Official incident report

Event ID:120

Rule Name: SOC170 - Passwd Found in Requested URL - Possible LFI Attack

Made By

LinkedIn: Engineer.Ahmed Mansour

Link: https://www.linkedin.com/in/ahmed-mansour-5631b5323/

Github link: https://github.com/AhmedMansour93

Table of contents

Event Details

Event ID:
120

Event Date and Time:
Mar, 01, 2022, 10:10 AM

Rule:
SOC170 - Passwd Found in Requested URL - Possible LFI Attack

Level:
Security Analyst

Hostname:
WebServer1006

HTTP Request Method:
GET

Requested URL:
https://172.16.17.13/?file=../../../../etc/passwd

User-Agent:
Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)

Alert Trigger Reason:
URL Contains passwd

Device Action:
Allowed

Network Information Details

Destination Address:
172.16.17.13 internal

Source Address:
106.55.45.162 external

External / Internal Attack:

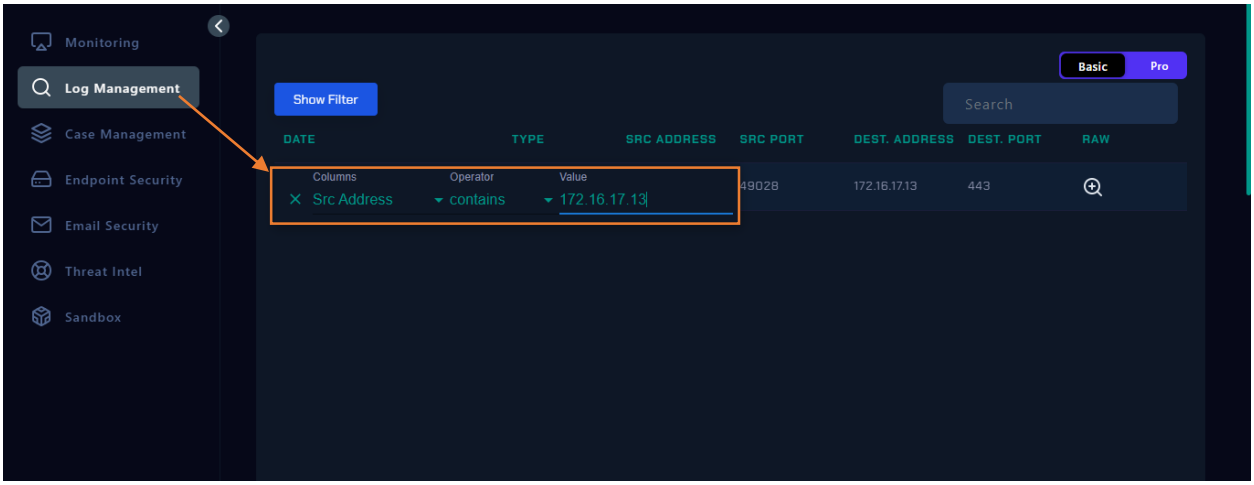
Based on the event details, the attack appears to be **external**.

Analysis:

Log Management

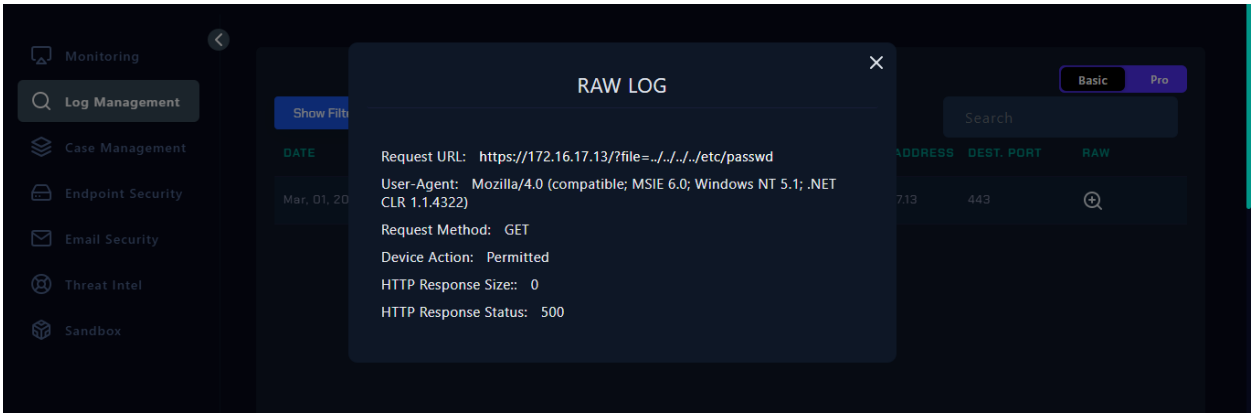
We'll proceed by entering the source IP address and reviewing the results.

Please refer to the attached image for further details regarding the attack.



Only 1 Log record for the source IP with the same date of the attack.

Please refer to the attached image for further details regarding the attack.



Log Breakdown

1. **Request URL**:

https://172.16.17.13/?file=../../../../etc/passwd

- **Description**: The attacker is attempting to perform a Local File Inclusion (LFI) attack by trying to access the file `/etc/passwd` on the web server. This file typically contains user account information on Unix/Linux systems.

2. **User-Agent**:

Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)

- **Description**: This indicates the web client making the request. The User-Agent string is meant to mimic an older version of Internet Explorer on a Windows XP system, but this can be easily spoofed.

3. **Request Method**:

GET

- **Description**: This is a standard HTTP method used to request data from the server. In this case, the GET method is used to try to retrieve the contents of the file.

4. **Device Action**:

Permitted

- **Description**: The device (likely a web server or firewall) allowed the request to proceed. This suggests that there was no blocking rule or mechanism in place to prevent this specific request.

5. **HTTP Response Size**:

0

- **Description**: This indicates that the server did not return any content in the response. This could be due to an error or because the requested file does not exist or could not be read.

6. **HTTP Response Status**:

500

- **Description**: This is an HTTP status code indicating a "500 Internal Server Error." This status code is returned when the server encounters an unexpected condition that prevents it from fulfilling the request. In this context, it means that the server encountered an error when trying to process the request.

Summary of the Incident

1. **Attack Attempt**:
   - The attacker made a GET request to the web server at `172.16.17.13` trying to access the file `/etc/passwd` using a URL parameter. This suggests an attempt to exploit a Local File Inclusion (LFI) vulnerability in the web application.
2. **Server Response**:
   - The server attempted to process the request but encountered an error, resulting in an HTTP 500 status code. This could mean that either the file `/etc/passwd` does not exist or the server's attempt to include or read the file resulted in a server-side error.
3. **Security Implications**:
   - Although the request was permitted and processed (the device action was allowed), the actual result was an internal server error, which may indicate that the server has some form of protection or configuration issue that prevented the inclusion of the file.
4. **Next Steps**:
   - **Investigate the Error**: Review server logs to understand why the 500 error occurred and if it's related to the LFI attempt.
   - **Check Security Configuration**: Ensure that proper security measures are in place to prevent LFI attacks, such as input validation and restrictions on file paths.
   - **Monitor for Further Attempts**: Keep an eye on similar requests to identify if this is part of a larger attack pattern.

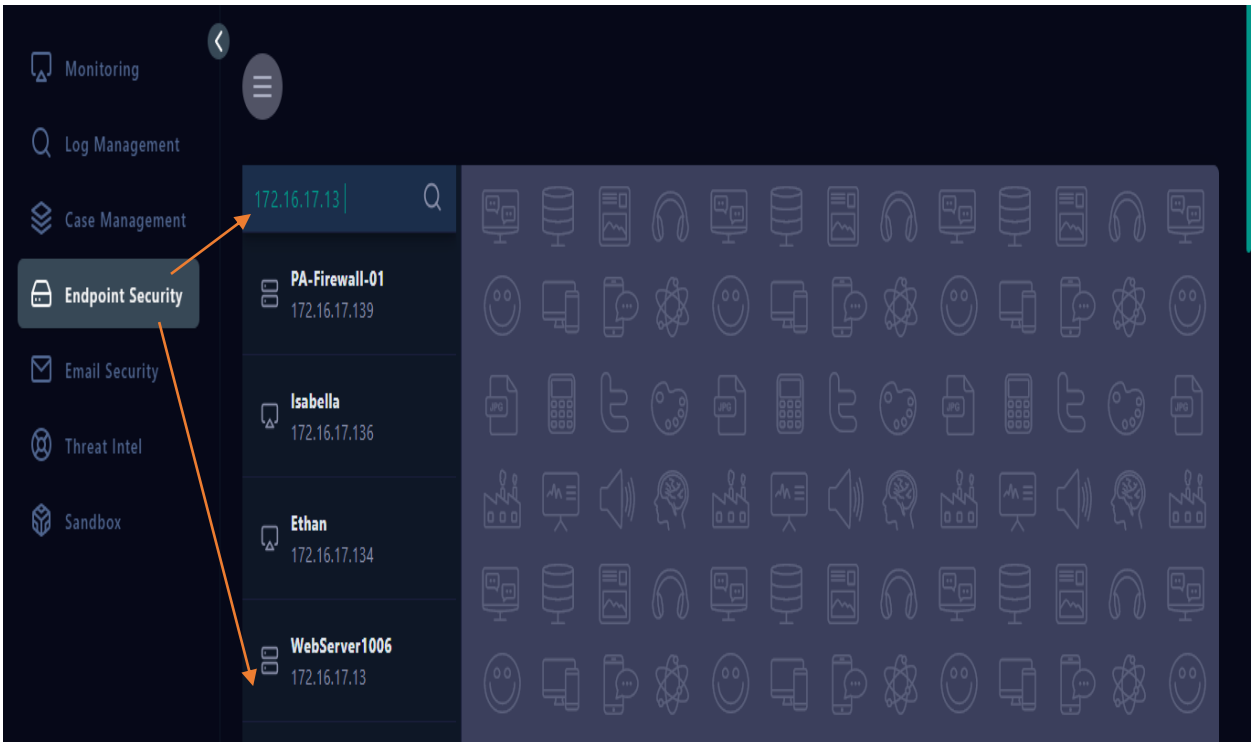
In summary, while the request was allowed by the server, the attempt to exploit a vulnerability resulted in an error. It's crucial to investigate further to ensure that the server is protected against such attacks and to understand any potential security gaps.

The attack was not successful in retrieving the `/etc/passwd` file or any other content.

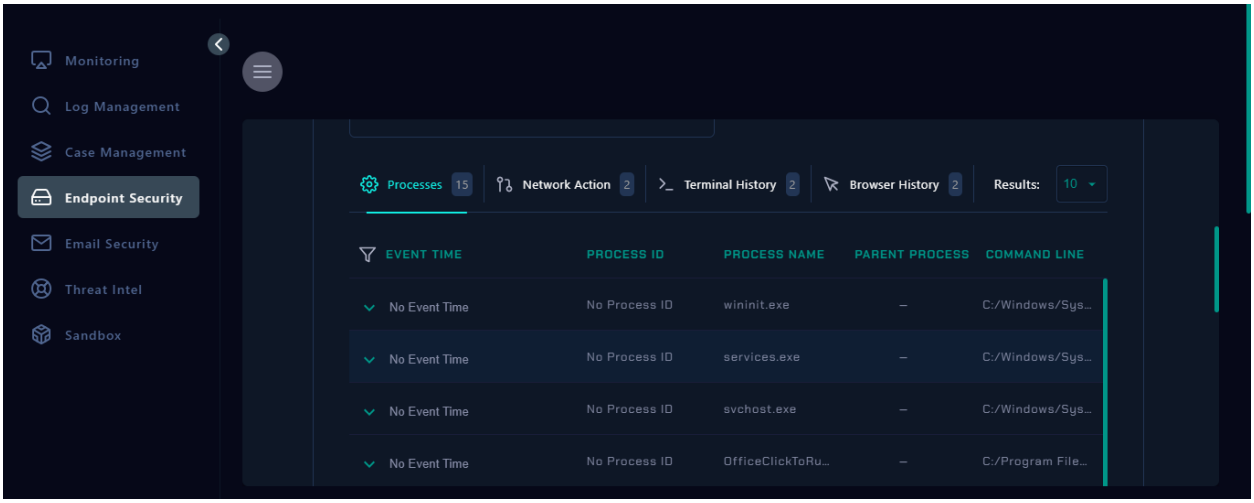
Endpoint Security

We will enter either the source IP to check the activity that happened on the server.

Refer to the attached image for further details.

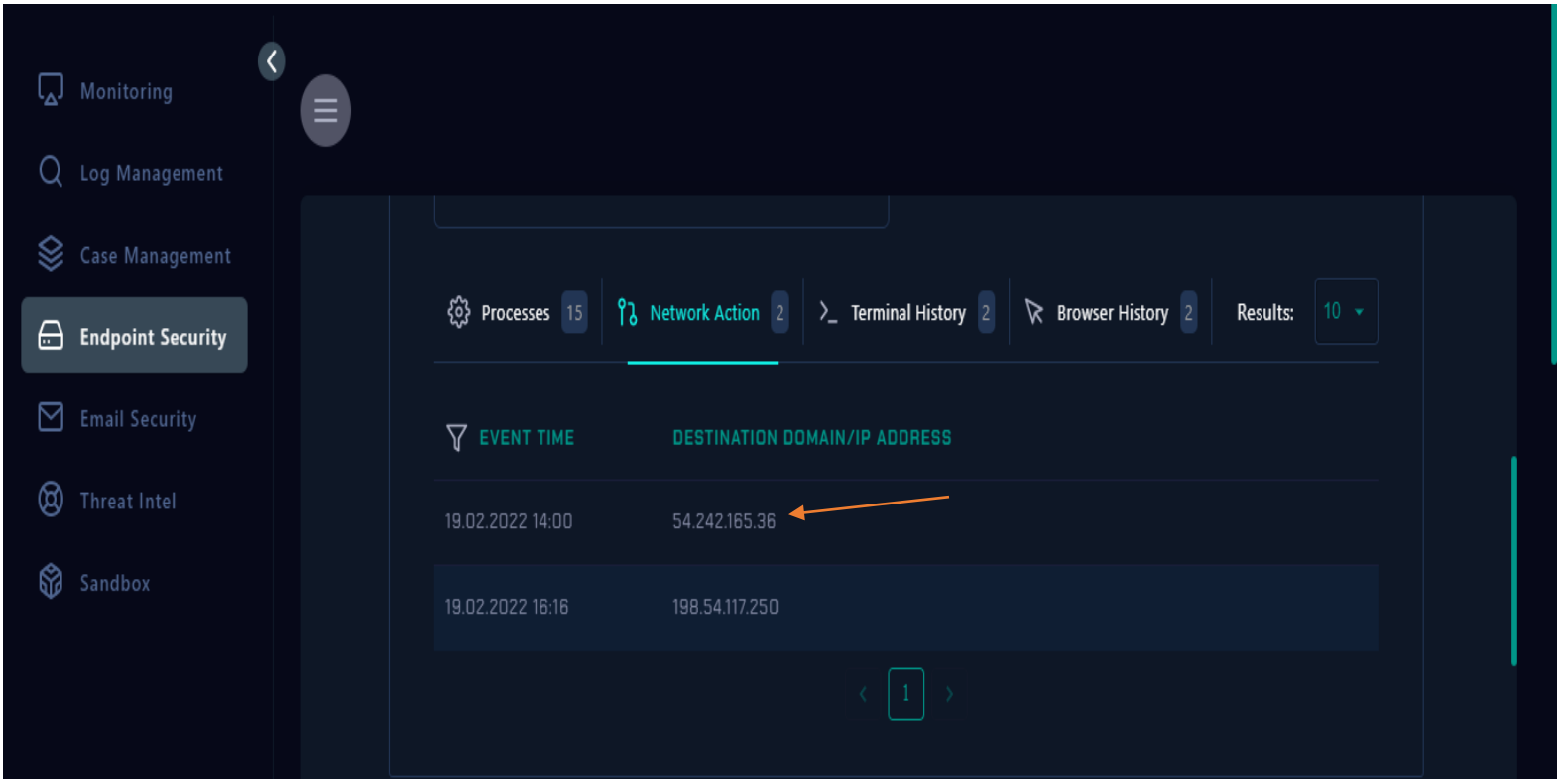


Incident Overview: During our routine endpoint security checks, no significant issues were found on the alert date. However, network activity logs revealed connections between our server and two potentially malicious IP addresses. Our firewall successfully detected and addressed these connections.

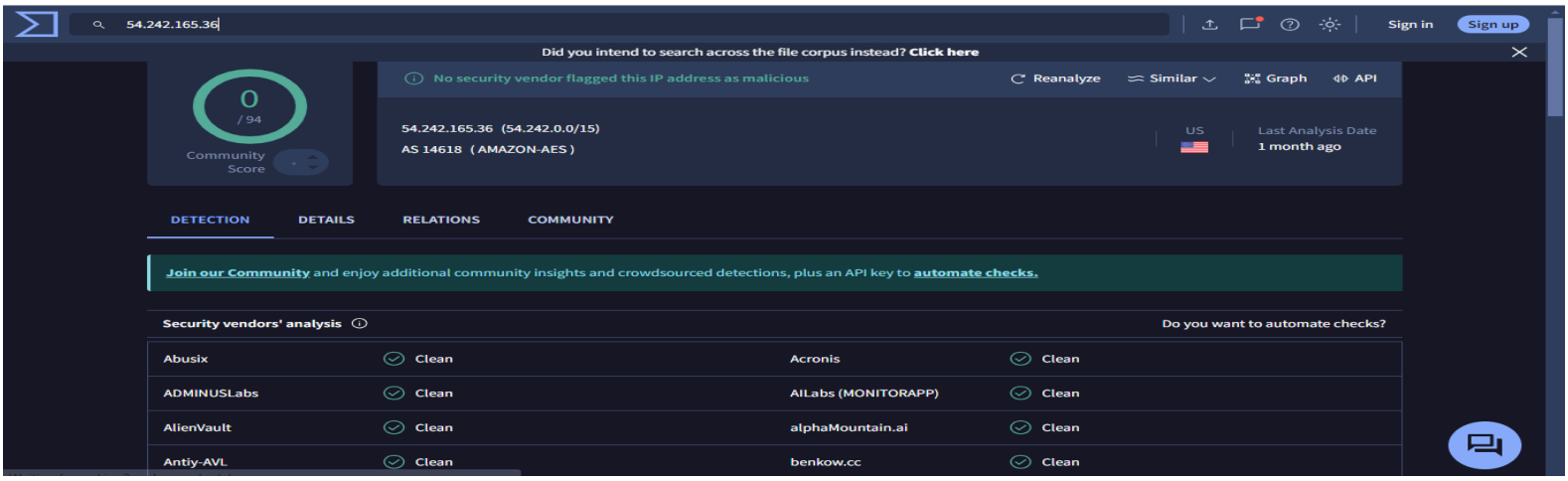


Detailed Analysis of Network Actions section:

1. **Record 1:**

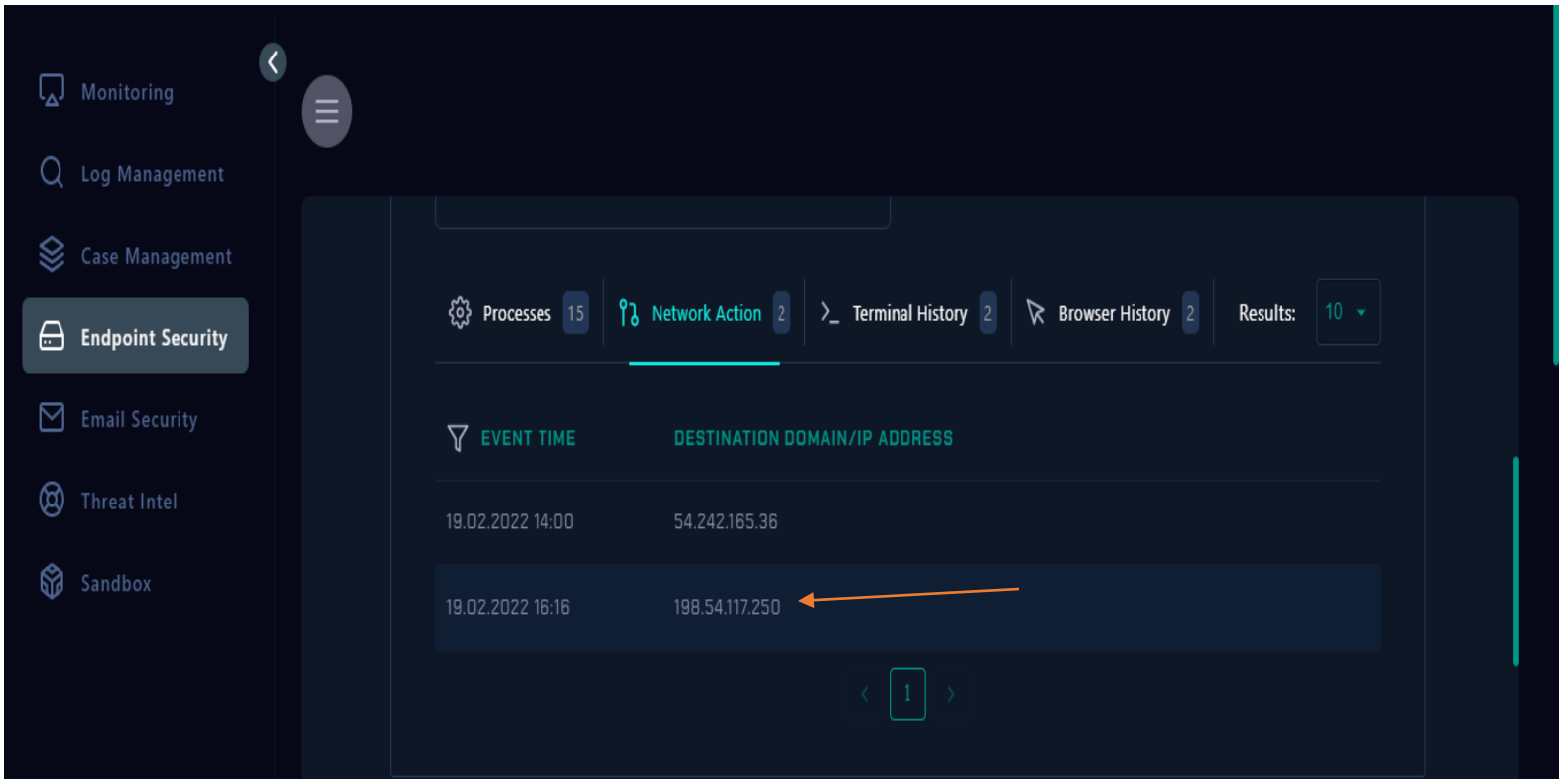


- **Date and Time**: February 19, 2022, 14:00
- **IP Address**: 54.242.165.36
- **Initial Assessment**: No issues detected
- **Additional Checks**: Refer to the attached photo for details and analysis.

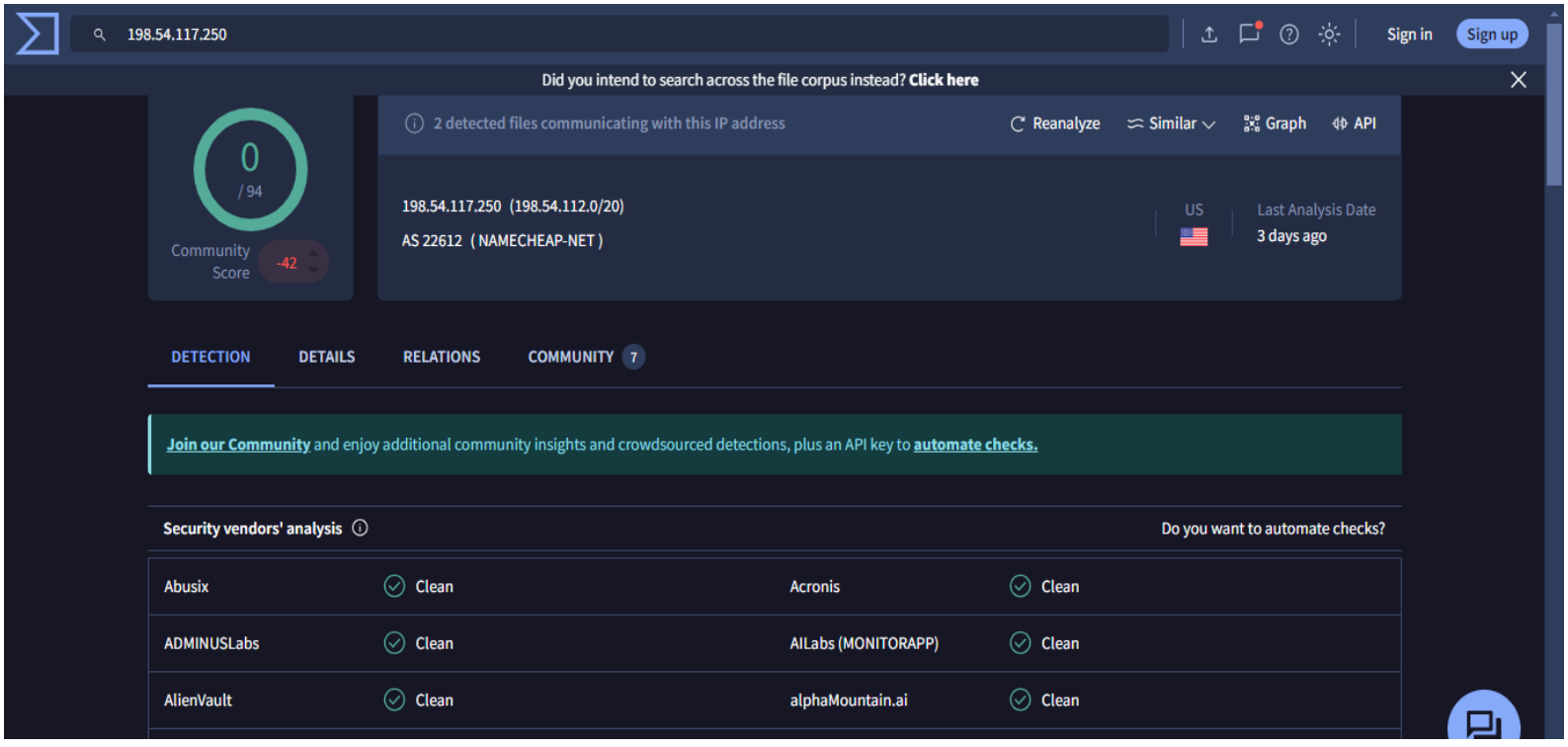


- **VirusTotal Analysis**: The results from VirusTotal for this IP address indicated no significant threats. For more details, please refer to the provided link.

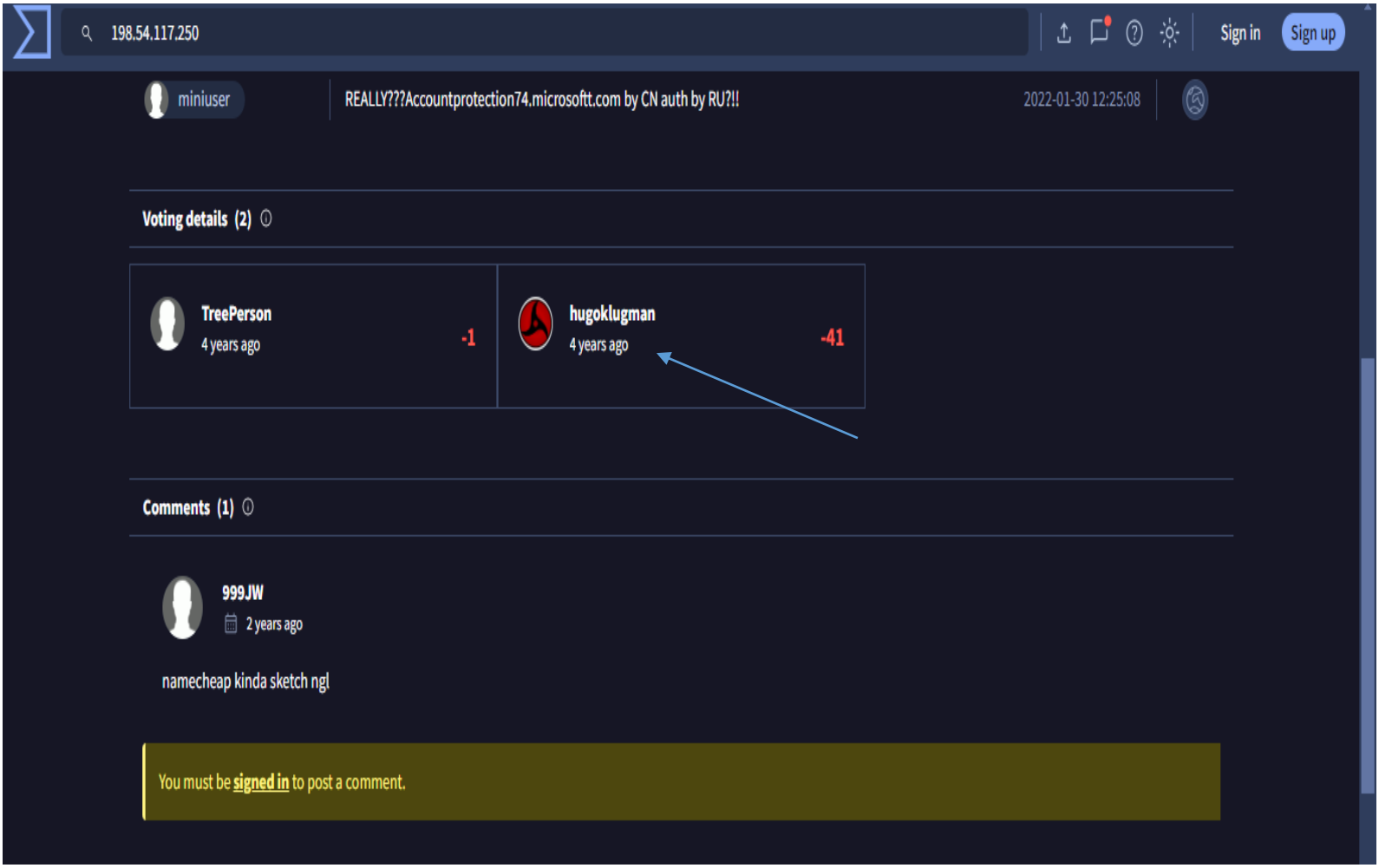
2. **Record 2:**



- **Date and Time**: February 19, 2022, 16:16
- **IP Address**: 198.54.117.250
-
- **Initial Assessment**: Analysis appears clear in the detection section.

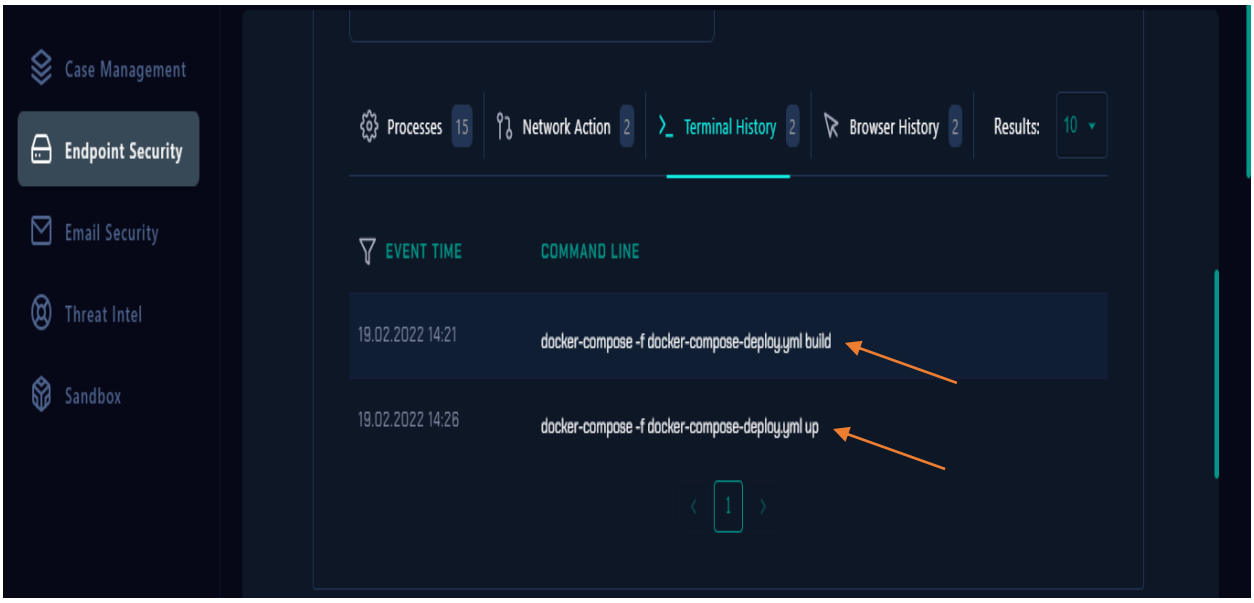


- **Additional Comments**: Notably, there are over 40 comments in the VirusTotal community section regarding this IP address. Please refer to the attached photo for detailed analysis and the provided link for community feedback.



Review of Terminal History Records

Please review the attached photos of the terminal history records. We need to conduct a step-by-step analysis of each record.



Terminal History Record 1:

- **Description:**

Record 1: `docker-compose -f docker-compose-deploy.yml build`

1. **Date and Time**: 19.02.2022 14:21
    - This indicates when the command was executed: February 19, 2022, at 14:21 (2:21 PM).
2. **Command**: `docker-compose -f docker-compose-deploy.yml build`
    - **`docker-compose`**: This is the command-line tool for Docker Compose.
    - **`-f docker-compose-deploy.yml`**: The `-f` flag specifies the Docker Compose file to use. Here, `docker-compose-deploy.yml` is the file that contains the configuration for the Docker services. This YAML file defines how Docker containers should be built and run.
    - **`build`**: This command tells Docker Compose to build the Docker images specified in the `docker-compose-deploy.yml` file.

    Purpose: The `build` command compiles the Docker images from the Dockerfiles specified in the YAML file. If there are changes to the application code or the Dockerfile, running this command will ensure that the latest version of the images is created.

Terminal History Record 2:

- **Description:**

Record 2: `docker-compose -f docker-compose-deploy.yml up`

1. **Date and Time**: 19.02.2022 14:26
   - This indicates when the command was executed: February 19, 2022, at 14:26 (2:26 PM).
2. **Command**: `docker-compose -f docker-compose-deploy.yml up`
   - **`docker-compose`**: Again, this is the Docker Compose command-line tool.
   - **`-f docker-compose-deploy.yml`**: The `-f` flag specifies the Docker Compose file to use, just like in the previous command.
   - **`up`**: This command starts the services defined in the `docker-compose-deploy.yml` file. If the images are not built, it will automatically build them first before starting the containers.

   Purpose: The `up` command creates and starts containers, networks, and volumes defined in the YAML file. It ensures that all services defined in the configuration file are running. If the images were built in the previous step, this command will start the application using those images.

Summary

- The **first command** (`docker-compose -f docker-compose-deploy.yml build`) is used to build the Docker images from the Dockerfile(s) defined in the `docker-compose-deploy.yml` file. It prepares the images for running.
- The **second command** (`docker-compose -f docker-compose-deploy.yml up`) is used to start the services defined in the `docker-compose-deploy.yml` file, using the built images. It makes the application up and running.

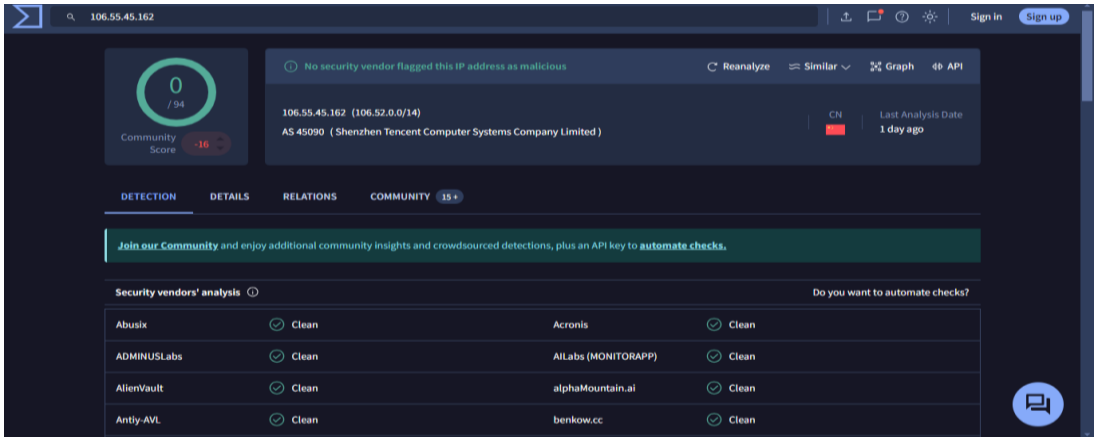
Detection:

Threat Intelligence Results

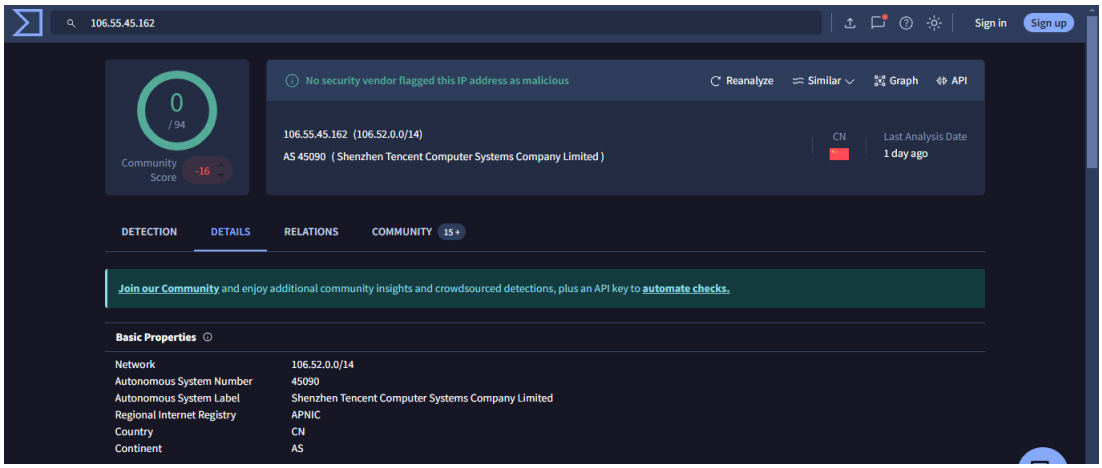
Source IP Analysis on VirusTotal

Objective: Evaluate the source IP address using VirusTotal and review the findings in various sections.

1. **Detection Section**
    - **Status:** The Detection section shows no detections. This indicates that none of the security vendors have flagged the IP address as malicious at this time.

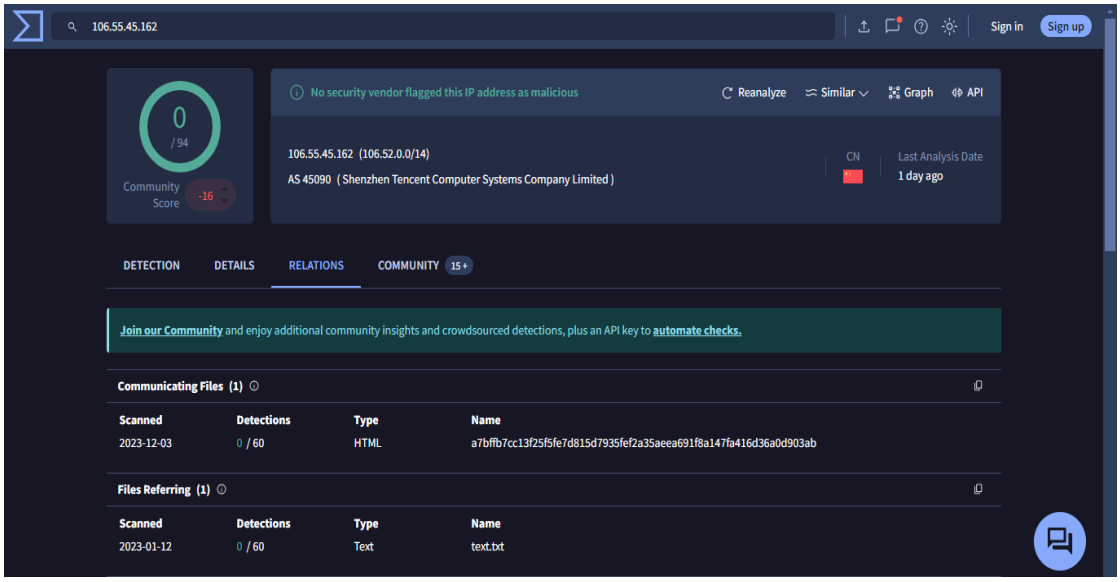


    - **Reference:** View Detection Section
2. **Details Section**
    - **Status:** The Details section is clear, with no additional issues or anomalies noted. This section provides standard information about the IP address.

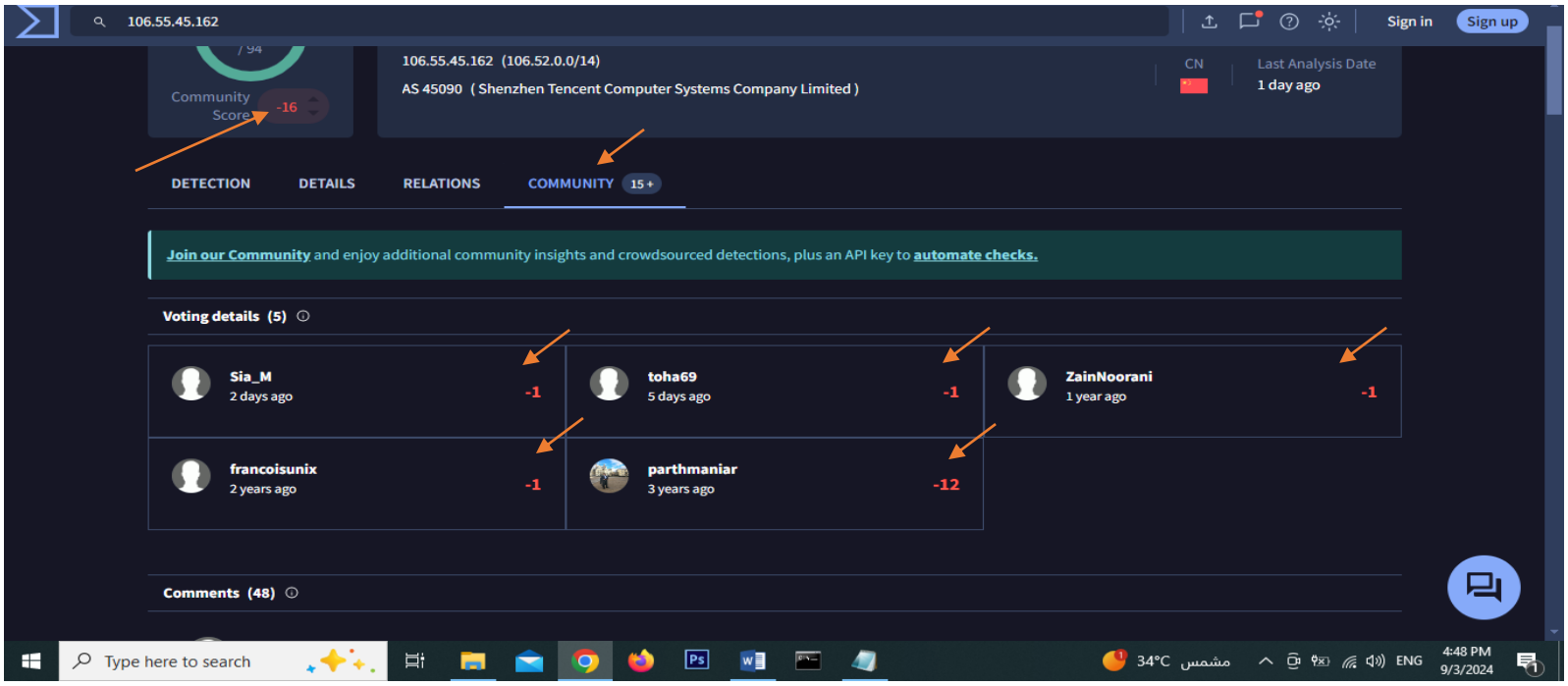


    - **Reference:** View Details Section

3. **Relation Section**
    - **Status:** The Relation section is clear, indicating no significant associations with other known malicious entities or infrastructure.



    - **Reference:** View Relations Section
4. **Community Section**
    - **Status:** The Community section shows substantial activity, with over 15 comments concerning this IP address. This indicates a significant level of interest or concern from the security community.



    - **Reference:** View Community Section

Conclusion

In conclusion, our investigation into the incident revealed several critical insights that provide a comprehensive understanding of the attack and its context.

Incident Summary: On March 1, 2022, at 10:10 AM, a Local File Inclusion (LFI) attack was attempted against the web server `WebServer1006` with the URL `https://172.16.17.13/?file=../../../../etc/passwd`. The HTTP request aimed to access the `/etc/passwd` file, a critical system file that can reveal sensitive user information. Despite the attempt, the server responded with a `500 Internal Server Error`, suggesting that either the file was inaccessible or an error occurred during processing.

Key Findings:

1. **Security Analysis:** The server's allowance of the request indicates a potential gap in security configurations that may have facilitated this attack attempt. However, the lack of successful file retrieval and the server error provide some level of protection against the immediate exploitation.
2. **Network Activity:** Log records and endpoint security checks showed no significant issues or successful threats from the external IPs involved. VirusTotal analysis of these IPs returned minimal concerns, though community feedback warrants further attention due to substantial activity.
3. **Infrastructure and Commands:** Analysis of Docker commands revealed routine maintenance activities, which appear unrelated to the incident. The commands executed were standard for building and deploying Docker containers, suggesting no immediate link to the attack.

Action Items:

1. **Investigate Server Configuration:** Further review of the server's configuration and logs is necessary to understand why the internal server error occurred and to ensure that proper defenses against LFI attacks are in place.
2. **Enhance Security Measures:** Implement robust input validation, enforce strict file path restrictions, and review existing security policies to mitigate future vulnerabilities.
3. **Monitor Network Traffic:** Continue to monitor network traffic and logs for any signs of recurring or related malicious activity. Engage with the security community to keep abreast of potential threats and gather additional context from the community comments on VirusTotal.

Final Assessment: While the immediate attack attempt did not succeed in compromising sensitive data, the incident highlights the need for ongoing vigilance and improvement in our security posture. By addressing the identified gaps and enhancing our protective measures, we can fortify our defenses against future threats and safeguard our critical assets more effectively.