Nubilum-2: Walkthourgh

Wednesday, August 7, 2024 12:15 AM

Story: Leading telecoms provider Forela uses AWS S3 as an essential part of their infrastructure. They can deploy applications quickly and do effective analytics on their sizable dataset thanks to it acting as both an application storage and a data lake storage. Recently, a user reported an urgent issue to the helpdesk: an inability to access files within a designated S3 directory. This disruption has not only impeded critical operations but has also raised immediate security concerns. The urgency of this situation demands a security-focused approach. Reports of a misconfigured S3 Bucket policy for the forela-fileshare bucket, resulting in unintended public access, highlight a potential security vulnerability that calls for immediate corrective measures. Consequently, a thorough investigation is paramount.

Task 1: What was the originating IP address the Threat Actor (TA) used to infiltrate the Forela's AWS account?

- What was the originating IP address the Threat Actor (TA) used to infiltrate the Forela's AWS account?

First, inspect the structure of your CloudTrail logs to understand the fields available with the command:
jq '.[0]' Merged.json

- Now we should to extract events related to **GetObject, PutObject, ListBuckets, and PutBucketPolicy to a new file.**
  We will do that with the command:
  jq '.[] .Records[] | select(.eventName == "GetObject" or .eventName == "PutObject" or .eventName == "ListBuckets" or .eventName == "PutBucketPolicy")' Merged.json > filtered_events.json

- After we filtered the suspicious API calls, let's extract the sourceIP from the filtered events via the command:
  cat filtered_events.json| grep -i 'sourceip' | sort | uniq

We identified three IP addresses that performed suspicious API calls.
Upon examining the operations of each, we found that the IP address '54.242.59.197' is associated with a Linux system and accessed suspicious files.

Task 2: What was the time, filename, and Account ID of the first recorded s3 object accessed by the TA?

- First, I extracted all the events which related to the attacker IP via the command:
  jq '.[] .Records[] | select(.sourceIPAddress == "54.242.59.197")' Merged.json > attacker_events1.json

- Now, we should search for access to the first bucket.
  I found several operations by 'Anonymous' user with 'AccessDenied
  I used 'less' command and filtered by 'Anonymous' operations and found the answer:

Task3: How many Access Keys were compromised, at a minimum?

- Pretty simple, I filtered 'accessKeyId' and found the Unique's access keys:

Task4: The TA executed a command to filter EC2 instances. What were the name and value used for filtering?

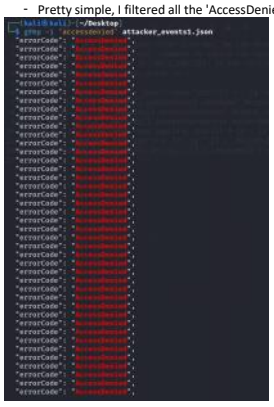
- I look for API calls which related to 'Instances' and found the following events:

- Via noterpad++ I searched for these specific logs and found the answer:

```
"eventVersion": "1.08",
"userIdentity": {
  "type": "IAMUser",
  "principalId": "AIDA52GPOBQCO7KTCFGXL",
  "arn": "arn:aws:iam::949622803460:user/prod-EC2-readonly",
  "accountId": "949622803460",
  "accessKeyId": "AKIA52GPOBQCLJPSAI53",
  "userName": "prod-EC2-readonly"
},
"eventTime": "2023-11-02T15:05:58Z",
"eventSource": "ec2.amazonaws.com",
"eventName": "DescribeInstances",
"awsRegion": "us-east-2",
"sourceIPAddress": "54.242.59.197",
"userAgent": "aws-cli/2.12.0 Python/3.11.4 Linux/6.3.0-kali1-cloud-amd64 source/x86_64.kali.2023 prompt/off command/ec2.describe-instances",
"requestParameters": {
  "instancesSet": {},
  "filterSet": {
    "items": [
      {
        "name": "instance-state-name",
        "valueSet": {
          "items": [
            {
              "value": "running"
            }
          ]
        }
      }
    ]
  }
},
```

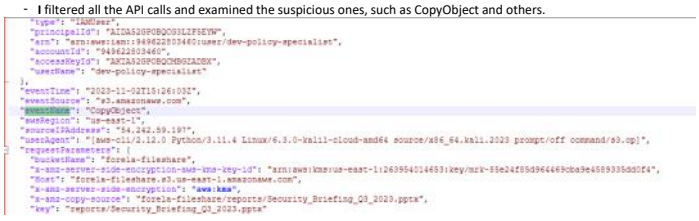

Task 5: Can you provide the count of unsuccessful discovery and privilege escalation attempts made by the TA before gaining devated access with the compromised keys?

- Pretty simple, I filtered all the 'AccessDenied' error codes:



Task 6: Which IAM user successfully gained elevated privileges in this incident?

- I filtered all the API calls and examined the suspicious ones, such as CopyObject and others.



Task7: Which event name permitted the threat actor to generate an admin-level policy?

- API call: **PutUserPolicy**


```
  "type": "IAMUser",
  "principalId": "AIDA52GPOBQCG3LZF5EYW",
  "arn": "arn:aws:iam::949622803460:user/dev-policy-specialist",
  "accountId": "949622803460",
  "accessKeyId": "AKIA52GPOBQCMBGZADBX",
  "userName": "dev-policy-specialist"
},
"eventTime": "2023-11-02T15:21:32Z",
"eventSource": "iam.amazonaws.com",
"eventName": "PutUserPolicy",
"awsRegion": "us-east-1",
"sourceIPAddress": "54.242.59.197",
"userAgent": "Boto3/1.10.2 Python/3.8.0 Linux/4.14.138-99.102.amzn2.x86_64 exec-env/AWS_Lambda_python3.8 Botocore/1.13.2 Botocore/1.31.49",
"requestParameters": {
  "userName": "dev-policy-specialist",
  "policyName": "sbhyy79zky",
  "policyDocument": "{\"Version\": \"2012-10-17\",\"Statement\": [{\"Effect\": \"Allow\",\"Action\": \"*\",\"Resource\": \"*\"}]}"
},
```


Task8: What is the name and statement of the policy that was created that gave a standard user account elevated privileges?

- Already found it on the previous log -
  sbhyy79zky,[{"Effect": "Allow","Action": "*","Resource": "*"}]


```
"userName": "dev-policy-specialist",
"policyName": "sbhyy79zky",
"policyDocument": "{\"Version\": \"2012-10-17\",\"Statement\": [{\"Effect\": \"Allow\",\"Action\": \"*\",\"Resource\": \"*\"}]}"
```


Task 9: What was the ARN (Amazon Resource Name) used to encrypt the files?

- On the previous logs we identifed the 'Copy Object' API call with server-side encryption provided by AWS KMS using the key ID
  arn:aws:kms:us-east-1:263954014653:key/mrk-85e24f85d964469cba9e4589335dd0f4
  You also can searched 'encrypt' via notepad++

```
"eventVersion": "1.09",
"userIdentity": {
  "type": "IAMUser",
  "principalId": "AIDA52GPOBQCG3LZF5EYW",
  "arn": "arn:aws:iam::949622803460:user/dev-policy-specialist",
  "accountId": "949622803460",
  "accessKeyId": "AKIA52GPOBQCMBGZADBX",
  "userName": "dev-policy-specialist"
},
"eventTime": "2023-11-02T15:26:03Z",
"eventSource": "s3.amazonaws.com",
"eventName": "CopyObject",
"awsRegion": "us-east-1",
"sourceIPAddress": "54.242.59.197",
"userAgent": "[aws-cli/2.12.0 Python/3.11.4 Linux/6.3.0-kali1-cloud-amd64 source/x86_64.kali.2023 prompt/off command/s3.cp]",
"requestParameters": {
  "bucketName": "forela-fileshare",
  "x-amz-server-side-encryption-aws-kms-key-id": "arn:aws:kms:us-east-1:263954014653:key/mrk-85e24f85d964469cba9e4589335dd0f4",
  "Host": "forela-fileshare.s3.us-east-1.amazonaws.com",
  "x-amz-server-side-encryption": "aws:kms",
  "x-amz-copy-source": "forela-fileshare/reports/Network_Security_Assessment_Report.docx",
  "key": "reports/Network_Security_Assessment_Report.docx"
},
"responseElements": {
  "x-amz-server-side-encryption": "aws:kms",
  "x-amz-server-side-encryption-aws-kms-key-id": "arn:aws:kms:us-east-1:263954014653:key/mrk-85e24f85d964469cba9e4589335dd0f4"
```


Task 10: What was the name of the file that the TA uploaded to the S3 bucket?

- I searched for "PutObject" API call and found the answer:


```
},
"eventTime": "2023-11-02T15:26:22Z",
"eventSource": "s3.amazonaws.com",
"eventName": "PutObject",
"awsRegion": "us-east-1",
"sourceIPAddress": "54.242.59.197",
"userAgent": "[aws-cli/2.12.0 Python/3.11.4 Linux/6.3.0-kali1-cloud-amd64 source/x86_64.kali.2023 prompt/off command/s3.cp]",
"requestParameters": {
  "bucketName": "forela-fileshare",
  "Host": "forela-fileshare.s3.us-east-1.amazonaws.com",
  "key": "README2DECRYPT.txt"
```


Task 11: Which IAM user account did the TA modify in order to gain additional persistent access?

- I searched the API Calls and found 'CreateAccessKey' by the compromised user to **'forela-admin'**


```
"eventVersion": "1.08",
"userIdentity": {
  "type": "IAMUser",
  "principalId": [illegible],
  "arn": "arn:aws:iam::[illegible]:user/dev-policy-specialist",
  "accountId": [illegible],
  "accessKeyId": [illegible],
  "userName": "dev-policy-specialist"
},
"eventTime": [illegible],
"eventSource": "iam.amazonaws.com",
"eventName": "CreateAccessKey",
"awsRegion": "us-east-1",
"sourceIPAddress": [illegible],
"userAgent": [illegible] command/iam.create-access-key",
"requestParameters": {
  "userName": "forela-admin"
},
"responseElements": {
  "accessKey": {
    "accessKeyId": [illegible],
    "status": "Active",
    "userName": "forela-admin",
    "createDate": [illegible]
  }
},
"requestID": [illegible],
"eventID": [illegible],
"readOnly": false,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": [illegible],
"eventCategory": "Management",
"tlsDetails": {
  "tlsVersion": "TLSv1.2",
  "cipherSuite": [illegible],
  "clientProvidedHostHeader": "iam.amazonaws.com"
}
```


Task 12:What action was the user not authorized to perform to view or download the file in the S3 bucket?
First, I attempted to search for 'AccessDenied' messages in the attacker events JSON but did not find anything related to downloads or S3 access. I then conducted the same search across all the files and discovered that the user 'forela-john' is not authorized to perform the kms:Decrypt operation on the resource.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ grep -i 'accessdenied' Merged.json -C10 | grep -i 'errormessage'
        "errorMessage": "Access Denied",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-EC2-readonly is not authorized to perform: iam:ListUsers on resource: arn:aws:iam::949622803460:user/ because no identity-based policy allows
he iam:ListUsers action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-S3-readonly is not authorized to perform: iam:ListGroupsForUser on resource: user prod-S3-readonly because no identity-based policy allows the
iam:ListGroupsForUser action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-S3-readonly is not authorized to perform: iam:ListUserPolicies on resource: user prod-S3-readonly because no identity-based policy allows the
am:ListUserPolicies action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-S3-readonly is not authorized to perform: iam:ListAttachedUserPolicies on resource: user prod-S3-readonly because no identity-based policy all
ws the iam:ListAttachedUserPolicies action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-VPC-readonly is not authorized to perform: iam:ListGroupsForUser on resource: user prod-VPC-readonly because no identity-based policy allows t
e iam:ListGroupsForUser action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-VPC-readonly is not authorized to perform: iam:ListUserPolicies on resource: user prod-VPC-readonly because no identity-based policy allows th
 iam:ListUserPolicies action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/prod-VPC-readonly is not authorized to perform: iam:ListAttachedUserPolicies on resource: user prod-VPC-readonly because no identity-based policy a
lows the iam:ListAttachedUserPolicies action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/forela-john is not authorized to perform: health:DescribeEventAggregates on resource: * because no identity-based policy allows the health:Describe
ventAggregates action",
        "errorMessage": "User: arn:aws:iam::949622803460:user/forela-john is not authorized to perform: kms:Decrypt on the resource associated with this ciphertext because the resource does not exist in this R
gion, no resource-based policies allow access, or a resource-based policy explicitly denies access",
        "errorMessage": "User: arn:aws:iam::949622803460:user/forela-john is not authorized to perform: kms:Decrypt on the resource associated with this ciphertext because the resource does not exist in this R
gion, no resource-based policies allow access, or a resource-based policy explicitly denies access",
        "errorMessage": "User: arn:aws:iam::949622803460:user/forela-john is not authorized to perform: kms:Decrypt on the resource associated with this ciphertext because the resource does not exist in this R
gion, no resource-based policies allow access, or a resource-based policy explicitly denies access",
```