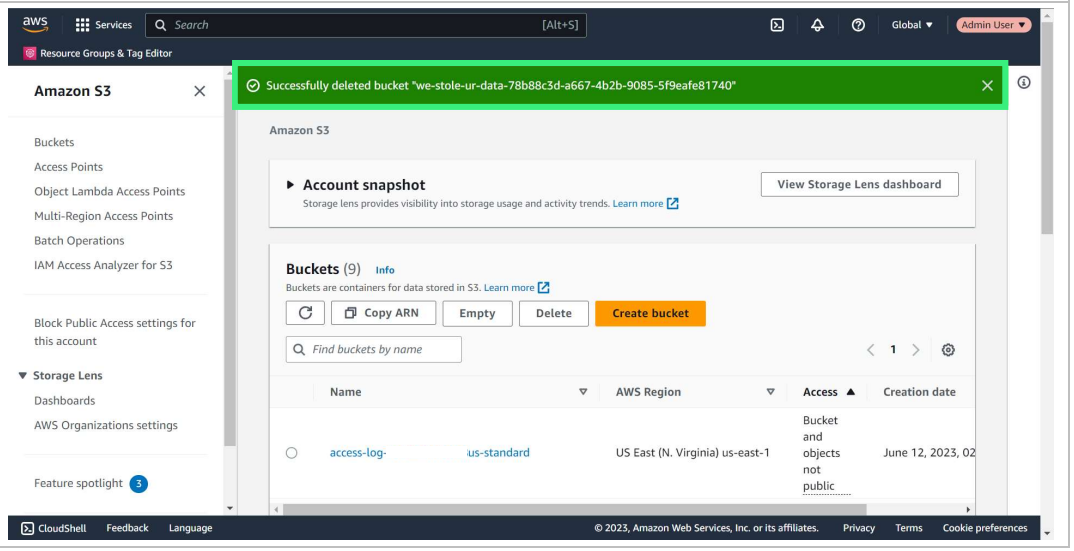
Use Amazon S3 console to remove the ransom note and delete the new bucket.

Step 6:	The we-stole-ur-data-* bucket page displays. Choose **Buckets** from the navigation menu.	
Step 7:	The Buckets page displays. Choose **we-stole-ur-data-* bucket**. Then, choose **Delete**.	



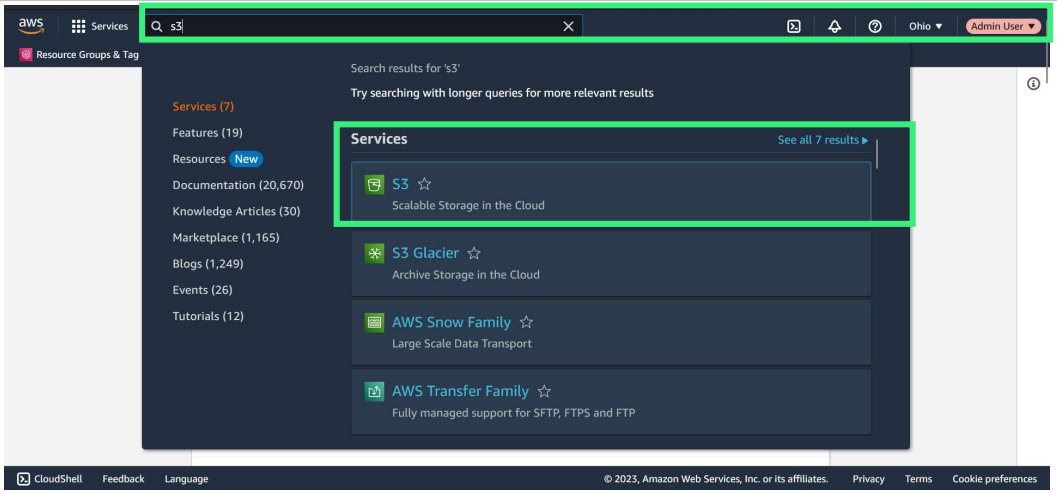
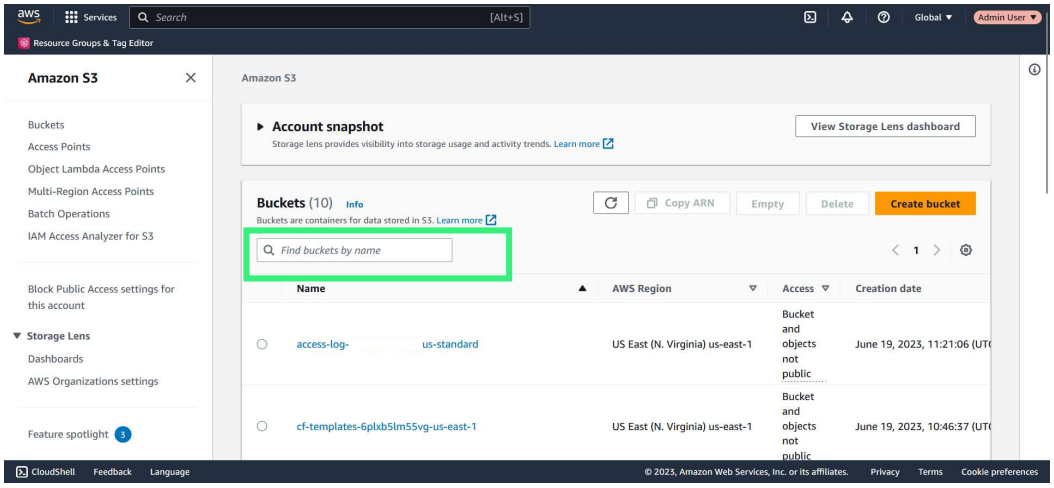
Eradicate

Use Amazon S3 console to remove the ransom note and delete the new bucket.

Step 8:	The Delete bucket page displays. In the last field on the page, enter **we-stole-ur-data-*** and then choose **Delete bucket**.	
Step 9:	The Buckets page displays with a status message indicating that the bucket was successfully deleted.	

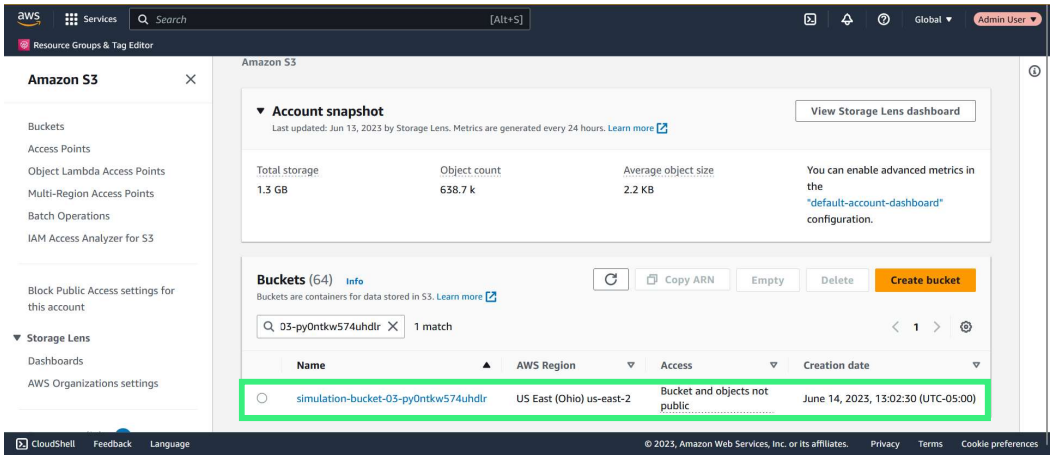
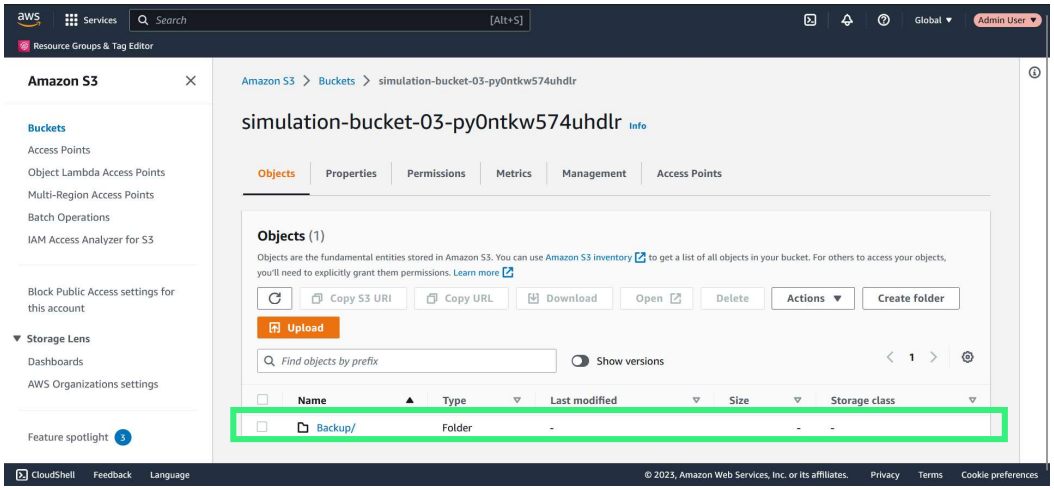
Recover

Use the Amazon S3 console to restore files to a previous version.

Steps	Content	Visual
Step 1:	Earlier, we confirmed that the **credit-card-data.csv** object was taken and deleted from the S3 bucket simulation-bucket-03-8lonc4g9dwxzrbim located in the backup/customers/payment_information/ folder. Let's restore this file. From the AWS Management Console: 1. In the search field, enter **S3**. 2. Under Services, choose **S3**.	
Step 2:	The S3 Buckets page displays. In the Find buckets by name text field, enter **simulation-bucket-03-8lonc4g9dwxzrbim**.	

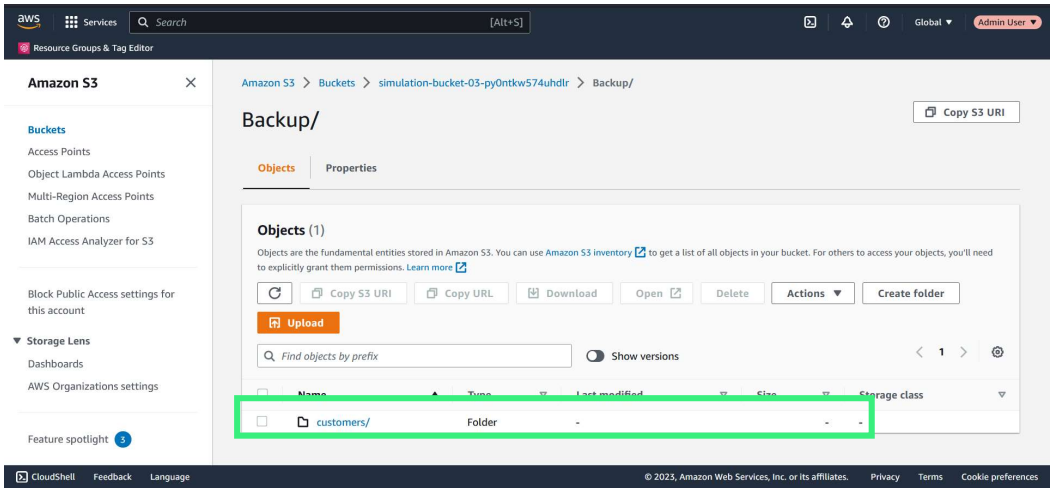
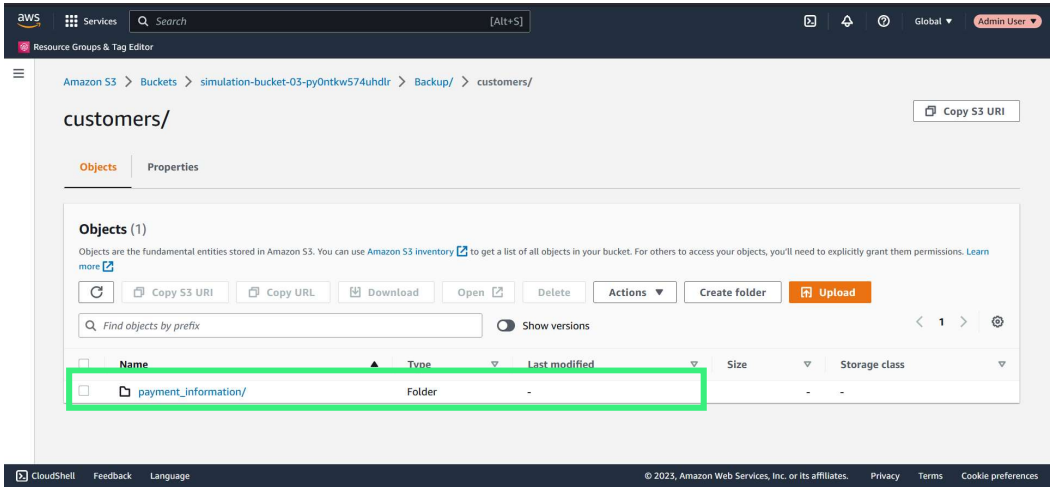
Recover

Use the Amazon S3 console to restore files to a previous version.

Steps	Content	Visual
Step 3:	In the Buckets section, choose **simulation-bucket-03-8lonc4g9dwxzrbim**.	
Step 4:	On the Objects tab, choose **Backup/**.	

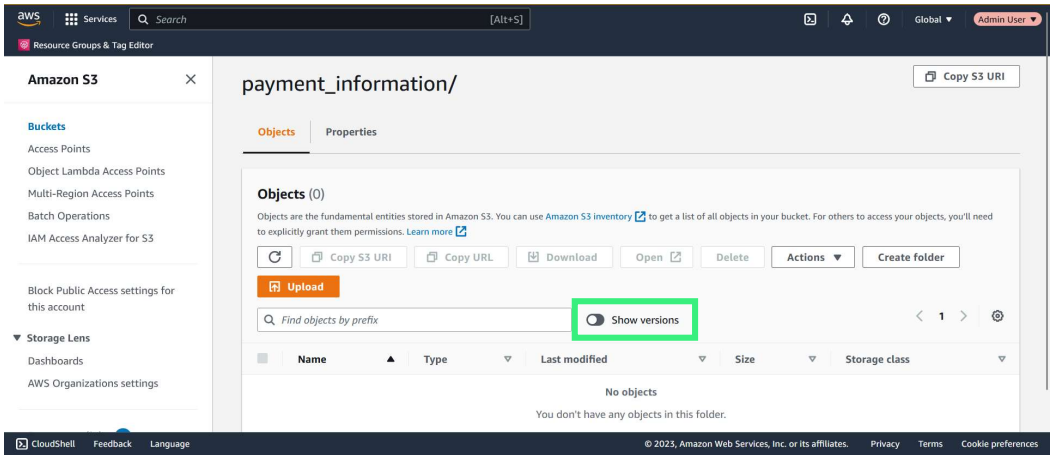
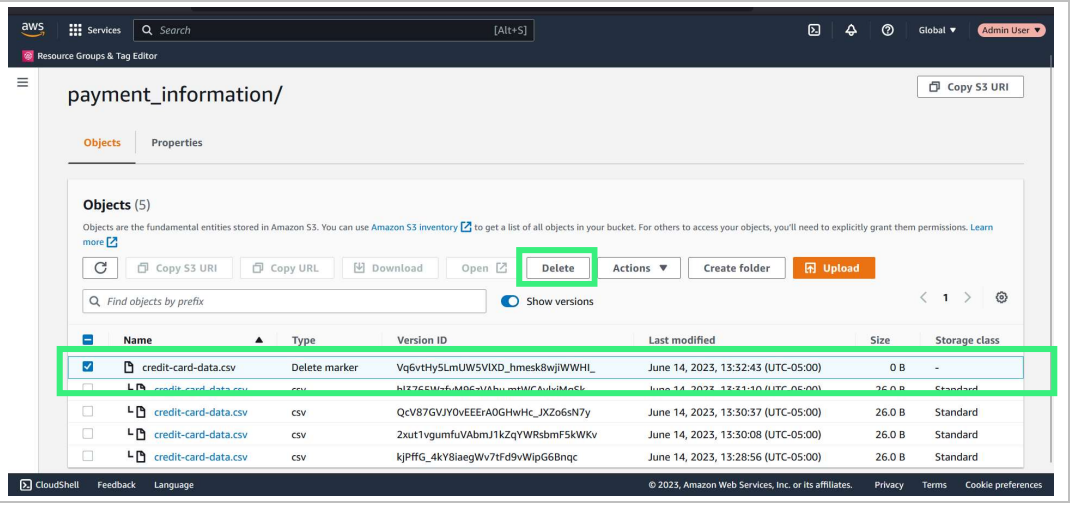
Recover

Use the Amazon S3 console to restore files to a previous version.

Steps	Content	Visual
Step 5:	In the Backup/ folder, choose **customers/**.	
Step 6:	In the customers/ folder, choose **payment_information/**.	

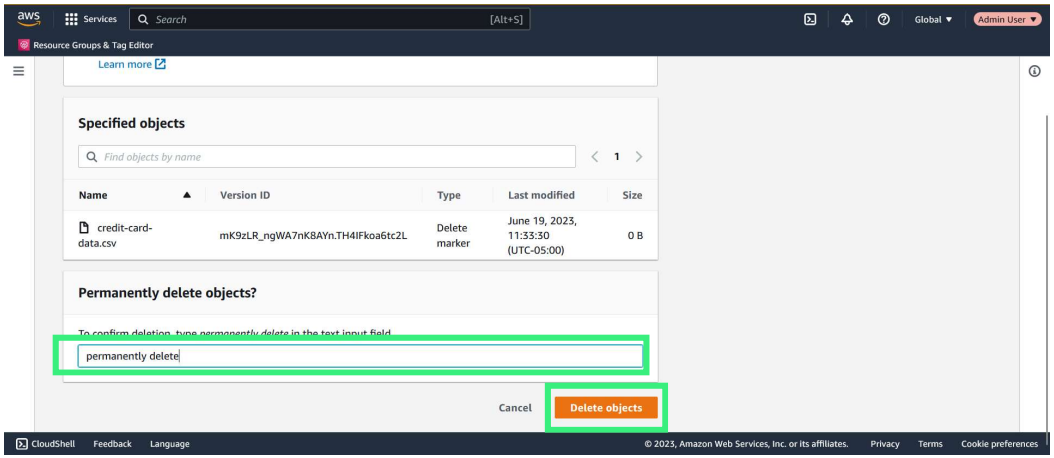
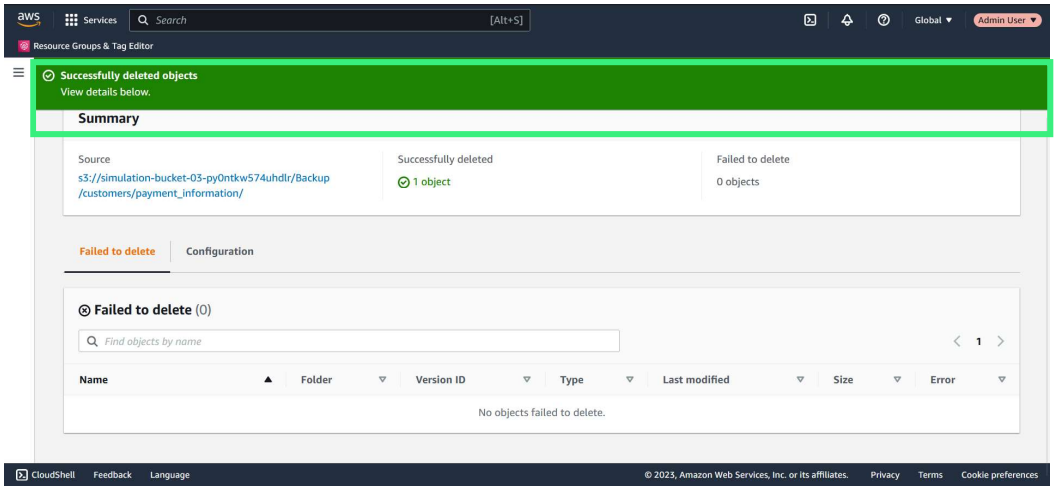
Recover

Use the Amazon S3 console to restore files to a previous version.

Steps	Content	Visual
Step 7:	The payment information folder displays. Because the threat actors deleted the credit-card-data.csv file, this folder no longer shows that object. To see previous versions of objects in this folder, select **Show versions**.	
Step 8:	Select **credit-card-data.csv** with **Delete marker** type, and then choose **Delete**. **Note:** A delete marker in Amazon S3 is a placeholder (or marker) for a versioned object that was named in a simple delete request.	

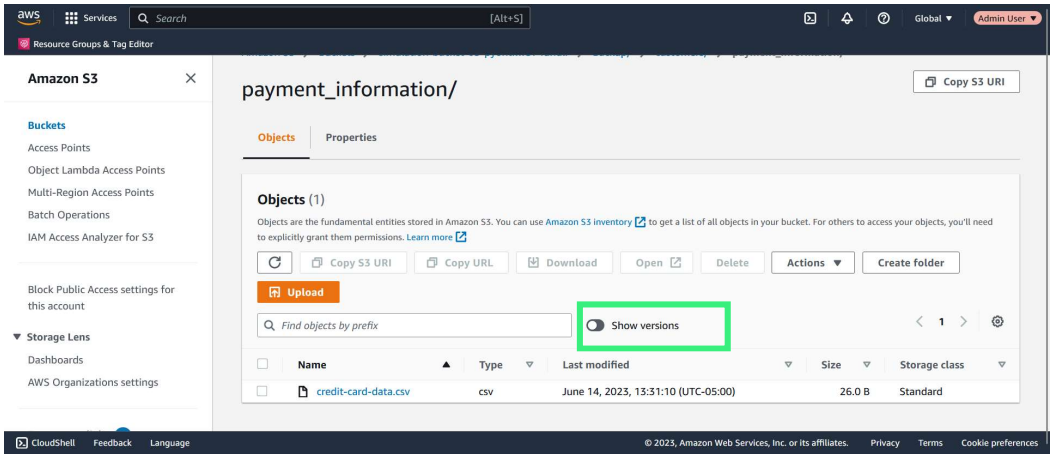
Recover

Use the Amazon S3 console to restore files to a previous version.

Steps	Content	Visual
Step 9:	The delete objects page displays. Scroll down to the Permanently delete objects? section. Enter **permanently delete,** and then choose **Delete objects**.	
Step 10:	The Delete objects: status page displays with a status message indicating that objects have been deleted. Choose **Close**.	

Recover

Use the Amazon S3 console to restore files to a previous version.

Steps	Content	Visual
Step 11:	Now, in the Objects section of the payment information screen, select **Show versions.**	
Step 12:	You can now review the files in the folders. The payment_information folder now shows the credit-card-data.csv object, confirming that it has been successfully restored.