**Observations from my Script Executions**

1. **\*\*Initial Python Environment Issue (`ModuleNotFoundError: No module named 'requests'`):\*\***

\* I was trying to run `python3 main.py` outside of my Python virtual environment (`venv`). This environment is where my installed packages (like `requests` and `boto3`) reside.

\* \*\*Learning:\*\* This highlights the importance of activating the correct virtual environment (`source venv/bin/activate`) before running Python scripts that depend on specific packages. This ensures my script can find its dependencies.

2. **\*\*Successful Alert Fetching (Without S3 Upload Logic Initially):\*\***

\* After activating the `venv`, one of my runs shows a successful fetch of the first page of alerts:

```

--- Attempting to fetch the first page of alerts ---

Fetching alerts page 0 with payload: {"request\_data": {"search\_from": 0, "search\_to": 100, "use\_page\_token": true, "sort": {"field": "creation\_time", "keyword": "desc"}}}

Successfully fetched alerts: {

"reply": {

"total\_count": 11,

"result\_count": 11,

"alerts": [ ... 11 alert objects ... ]

}

}

```

\* \*\*Key Takeaway:\*\* The API call to `/public\_api/v2/alerts/get\_alerts\_multi\_events/` was successful. The API reported `total\_count: 11` and `result\_count: 11`, meaning all alerts were fetched in a single page (as the total was less than the `search\_to: 100` limit). The `next\_page\_token` would be `None` in this case, correctly terminating the pagination loop.

3.  **\*\*AWS Credentials Issue During S3 Operations:\*\***

\* Several runs show the S3 operations (clearing folder, uploading alerts, verifying) failing with:

```

AWS credentials not found. S3 folder clearing failed.

...

AWS credentials not found for alert 310. S3 upload failed.

...

AWS credentials not found. S3 verification failed.

```

\* \*\*Learning:\*\* This is a classic `NoCredentialsError` from `boto3`. When the S3 client was initialized, `boto3` couldn't find AWS credentials in any of its default locations (environment variables, shared credentials file, IAM role). This happened because I had not yet hard coded them or set them as environment variables \*for that specific terminal session\*.

**4. \*\*Successful S3 Operations After Hard Coding Credentials:\*\***

\* The last provided log block shows successful S3 operations:

```

--- Attempting to clear alerts folder in s3://gbcortex/alerts/ ---

Found 11 objects to delete in s3://gbcortex/alerts/

Deleting batch 1 of 11 objects...

Batch delete successful. Deleted 11 objects.

Finished clearing alerts folder.

--- S3 folder clearing process finished ---

--- Starting alert fetching and S3 upload process ---

Fetching page: 0...

Fetching alerts with payload: {"request\_data": {"use\_page\_token": true, "search\_to": 100, "sort": {"field": "creation\_time", "keyword": "desc"}}}

Fetched 11 alerts on page 0.

--- Starting S3 upload for 11 alerts on page 0 ---

Processing alert 1/11 on page 0...

Attempting to upload alert 310 to s3://gbcortex/alerts/310.json

Successfully uploaded alert 310 to S3.

... (10 more successful uploads) ...

--- Finished S3 upload for alerts on page 0 ---

Next page token for page 0: None

No more pages to fetch or no alerts on the last page.

--- Finished alert fetching and S3 upload. Total alerts uploaded: 11 ---

--- Verifying alerts in S3 bucket 'gbcortex' ---

Found 11 alert objects in s3://gbcortex/alerts/

--- S3 verification complete ---

```

\* \*\*Learning:\*\* This confirms that once `boto3` had access to valid AWS credentials (which I provided by editing `main.py` to hardcode them for the demo), all S3 operations (clearing the `alerts/` folder, uploading each of the 11 alerts, and verifying the count) worked as intended.

\* The `delete\_objects` and `put\_object` calls were successful.

\* The verification step correctly counted 11 alert objects in S3.

5.  **\*\*NameError (`AWS\_ACCESS\_KEY\_ID` not defined):\*\***

\* One of the execution attempts shows: `NameError: name 'AWS\_ACCESS\_KEY\_ID' is not defined`.

\* This likely happened during an intermediate stage of my manual edits where I tried to use these variables in `boto3.client()` initialization before they were actually defined as global constants in my script. The final version of the file \*does\* define them, resolving this.

**\*\*Summary of Learnings and How It Relates to the Exercise:\*\***

\* **\*\*API Interaction is Key:\*\*** The core of this exercise was successfully interacting with the Cortex XSIAM API. I identified the correct v2 endpoint, headers (`Authorization` and `x-xdr-auth-id`), and payload structure for pagination and sorting. The initial `500 Internal Server Error` was likely due to using an incorrect API hostname or payload structure, which I debugged.

\* **\*\*Cloud Storage:\*\*** Uploading to my AWS S3 using `boto3`.

\* **\*\*Credentials Matter:\*\*** The `NoCredentialsError` is a very common issue when working with AWS SDKs. \*\*For my presentation, stress the importance of secure credential management (environment variables, IAM roles) over hardcoding.\*\*

\* \*\***\*Iterative Development & Debugging:\*\*** I went through several iterations to fix issues:

\* Incorrect API endpoints/payloads.

\* The `ModuleNotFoundError` (Python environment).

\* `NoCredentialsError` (AWS auth).

\* Linter errors.

This iterative process of coding, testing, observing logs, and refining is fundamental to development.

\*  **\*\*Logging for Clarity:\*\*** The detailed logs I added were crucial in seeing what the script was doing at each step, what data it was sending, what responses it received, and where errors occurred. This is invaluable for demos and for debugging.

\*\*Covering Requirements \*\*

\* **\*\*Step 2: Pull Alerts from Cortex XSIAM:\*\***

\* \*\*API Endpoint Identification:\*\* `POST /public\_api/v2/alerts/get\_alerts\_multi\_events/` is correctly used.

\* \*\*API Calls & Pagination:\*\* The logs show my script fetching page 0. Since `total\_count` was 11 and `search\_to` was 100, it correctly determined there were no more pages (`Next page token for page 0: None`). If there were more than 100 alerts, my `while True` loop with `next\_page\_token` would have handled it.

\* \*\*Documentation:\*\* The logs themselves provide a live "document" of the request payload. I can use this for my presentation. The alert JSON structure is also visible in one of the earlier successful fetch logs.

\*  **\*\*Step 3: Store Alerts in Cloud Storage:\*\***

\* \*\*Cloud Service:\*\* AWS S3 (`gbcortex` bucket).

\* \*\*Script for Upload:\*\* `main.py` with `boto3`. The logs clearly show the "Attempting to clear alerts folder," "Attempting to upload alert...", and "Successfully uploaded alert..." messages.

\* \*\*Structured Format:\*\* Alerts are stored as individual JSON files (e.g., `alerts/310.json`).

\* \*\*Documentation:\*\* The script (`clear\_s3\_alerts\_folder`, `upload\_alert\_to\_s3` functions) and the logs serve as documentation.

**\*\*Presentation Points Highlighted by This Output:\*\***

\* \*\*Virtual Environments:\*\* Briefly mention their importance for managing dependencies.

\* \*\*Credential Management:\*\* Use the "AWS credentials not found" errors as a talking point to emphasize how critical proper credential setup is and why hardcoding (as done for the demo) is insecure in real applications.

\* \*\*Successful API Interaction:\*\* Show the log where alerts are fetched successfully from the v2 endpoint. Point out the `total\_count` and `result\_count`.

\* \*\*S3 Workflow:\*\*

\* Show the logs for clearing the S3 `alerts/` folder.

\* Show the logs for uploading individual alerts, highlighting the unique object keys based on `alert\_id`.

\* Show the verification step confirming the number of objects in S3.

\* \*\*Alert Schema:\*\* Refer to the example alert JSON that was printed in one of the earlier successful fetch logs. Discuss a few key fields.

Schema

{

"action\_process\_instance\_id": null,

"actor\_process\_instance\_id": [

"AduNGcatb/YAABiAAAAAAA=="

],

"agent\_os\_sub\_type": "Windows Server 2025 [10.0 (Build 26100)]",

"fw\_app\_category": null,

"fw\_app\_id": [

"ip,tcp,ssl,ssl"

],

"fw\_app\_subcategory": null,

"fw\_app\_technology": null,

"initial\_cloud\_providers": null,

"causality\_actor\_process\_command\_line": [

"\"C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe\" --profile-directory=Default"

],

"causality\_actor\_process\_image\_md5": [

"a9b795c2e379f8926c41aa60fc2c2b1e"

],

"causality\_actor\_process\_image\_name": [

"msedge.exe"

],

"causality\_actor\_process\_image\_path": [

"C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe"

],

"causality\_actor\_process\_image\_sha256": [

"692d885a9fa2f45b5d98a594068cea5ca1099295600060121e05a8c2a8121339"

],

"causality\_actor\_process\_signature\_status": [

"Signed"

],

"causality\_actor\_process\_signature\_vendor": [

"Microsoft Corporation"

],

"causality\_actor\_causality\_id": [

"AduNGb7PaY0AABU4AAAAAA=="

],

"identity\_sub\_type": null,

"identity\_type": [

"unknown"

],

"operation\_name": null,

"project": null,

"cloud\_provider": null,

"referenced\_resource": null,

"resource\_sub\_type": null,

"resource\_type": null,

"cluster\_name": null,

"container\_id": null,

"contains\_featured\_host": [

"NO"

],

"contains\_featured\_ip": [

"NO"

],

"contains\_featured\_user": [

"NO"

],

"action\_country": [

"UNKNOWN"

],

"fw\_interface\_to": null,

"dns\_query\_name": null,

"agent\_device\_domain": null,

"fw\_email\_recipient": null,

"fw\_email\_sender": null,

"fw\_email\_subject": null,

"event\_type": [

"Network Connections"

],

"is\_whitelisted": false,

"action\_file\_macro\_sha256": null,

"action\_file\_md5": null,

"action\_file\_name": null,

"action\_file\_path": null,

"action\_file\_sha256": null,

"fw\_device\_name": null,

"fw\_rule\_id": null,

"fw\_rule": null,

"fw\_serial\_number": null,

"agent\_fqdn": null,

"agent\_os\_type": "Windows",

"image\_name": null,

"actor\_process\_image\_name": [

"msedge.exe"

],

"actor\_process\_command\_line": [

"\"C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe\" --type=utility --utility-sub-type=network.mojom.NetworkService --lang=en-US --service-sandbox-type=none --string-annotations --always-read-main-dll --field-trial-handle=2324,i,4083441262591821369,5405709462413476466,262144 --variations-seed-version --mojo-platform-channel-handle=2536 /prefetch:11"

],

"actor\_process\_image\_md5": [

"a9b795c2e379f8926c41aa60fc2c2b1e"

],

"actor\_process\_image\_path": [

"C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe"

],

"actor\_process\_os\_pid": [

6272

],

"actor\_process\_image\_sha256": [

"692d885a9fa2f45b5d98a594068cea5ca1099295600060121e05a8c2a8121339"

],

"actor\_process\_signature\_status": [

"Signed"

],

"actor\_process\_signature\_vendor": [

"Microsoft Corporation"

],

"actor\_thread\_thread\_id": [

6404

],

"fw\_is\_phishing": [

"N/A"

],

"action\_local\_ip": [

"172.30.1.133"

],

"action\_local\_port": [

50415

],

"fw\_misc": null,

"mitre\_tactic\_id\_and\_name": null,

"mitre\_technique\_id\_and\_name": null,

"module\_id": null,

"fw\_vsys": null,

"os\_actor\_process\_command\_line": [

"\"C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe\" --type=utility --utility-sub-type=network.mojom.NetworkService --lang=en-US --service-sandbox-type=none --string-annotations --always-read-main-dll --field-trial-handle=2324,i,4083441262591821369,5405709462413476466,262144 --variations-seed-version --mojo-platform-channel-handle=2536 /prefetch:11"

],

"os\_actor\_thread\_thread\_id": [

6404

],

"os\_actor\_process\_image\_name": [

"msedge.exe"

],

"os\_actor\_process\_os\_pid": [

6272

],

"os\_actor\_process\_image\_sha256": [

"692d885a9fa2f45b5d98a594068cea5ca1099295600060121e05a8c2a8121339"

],

"os\_actor\_process\_signature\_status": [

"Signed"

],

"os\_actor\_process\_signature\_vendor": [

"Microsoft Corporation"

],

"os\_actor\_effective\_username": null,

"action\_process\_signature\_status": [

"N/A"

],

"action\_process\_signature\_vendor": null,

"action\_registry\_data": null,

"action\_registry\_full\_key": null,

"action\_external\_hostname": [

"wicar.org"

],

"action\_remote\_ip": [

"199.34.228.69"

],

"action\_remote\_port": [

443

],

"matching\_service\_rule\_id": "2",

"fw\_interface\_from": null,

"starred": false,

"action\_process\_image\_command\_line": null,

"action\_process\_image\_name": null,

"action\_process\_image\_sha256": null,

"fw\_url\_domain": null,

"user\_agent": null,

"fw\_xff": null,

"alert\_domain": "DOMAIN\_SECURITY",

"external\_id": "be9df7b5-e192-4a3d-98a2-c2d0b4989ffb",

"severity": "high",

"matching\_status": "MATCHED",

"end\_match\_attempt\_ts": null,

"local\_insert\_ts": 1741104698580,

"last\_modified\_ts": 1742404061788,

"bioc\_indicator": null,

"attempt\_counter": 0,

"bioc\_category\_enum\_key": "DOMAIN\_NAME",

"case\_id": 1,

"deduplicate\_tokens": null,

"filter\_rule\_id": null,

"agent\_version": "8.7.0.7735",

"agent\_ip\_addresses\_v6": null,

"agent\_data\_collection\_status": null,

"agent\_is\_vdi": false,

"agent\_install\_type": "STANDARD",

"agent\_host\_boot\_time": [

1741097260399

],

"event\_sub\_type": [

4

],

"association\_strength": [

50

],

"dst\_association\_strength": [

10

],

"story\_id": [

"Njc3MTI5MDAxMDEzNzczNzI5Nw=="

],

"event\_id": [

"Njc3MTI5MDAxMDEzNzczNzI5Nw=="

],

"event\_timestamp": [

1741104253398

],

"actor\_process\_causality\_id": [

"AduNGb7PaY0AABU4AAAAAA=="

],

"actor\_causality\_id": [

"AduNGb7PaY0AABU4AAAAAA=="

],

"causality\_actor\_process\_execution\_time": [

1741101965392

],

"action\_registry\_key\_name": null,

"action\_registry\_value\_name": null,

"action\_local\_ip\_v6": null,

"action\_remote\_ip\_v6": null,

"action\_process\_causality\_id": null,

"os\_actor\_process\_instance\_id": [

"AduNGcatb/YAABiAAAAAAA=="

],

"os\_actor\_process\_image\_path": [

"C:\\Program Files (x86)\\Microsoft\\Edge\\Application\\msedge.exe"

],

"os\_actor\_process\_causality\_id": [

"AduNGb7PaY0AABU4AAAAAA=="

],

"os\_actor\_causality\_id": null,

"dst\_agent\_id": [

"199.34.228.69"

],

"dst\_causality\_actor\_process\_execution\_time": null,

"dst\_action\_external\_hostname": [

"wicar.org"

],

"dst\_action\_country": [

"US"

],

"dst\_action\_external\_port": null,

"is\_pcap": false,

"image\_id": null,

"container\_name": null,

"namespace": null,

"alert\_type": "Unclassified",

"resolution\_status": "STATUS\_020\_UNDER\_INVESTIGATION",

"resolution\_comment": null,

"dynamic\_fields": null,

"tags": [

"DS:PANW/XDR Agent",

"DOM:Security"

],

"malicious\_urls": null,

"alert\_id": "10",

"detection\_timestamp": 1741104253398,

"name": "IOC (wicar.org)",

"category": "Domain Name",

"endpoint\_id": "03544471722340dd99cb5ac21ec6e8b5",

"description": "IOC ( Domain Name = wicar.org )",

"host\_ip": [

"172.30.2.130",

"172.30.1.133"

],

"host\_name": "EC2AMAZ-CQMLOOA",

"action": "DETECTED",

"source": "XDR IOC",

"original\_tags": [

"DOM:Security",

"DS:PANW/XDR Agent"

],

"user\_name": [

"EC2AMAZ-CQMLOOA\\Administrator"

],

"mac\_addresses": null,

"action\_pretty": "Detected"

}

**Logs:**

S3 Client is set for bucket: 'gbcortex'

--- Attempting to clear alerts folder in s3://gbcortex/alerts/ ---

Found 11 objects to delete in s3://gbcortex/alerts/

Deleting batch 1 of 11 objects...

Batch delete successful. Deleted 11 objects.

✅ Finished clearing alerts folder.

--- S3 folder clearing process finished ---

--- Starting alert fetching and S3 upload process ---

Fetching page: 0...

Fetching alerts with payload: {"request\_data": {"use\_page\_token": true, "search\_to": 100, "sort": {"field": "creation\_time", "keyword": "desc"}}}

Fetched 11 alerts on page 0.

--- Starting S3 upload for 11 alerts on page 0 ---

Processing alert 1/11 on page 0...

⬆️ Attempting to upload alert 310 to s3://gbcortex/alerts/310.json

✅ Successfully uploaded alert 310 to S3.

Processing alert 2/11 on page 0...

⬆️ Attempting to upload alert 12 to s3://gbcortex/alerts/12.json

✅ Successfully uploaded alert 12 to S3.

Processing alert 3/11 on page 0...

⬆️ Attempting to upload alert 10 to s3://gbcortex/alerts/10.json

✅ Successfully uploaded alert 10 to S3.

Processing alert 4/11 on page 0...

⬆️ Attempting to upload alert 6 to s3://gbcortex/alerts/6.json

✅ Successfully uploaded alert 6 to S3.

Processing alert 5/11 on page 0...

⬆️ Attempting to upload alert 11 to s3://gbcortex/alerts/11.json

✅ Successfully uploaded alert 11 to S3.

Processing alert 6/11 on page 0...

⬆️ Attempting to upload alert 7 to s3://gbcortex/alerts/7.json

✅ Successfully uploaded alert 7 to S3.

Processing alert 7/11 on page 0...

⬆️ Attempting to upload alert 9 to s3://gbcortex/alerts/9.json

✅ Successfully uploaded alert 9 to S3.

Processing alert 8/11 on page 0...

⬆️ Attempting to upload alert 8 to s3://gbcortex/alerts/8.json

✅ Successfully uploaded alert 8 to S3.

Processing alert 9/11 on page 0...

⬆️ Attempting to upload alert 4 to s3://gbcortex/alerts/4.json

✅ Successfully uploaded alert 4 to S3.

Processing alert 10/11 on page 0...

⬆️ Attempting to upload alert 3 to s3://gbcortex/alerts/3.json

✅ Successfully uploaded alert 3 to S3.

Processing alert 11/11 on page 0...

⬆️ Attempting to upload alert 2 to s3://gbcortex/alerts/2.json

✅ Successfully uploaded alert 2 to S3.

--- Finished S3 upload for alerts on page 0 ---

Next page token for page 0: None

No more pages to fetch or no alerts on the last page.

--- Finished alert fetching and S3 upload. Total alerts uploaded: 11 ---

--- Verifying alerts in S3 bucket 'gbcortex' ---

✅ Found 11 alert objects in s3://gbcortex/alerts/

--- S3 verification complete ---