

Attackers exploiting the PHP flaws hand the break-in to AI agents, Unit 42 says

Source copy kept by wheresthe.ai for the story "Attackers exploiting the PHP flaws hand the break-in to AI agents, Unit 42 says" (<https://wheresthe.ai/d/intl-attackers-use-ai-agents-after-php2shell-exploits-2026-10/>).

Source: <https://unit42.paloaltonetworks.com/cve-2026-109618-cve-2026-109619/>

Text saved 25 Oct 2026 (the page as published).

SHA-256 of the saved text: 9790032a4a59500a97c80860441c9a13caa539435b779701a3254b3837060eab

Threat Research Center High Profile Threats Vulnerabilities

Vulnerabilities Threat Brief: Critical Vulnerabilities in PHP: CVE-2026-109618, CVE-2026-109619 (Updated October 23)

7 min read

Related Products Advanced Threat Prevention Cloud-Delivered Security Services Cortex Cortex Cloud Cortex XDR Cortex Xpanse Cortex XSIAM Next-Generation Firewall Prisma Cloud Prisma Cloud Compute Unit 42 Deep and Dark Web Service Unit 42 Incident Response Unit 42 Threat Intelligence Services

By: Unit 42

Published: October 23, 2026

Categories: High Profile Threats Vulnerabilities

Tags: CVE-2026-109618 CVE-2026-109619 PHP2Shell

Share

Executive Summary

Unit 42 is aware of two critical vulnerabilities in PHP: CVE-2026-109618 and CVE-2026-109619. Both are remote code execution (RCE) vulnerabilities rated as 9.2 (CVSS 4.0). The vulnerabilities are being referred to as PHP2Shell. We encourage reviewing the PHP Group's security advisory and patching guidance .

PHP is widely used across web servers and content management systems globally. We recommend assessing your exposure to these vulnerabilities and patching immediately.

Unit 42 has seen proof of concept of these vulnerabilities. At least one European government has taken its website offline as a precaution.

Palo Alto Networks has evaluated our product portfolio regarding this issue. Refer to our security advisory for the most up to date details.

CVE-2026-109618 involves a remote code execution vulnerability where handling out-of-memory errors via `longjmp()` leaves request memory uninitialized due to improper control flow management, allowing attackers to reliably achieve RCE using out-of-memory (OOM) triggers.

CVE-2026-109619 describes a remote code execution vulnerability where `longjmp()` fatal error handling leaves global memory pointing to freed memory regions. These regions can later be read or branched on during subsequent requests, enabling attackers to execute arbitrary code.

Unit 42 researchers also observed attackers using agentic AI attack platforms for post-exploitation activity. In one case, the attackers targeted a government entity in Europe and within two hours of the initial exploitation they had successfully accessed password hashes and multi-factor authentication secrets. They also established an NT LAN Manager (NTLM)-relay tunnel into Active Directory and attempted Domain Admin privilege escalation.

We observed an attacker enumerating the compromised endpoint. The commands and behavioral markers used in the attack are strong indicators of an agentic attacker. Each command batch is wrapped in section-header delimiters, such as `==SHELLUSER==` and `==PASSWD==` . These delimiters produce machine-parseable output for an AI model to read. We also noted the use of the short, serially spaced tasks and quick pivots after failures.

Figure 1 shows the commands that were executed, as seen in Cortex XDR.

Figure 1. Recon commands executed via web shell by an agentic attacker. At the time of publication, Palo Alto Networks Cortex Xpanse has identified the presence of over 22.6 million exposed instances potentially vulnerable to CVE-2026-109618 and CVE-2026-109619 based on our telemetry.

Palo Alto Networks customers receive protections from and mitigations for CVE-2026-109618 and CVE-2026-109619 in the following ways:

Cortex XDR and XSIAM

Cortex Cloud and Prisma Cloud

Cortex Xpanse

Advanced Threat Prevention Plus with Frontier Virtual Patching

Advanced Threat Prevention

Palo Alto Networks also recommends assessing your exposure to these vulnerabilities and patching immediately.

Organizations can access tailored adversary intelligence, direct analyst consultations, and specialized exposure tracking through Unit 42 Threat Intel Services and our Deep and Dark Web Service .

The Unit 42 Incident Response team can also be engaged to help with a compromise or to provide a proactive assessment to lower your risk.

Vulnerabilities Discussed CVE-2026-109618 , CVE-2026-109619 , PHP2Shell

Unit 42 Managed Threat Hunting Queries

The Unit 42 Managed Threat Hunting team continues to track any attempts to exploit these CVEs across our customers, using Cortex XDR and the XQL queries below. Cortex XDR customers can also use these XQL queries to search for signs of exploitation.

// Description: This query attempts to identify post-exploitation command execution within the context of a PHP web server or worker process.

// Notes: This hunting query is not specific to CVE-2026-109618 and CVE-2026-109619. It is capable of identifying exploitation of various web application vulnerabilities. This query may also produce false positive findings depending on the specific web applications running. We recommend validating if any identified activity is expected for the web applications running on systems.

```
config case_sensitive = false | dataset = xdr_data | filter event_type = ENUM.PROCESS and event_sub_type = ENUM.PROCESS_START and action_process_image_name != null | fields _time, agent_hostname, agent_id, event_id, actor_effective_username, actor_process_image_name, actor_process_command_line, action_process_image_name, action_process_image_path, action_process_image_command_line, causality_actor_process_image_name, causality_actor_process_command_line
```

```
// Lowercase fields once | alter actor_effective_username = lowercase(actor_effective_username), actor_process_image_name = lowercase(actor_process_image_name), actor_process_command_line = lowercase(actor_process_command_line), action_process_image_name = lowercase(action_process_image_name), action_process_image_path = lowercase(action_process_image_path), action_process_image_command_line = lowercase(action_process_image_command_line), causality_actor_process_image_name = lowercase(causality_actor_process_image_name)
```

```
// Anchor: PHP web worker (or the launcher that started it) as causality, PHP worker as direct parent, // a shell directly under an IIS worker, or the web service account outside an interactive login session | filter causality_actor_process_image_name ~="^(?:apache2|httpd|apachectl|php-fpm[0-9.]*|php-cgi[0-9.]*|php-fcgi|sw-engine(?:-fpm)?|frankenphp|lsphp[0-9]*|lshttpd|litespeed|xampp-control|wampmanager|laragon|de\appsolute\mampprohelper)(?:\.exe)?$" or actor_process_image_name ~="^(?:apache2|httpd|php-fpm[0-9.]*|php-cgi[0-9.]*|php-fcgi|sw-engine(?:-fpm)?|frankenphp|lsphp[0-9]*|lshttpd|litespeed)(?:\.exe)?$" or (causality_actor_process_image_name = "w3wp.exe" and actor_process_image_name in ("cmd.exe", "powershell.exe", "pwsh.exe")) or (actor_effective_username ~="(?:^[\^a-z0-9_-])(?:www-data|apache|nginx|
```

```

wwwrun|iusr)$" and coalesce(causality_actor_process_image_name, "") !~="^(?:sshd(?:-session)?|screen|tmux|
login|mosh-server|dropbear)$") or actor_effective_username contains "iis apppool"

// Known-benign workloads (monitoring plugins, hosting control panels) and commands launched directly by CDK
Drive DMS scripts | filter action_process_image_command_line !~="(?:\Vusr\lib(?:64)?\Vnagios\plugins\|\Vusr\
local\cpanel\|\Vusr\local\psa\|\Vopt\psa\|\Vusr\local\directadmin\|\Vusr\local\vispconfig\|\Vusr\local\virtualizor\|
richdocumentscode|phpsysinfo|librenms|zabbix)" and coalesce(actor_process_command_line, "") !~="^(?:\S+
\s+)?(?:\Vadp\(\?:bin|3party|etc)\V\|local\bin\|uminfo)" // Phusion Passenger app spawns run id/uname/env from
their spawn helpers | filter coalesce(action_process_image_command_line, "") !~="^(?:\Vusr\bin\|)?(?:id(?:\s+[-a-
z]+)?|uname(?:\s+[-a-z]+)?|hostname|env)$|passenger-config\s+system-metrics" or
lowercase(coalesce(actor_process_command_line, "")) !~="passengeragent\s+spawn-|spawn(?:preparer|
envsetter|shell)\V\helper-scripts\rack-preloader\rb|passenger_ruby_wrapper|phusion_passenger" // RHEL-
family /etc/profile runs these in every bash login shell | filter coalesce(action_process_image_command_line, "") !
~="^\Vusr\bin\(\?:id\s+[-ugn]|hostname)$" or lowercase(coalesce(actor_process_command_line, "")) !~="^(?:
\S*)?bash\s+[-a-z]*|^-bash$"

// parsed_lookup: the tool is the first command of its parent "sh -c" and its output is piped into a text filter (scripted
lookup) | alter tool = replace(action_process_image_name, ".exe", ""), shell_first_cmd =
arrayindex(regextract(coalesce(actor_process_command_line, ""), "^\s*(?:\S*)?(?:ba|da|k)?sh\s+-c\s+(\S+)"), 0) |
alter parsed_lookup = if(arrayindex(regextract(coalesce(shell_first_cmd, ""), "([^\V]+)$"), 0) = tool and
actor_process_command_line ~="^\s*(?:\S*)?(?:ba|da|k)?sh\s+-c\s+[^;&|]+\s*(?:e?grep|awk|cut|sed|wc|head|
tail|sort|tr)\b", 1, 0)

// Signals - one event can carry several | alter signals = concat( // SCORED - host/user/network/container recon,
excluding scripted lookups (distinct tools are counted later) if(parsed_lookup = 0 and (tool in ("whoami",
"hostname", "hostnamectl", "ifconfig", "netstat", "ss", "arp", "route", "w", "who", "last", "lastlog", "getent",
"lsb_release", "lscpu", "printenv", "env", "getcap", "capsh", "systeminfo", "ipconfig", "nltst", "tasklist", "quser",
"qwinsta") or (tool = "uname" and action_process_image_command_line ~="\buname\s+(\?:[-a-z]*[arv]|--all|--
kernel-)" or (tool = "id" and action_process_image_command_line !~="\bid\s+[-ugnrz]+b") or (tool = "ip" and
action_process_image_command_line ~="\bip\s+(\?:[-a-z0-9]+s+)*(?:a|addr|address|r|route|n|neigh|link)\b") or
(tool = "ps" and action_process_image_command_line ~="\bps\s+(\?:-?aux|-ef|-e\b|axjf|-auxf)") or (tool in ("net",
"net1") and action_process_image_command_line ~="\bnet1?(?:\.exe)?\s+(\?:user|group|localgroup|accounts|
view|share)\b") or (tool = "cat" and action_process_image_command_line ~="\Vetc\(\?:passwd|shadow|group|
issue|os-release|hosts|hostname|resolv\.\conf|crontab|sudoers)\V\proc\(\?:version|net\(\?:self[[0-9]+]\V\(\?:cgroup|
mountinfo)\)\V\.\dockerenv") or (tool = "find" and action_process_image_command_line ~="-perm\s+(\?:-4000|-u=s|
V4000|-2000)" or (tool = "sudo" and action_process_image_command_line ~="\bsudo\s+-lb")), "recon", "")),

// SCORED - payload transfer to a file (curl with an output file, wget not to stdout), excluding /dev/null and
localhost if(((tool = "curl" and action_process_image_command_line ~="\s(\?:[-a-z]*o\b|--output\b|--remote-name)|
>\s*\S)") or (tool = "wget" and action_process_image_command_line !~="\s[-a-z]*o\s*(?:\s|$)|--output-
document=|--spider") or (tool in ("tftp", "fetch") and action_process_image_command_line ~="(\?:https?|ftp|tftp):\V
\|s[-a-z]*o(?:\s|-$)|") or (tool = "certutil" and action_process_image_command_line ~="urlcache|verifyctl") or (tool
= "bitsadmin" and action_process_image_command_line contains "/transfer") or (tool in ("powershell", "pwsh")
and action_process_image_command_line ~="downloadstring|downloadfile|invoke-webrequest|biwr\b|
net.webclient|start-bitstransfer")) and action_process_image_command_line !~="-o\s*\Vdev\|null|--output-
document=\Vdev\|null|:\V\(\?:localhost|127\.\0\.\0\.\1|[:1]\)?[:\V]", "download", "")),

// SCORED - chmod making a file in a tmp-class directory executable if(tool = "chmod" and
action_process_image_command_line ~="\bchmod\s+(\?:[-a-z]+\s+)*(\?:[0-7](?:\{[1357][0-7]{2}[0-7][1357][0-7]|
[0-7]{2}[1357])|)[ugoa,+=rwt]*[+=][rwt]*x[ugoa,+=rwtst-]*\s" and action_process_image_command_line ~="\s\V
(?:tmp|var/tmp|dev/shm|run/shm)\V", "staging", "")),

// STANDALONE - binary executed from a writable/memory-backed dir, or from a deleted/memfd image
if(action_process_image_path ~="^\V(\?:tmp|var/tmp|dev/shm|run/shm|run/user)\V|^/proc\(\?:self[[0-9]+]\Vfd\|^V
memfd:\(deleted\)\s*$", "writable_exec", "")),

// STANDALONE - reverse/bind shell constructs if(action_process_image_command_line ~="\Vdev\(\?:tcp|udp)\V|
(?:\s|&|(\V))(\?:nc|ncat|netcat)(?:\s+[-a-z]+)?\s+(\?:\S+\s+){0,6}(?:\s+[-a-z]{0,3}[ec]|--exec|--sh-exec)\s+\S*(?:sh|bash|
cmd|python[0-9.]*|perl)\b|\bsocat\b\^&|]{0,80}exec:\bmkfifo\b|\b(\?:ba|da|z|a)?sh\s+-i\b|socket\.\socket\(|

```

```
// fsockopen("\\bpty\spawn(\" , \"revshell\", \"\"),  
// STANDALONE - remote content piped straight into an interpreter if(action_process_image_command_line =~  
"(?:curl|wget|fetch)\b[^\]]*\s*(?:\\$S*)?(?:(?:ba|da|z)a)?sh|python[23]?|perl|php)\b", "pipe_to_shell", ""),  
// SCORED - encoded/obfuscated execution if(action_process_image_command_line =~ "base64\s+(?:-d|--  
decode)|frombase64string|-e(?:nc|ncodedcommand)?\s+[a-z0-9+V]{40,}|echo\s+[a-z0-9+V]{40,}|  
\bopenssl\s+(?:enc\b[^\]]*\s-d\b|base64\s+-d\b)|\bxxd\s+(?:-[a-z]+\s+)*-[a-z]*r", "encoded", ""),  
// SCORED - persistence / backgrounding (crontab edits, authorized_keys writes, new users, nohup/setuid of tmp-  
class binaries or shells, // scheduled tasks, Run keys, systemd units, chattr +i) if((tool = "crontab" and  
action_process_image_command_line !~= "\bcrontab\s+-$") or (action_process_image_command_line =~  
"(?:>>?\s*\btee\s+(?:-a\s+)?|\b(?:cp|mv|install|ln)\s+(?:-\S+\s+)*\S+\s+)\S*authorized_keys" and  
action_process_image_command_line !~= "\bcp\s+\S*\.\ssh/id_[a-z0-9]+\.pub\s+\S*\.\ssh/authorized_keys") or  
action_process_image_command_line =~ "\b(?:useradd|adduser)\b|\b(?:nohup|setuid)\s+(?:-\S+\s+)*(?:\.\V|/  
(?:tmp|var/tmp|dev/vshm/run/vshm)/|(?:ba|da|z)?sh\s+[-ci]\b)|schtasks\s+\Vcreate|currentversion.run\b|  
\bsystemctl\s+(?:--user\s+)?enable\b|\.config/systemd/user/V/etc/systemd/system/V\bchattr\s+|[a-z]*i",  
"persistence", "")),  
// SCORED - inline interpreter one-liners, excluding PHP version/feature probes and Ruby gem-path probes  
if(action_process_image_command_line =~ "(?:^[^a-z0-9._-])(?:python[23]?(?:\.[0-9]+)?\s+c|perl\s+-e|ruby\s+-e|  
node(?:js)?\s+-e|lua\s+-e|php[0-9.]*\s+-r)\b" and action_process_image_command_line !~= "phpversion\(|  
php_version|date('t')|function_exists\(|(extension_loaded\(|(gem\.`path", "interpreter", "")),  
// SCORED - cloud/container credential and secret access (credential files, env-var names, instance metadata  
service, .env reads) if(action_process_image_command_line =~ "\.awsV(?:credentials|config)|  
aws_secret_access_key|aws_session_token|\\.azureV|azure_client_secret|identity_endpoint|\\.configVgcloudV|  
application_default_credentials\.jsonVsecretsVkubernetes.ioVserviceaccountVrun/secretsV|\\.dockerV/  
config\.json|169\.254\.169\.254/metadata/google/internalVprocV(?:self|[0-9]+)Venviron\b(?:cat|head|tail|less|  
more|stringsgrep)\b[^\]]*\.env\b", "cred_access", "") ) | filter signals != "  
| alter recon_tool = if(signals contains "recon", tool, null)  
  
// Group per host, PHP worker and account in 1-hour buckets | bin _time span = 1h | comp count() as events,  
count_distinct(recon_tool) as recon_tool_count, values(recon_tool) as recon_tools, values(signals) as signals,  
values(action_process_image_command_line) as action_process_image_command_line,  
values(actor_process_command_line) as actor_process_command_line,  
values(causality_actor_process_command_line) as causality_actor_process_command_line, first(event_id) as  
event_id by _time, agent_id, agent_hostname, causality_actor_process_image_name, actor_effective_username  
  
| alter all_signals = arraystring(signals, ",")  
  
// Threshold: any standalone signal, OR 3+ distinct recon tools, OR recon plus a follow-on scored signal | filter  
all_signals =~ "revshell|pipe_to_shell|writable_exec" or recon_tool_count >= 3 or (recon_tool_count >= 1 and  
all_signals =~ "download|staging|encoded|persistence|interpreter|cred_access")  
  
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40  
41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76  
77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108  
109 110 111 112 113 114 115 116 117  
  
// Description: This query attempts to identify post-exploitation command execution within the context of a PHP  
web server or worker process.  
  
// Notes: This hunting query is not specific to CVE-2026-109618 and CVE-2026-109619. It is capable of identifying  
exploitation of various web application vulnerabilities. This query may also produce false positive findings  
depending on the specific web applications running. We recommend validating if any identified activity is expected  
for the web applications running on systems.  
  
config case_sensitive = false | dataset = xdr_data | filter event_type = ENUM . PROCESS and event_sub_type =  
ENUM . PROCESS_START and action_process_image_name != null | fields _time , agent_hostname , agent_id ,  
event id , actor effective username , actor process image name , actor process command line ,
```

```
action_process_image_name , action_process_image_path , action_process_image_command_line ,
causality_actor_process_image_name , causality_actor_process_command_line
```

```
// Lowercase fields once | alter actor_effective_username = lowercase ( actor_effective_username ) ,
actor_process_image_name = lowercase ( actor_process_image_name ) , actor_process_command_line =
lowercase ( actor_process_command_line ) , action_process_image_name = lowercase (
action_process_image_name ) , action_process_image_path = lowercase ( action_process_image_path ) ,
action_process_image_command_line = lowercase ( action_process_image_command_line ) ,
causality_actor_process_image_name = lowercase ( causality_actor_process_image_name )
```

```
// Anchor: PHP web worker (or the launcher that started it) as causality, PHP worker as direct parent, // a shell
directly under an IIS worker, or the web service account outside an interactive login session | filter
causality_actor_process_image_name ~ = "(?:apache2|httpd|apachectl|php-fpm[0-9]*|php-cgi[0-9]*(?:
\\.cagefs)?|php-fcgi|sw-engine(?:-fpm)?|frankenphp|lsphp[0-9]*|lshttpd|litespeed|xampp-control|wampmanager|
laragon|de\\.appsolute\\.mampprohelper)(?:\\.exe)?$" or actor_process_image_name ~ = "(?:apache2|httpd|
php-fpm[0-9]*|php-cgi[0-9]*(?:\\.cagefs)?|php-fcgi|sw-engine(?:-fpm)?|frankenphp|lsphp[0-9]*|lshttpd|litespeed)(?:
\\.exe)?$" or ( causality_actor_process_image_name = "w3wp.exe" and actor_process_image_name in (
"cmd.exe" , "powershell.exe" , "pwsh.exe" ) ) or ( actor_effective_username ~ = "(?:^[^a-z0-9_-])(?:www-data|
apache|nginx|wwwrun|iusr)$" and coalesce ( causality_actor_process_image_name , "" ) ! ~ = "(?:sshd(?:-
session)?|screen|tmux|login|mosh-server|dropbear)$" ) or actor_effective_username contains "iis appool"
```

```
// Known-benign workloads (monitoring plugins, hosting control panels) and commands launched directly by CDK
Drive DMS scripts | filter action_process_image_command_line ! ~ = "(?:\\usr\\lib(?:64)?\\nagios\\plugins\\|\\usr\\
local\\cpanel\\|\\usr\\local\\psa\\|\\opt\\psa\\|\\usr\\local\\directadmin\\|\\usr\\local\\ispconfig\\|\\usr\\local\\virtualizor\\
richdocumentscode|phpsysinfo|librenms|zabbix)" and coalesce ( actor_process_command_line , "" ) ! ~ = "(?:
\\s+\\s+)?(?:\\vadp(?:\\.bin|3party|etc)\\|\\local\\bin\\uminfo)" // Phusion Passenger app spawns run id/uname/env from
their spawn helpers | filter coalesce ( action_process_image_command_line , "" ) ! ~ = "(?:\\usr\\bin\\)(?:id(?:\\s+
[a-z]+)?|uname(?:\\s+[-a-z]+)?|hostname|env)$|passenger-config\\s+system-metrics" or lowercase ( coalesce (
actor_process_command_line , "" ) ) ! ~ = "passengeragent\\s+spawn-[spawn(?:preparer|envsetupper)|shell|\\
helper-scripts\\rack-preloader\\.rb|passenger_ruby_wrapper|phusion_passenger" // RHEL-family /etc/profile runs
these in every bash login shell | filter coalesce ( action_process_image_command_line , "" ) ! ~ = "\\usr\\bin\\
(?:id\\s+[-ug]n|hostname)$" or lowercase ( coalesce ( actor_process_command_line , "" ) ) ! ~ = "(?:\\S*)?
bash\\s+[-a-z]*|^-bash$"
```

```
// parsed_lookup: the tool is the first command of its parent "sh -c" and its output is piped into a text filter (scripted
lookup) | alter tool = replace ( action_process_image_name , ".exe" , "" ) , shell_first_cmd = arrayindex (
regexextract ( coalesce ( actor_process_command_line , "" ) , "^-(?:\\S*)?(?:ba|da|k)?sh\\s+-c\\s+(\\S+)" ) , 0 ) | alter
parsed_lookup = if ( arrayindex ( regexextract ( coalesce ( shell_first_cmd , "" ) , "(\\^\\+)$" ) , 0 ) = tool and
actor_process_command_line ~ = "^-(?:\\S*)?(?:ba|da|k)?sh\\s+-c\\s+[^;&|]+\\|\\s*(?:e?grep|awk|cut|sed|wc|head|
tail|sort|tr|b" , 1 , 0 )
```

```
// Signals - one event can carry several | alter signals = concat ( // SCORED - host/user/network/container
recon, excluding scripted lookups (distinct tools are counted later) if ( parsed_lookup = 0 and ( tool in (
"whoami" , "hostname" , "hostnamectl" , "ifconfig" , "netstat" , "ss" , "arp" , "route" , "w" , "who" , "last" , "lastlog" ,
"getent" , "lsb_release" , "lscpu" , "printenv" , "env" , "getcap" , "capsh" , "systeminfo" , "ipconfig" , "nltest" ,
"tasklist" , "quser" , "qwinsta" ) or ( tool = "uname" and action_process_image_command_line ~ =
"\\buname\\s+(?:[-a-z]*[arv]|--all|--kernel-)" ) or ( tool = "id" and action_process_image_command_line ! ~ =
"\\bid\\s+[-ugnrz]+\\b" ) or ( tool = "ip" and action_process_image_command_line ~ = "\\bip\\s+(?:[-a-z0-9]+
\\s+)*(?:a|addr|address|r|route|n|neigh|link)\\b" ) or ( tool = "ps" and action_process_image_command_line ~ =
"\\bps\\s+(?:-?aux|-ef|-e|b|axjf|-auxf)" ) or ( tool in ( "net" , "net1" ) and action_process_image_command_line
~ = "\\bnet1(?:\\.exe)?\\s+(?:user|group|localgroup|accounts|view|share)\\b" ) or ( tool = "cat" and
action_process_image_command_line ~ = "\\etc\\(?:passwd|shadow|group|issue|os-release|hosts|hostname|
resolv\\.conf|crontab|sudoers)\\|\\proc\\(?:version|net\\(?:self[0-9]+)\\|\\(?:cgroup|mountinfo)\\|\\\\.dockerenv)" ) or (
tool = "find" and action_process_image_command_line ~ = "-perm\\s+(?:-4000|-u=s|\\V4000|-2000)" ) or ( tool
= "sudo" and action_process_image_command_line ~ = "\\bsudo\\s+\\b" ) ) , "recon" , "" ) ,
```

```
// SCORED - payload transfer to a file (curl with an output file, wget not to stdout), excluding /dev/null and
localhost if ( ( tool = "curl" and action_process_image_command_line ~ = "\\s(?:[-a-z]*o\\b|--output\\b|--remote-
name)|>\\s*\\S" ) or ( tool = "wget" and action_process_image_command_line ! ~ = "\\s[-a-z]*o\\s*(-(?:\\s|\\$))|--
```

```

output-document=|--spider" ) or ( tool in ( "tftp" , "fetch" ) and action_process_image_command_line ~ =
"(?:https?|ftp|tftp):\\V\\s-[a-z]*o(?:\\s|-$)" ) or ( tool = "certutil" and action_process_image_command_line ~
= "urlcache|verifycl" ) or ( tool = "bitsadmin" and action_process_image_command_line contains "/"
transfer" ) or ( tool in ( "powershell" , "pwsh" ) and action_process_image_command_line ~ =
"downloadstring|downloadfile|invoke-webrequest|biwr\\b|net\\.webclient|start-bitstransfer" ) ) and
action_process_image_command_line ! ~ = "-o\\s*\\dev\\null|--output-document=\\dev\\null|\\V\\(?:localhost|
127\\.0\\.0\\.1|\\[:1\\])?[:\\V]" , "download," , "" ) ,

// SCORED - chmod making a file in a tmp-class directory executable if ( tool = "chmod" and
action_process_image_command_line ~ = "\\bchmod\\s+(?:-[a-z]+\\s+)*(?:[0-7]?(?:[1357][0-7]{2}|[0-7][1357][0-7]|
[0-7]{2}[1357])|\\ugoa,+=rwst-]*[+=][rwst]*x\\ugoa,+=rwxst-]*\\s" and action_process_image_command_line ~ = "\\sV
(?:tmp|var\\tmp|dev\\shm|run\\shm)\\V" , "staging," , "" ) ,

// STANDALONE - binary executed from a writable/memory-backed dir, or from a deleted/memfd image if (
action_process_image_path ~ = "^\\(?:tmp|var\\tmp|dev\\shm|run\\shm|run\\user)\\V|^\\proc\\(?:self|[0-9]+)\\Vfd\\V|^\\V
memfd:\\ ( deleted \\ ) \\s*$" , "writable_exec," , "" ) ,

// STANDALONE - reverse/bind shell constructs if ( action_process_image_command_line ~ = "\\dev\\(?:tcp|
udp)\\V|(?:^|[\\s;&|\\V])(?:nc|ncat|netcat)(?:\\.[a-z]+)?\\s+(?:\\S+\\s+){0,6}(?:-[a-z]{0,3}[ec]|--exec|--sh-exec)\\s+S*(?:sh|
bash|cmd|python[0-9.]*|perl)\\b|\\bsocat\\b[^&|]{0,80}exec:\\bmkfifo\\b|\\b(?:ba|da|z|a)?sh\\s+-i\\b|socket\\.socket \\ ( |
fsockopen \\ ( | \\ bpty \\ . spawn \\ ( , " revshell , " , " ) ,

// STANDALONE - remote content piped straight into an interpreter if(action_process_image_command_line
~ = " ( ? : curl | wget | fetch ) \\ b [ ^ | ] * \\ | \\ s * ( ? : \\ \\ S * \\ / ) ? ( ? : ( ? : ba | da | z | a ) ? sh | python [ 23 ] ? | perl |
php ) \\ b " , " pipe_to_shell , " , " ) ,

// SCORED - encoded/obfuscated execution if(action_process_image_command_line ~ = " base64 \\ s + ( ? : -
d | -- decode ) | frombase64string | - e ( ? : nc | ncodedcommand ) ? \\ s + [ a - z0 - 9 + \\ / ] { 40 , } | echo \\ s + [ a -
z0 - 9 + \\ / ] { 40 , } | \\ bopenssl \\ s + ( ? : enc \\ b [ ^ | ; ] * \\ s - d \\ b | base64 \\ s + - d \\ b ) | \\ bxxd \\ s + ( ? : - [ a - z ]
+ \\ s + ) * - [ a - z ] * r " , " encoded , " , " ) ,

// SCORED - persistence / backgrounding (crontab edits, authorized_keys writes, new users, nohup/setsid of
tmp-class binaries or shells, // scheduled tasks, Run keys, systemd units, chatr +i) if((tool = " crontab " and
action_process_image_command_line ! ~ = "\\ bcrontab \\ s + - l \\ s * $ " ) or
(action_process_image_command_line ~ = " ( ? : > > ? \\ s * | \\ btee \\ s + ( ? : - a \\ s + ) ? | \\ b ( ? : cp | mv | install |
ln ) \\ s + ( ? : - \\ S + \\ s + ) * \\ S + \\ s + ) \\ S * authorized_keys " and action_process_image_command_line !
~ = "\\ bcp \\ s + \\ S * \\ \\ . ssh \\ / id_ [ a - z0 - 9 ] + \\ . pub \\ s + \\ S * \\ \\ . ssh \\ / authorized_keys " ) or
action_process_image_command_line ~ = "\\ b ( ? : useradd | adduser ) \\ b | \\ b ( ? : nohup | setsid ) \\ s + ( ? : - \\ S
+ \\ s + ) * ( ? : \\ . \\ / | \\ / ( ? : tmp | var \\ / tmp | dev \\ / shm | run \\ / shm ) \\ / | ( ? : ba | da | z ) ? sh \\ s + - [ ci ] \\ b ) |
schtasks \\ s + \\ create | currentversion . run \\ b | \\ bsystemctl \\ s + ( ? : -- user \\ s + ) ? enable \\ b | \\ . config \\ /
systemd \\ / user \\ / | \\ / etc \\ / systemd \\ / system \\ / | \\ bchatr \\ s + + [ a - z ] * i " , " persistence , " , " ) ,

// SCORED - inline interpreter one-liners, excluding PHP version/feature probes and Ruby gem-path probes
if(action_process_image_command_line ~ = " ( ? : ^ [ ^ a - z0 - 9 . _ - ] ) ( ? : python [ 23 ] ? ( ? : \\ . [ 0 - 9 ] + ) ? \\
s + - c | perl \\ s + - e | ruby \\ s + - e | node ( ? : js ) ? \\ s + - e | lua \\ s + - e | php [ 0 - 9 . ] * \\ s + - r ) \\ b " and
action_process_image_command_line ! ~ = " phpversion \\ ( | php_version | date \\ ( ' t ' ) | function_exists \\ ( |
extension_loaded \\ ( | gem \\ . path , " interpreter , " , " ) ,

// SCORED - cloud/container credential and secret access (credential files, env-var names, instance metadata
service, .env reads) if ( action_process_image_command_line ~ = "\\ .aws\\(?:credentials|config)|
aws_secret_access_key|aws_session_token|\\ .azure\\|azure_client_secret|identity_endpoint|\\ .config\\gcloud\\|
application_default_credentials\\.json|\\secrets\\kubernetes\\.io\\serviceaccount\\Vrun\\secrets\\|\\ .docker\\V
config\\.json|169\\.254\\.169\\.254|metadata\\.google\\.internal\\Vproc\\(?:self|[0-9]+)\\Venviron|\\b(?:cat|head|tail|less|
more|strings|grep)\\b[^|;]*\\.env\\b" , "cred_access," , "" ) ) | filter signals != ""

| alter recon_tool = if ( signals contains "recon" , tool , null )

// Group per host, PHP worker and account in 1-hour buckets | bin_time span = 1h | comp count ( ) as events ,
count_distinct ( recon_tool ) as recon_tool_count , values ( recon_tool ) as recon_tools , values (
signals ) as signals , values ( action_process_image_command_line ) as
action_process_image_command_line , values ( actor_process_command_line ) as

```

```

actor_process_command_line ,      values ( causality_actor_process_command_line ) as
causality_actor_process_command_line ,      first ( event_id ) as event_id      by _time , agent_id ,
agent_hostname , causality_actor_process_image_name , actor_effective_username

| alter all_signals = arraystring ( signals , "," )

// Threshold: any standalone signal, OR 3+ distinct recon tools, OR recon plus a follow-on scored signal | filter
all_signals ~ = "revshell|pipe_to_shell|writable_exec"      or recon_tool_count > = 3      or ( recon_tool_count > = 1
and all_signals ~ = "download|staging|encoded|persistence|interpreter|cred_access" )

// Description: This query identifies potential inbound exploitation attempts from publicly routed IP addresses using
Palo Alto Networks NGFW Threat Logs.

config case_sensitive = false | dataset = panw_ngfw_threat_raw | filter sub_type = "vulnerability" | filter to_zone !
~ = "internet|untrust|outside|external|wan" | filter not ( incidr ( source_ip , "10.0.0.0/8" ) or incidr ( source_ip ,
"172.16.0.0/12" ) or incidr ( source_ip , "192.168.0.0/16" ) ) | filter to_string ( threat_id ) contains "543004" or
threat_name ~ = "php|multipart|upload.progress|longjmp|CVE-2026-10961[89]" | comp count ( ) as hits ,
count_distinct ( source_ip ) as sources , values ( source_ip ) as source_ips , values ( action ) as actions , values ( severity )
as severities , min ( _time ) as first_seen , max ( _time ) as last_seen by threat_name , threat_id , dest_ip , nat_dest ,
dest_port , to_zone | sort desc hits

1 2 3 4 5 6 7 8 9 10 11 12

// Description: This query identifies potential inbound exploitation attempts from publicly routed IP addresses using
Palo Alto Networks NGFW Threat Logs.

config case_sensitive = false | dataset = panw_ngfw_threat_raw | filter sub_type = "vulnerability" | filter to_zone ! ~
= "internet|untrust|outside|external|wan" | filter not ( incidr ( source_ip , "10.0.0.0/8" ) or incidr ( source_ip ,
"172.16.0.0/12" ) or incidr ( source_ip , "192.168.0.0/16" ) ) | filter to_string ( threat_id ) contains "543004" or
threat_name ~ = "php|multipart|upload.progress|longjmp|CVE-2026-10961[89]" | comp count ( ) as hits ,
count_distinct ( source_ip ) as sources , values ( source_ip ) as source_ips , values ( action ) as actions ,
values ( severity ) as severities , min ( _time ) as first_seen , max ( _time ) as last_seen      by threat_name ,
threat_id , dest_ip , nat_dest , dest_port , to_zone | sort desc hits

```

Palo Alto Networks Product Protections for CVE-2026-109618 and CVE-2026-109619

Palo Alto Networks customers can leverage several product protections and updates to identify and defend against these vulnerabilities.

Organizations can access tailored adversary intelligence, direct analyst consultations, and specialized exposure tracking through Unit 42 Threat Intel Services and its Deep and Dark Web Service .

If you think you might have been compromised or have an urgent matter, get in touch with the Unit 42 Incident Response team or call:

North America: Toll Free: +1 (866) 486-4842 (866.4.UNIT42)

UK: +44.20.3743.3660

Europe and Middle East: +31.20.299.3130

Asia: +65.6983.8730

Japan: +81.50.1790.0200

Australia: +61.2.4062.7950

India: 000 800 050 45107

South Korea: +82.080.467.8774

Cortex XDR and XSIAM

Cortex XDR and XSIAM help protect against post-exploitation activities using the multi-layer protection approach. provide multilayered protection against post-exploitation activities. Cortex XDR also provides visibility into installed software across endpoints and servers, including on-premises and legacy environments, to help security teams

identify systems running vulnerable PHP versions and prioritize remediation.

Cortex Cloud and Prisma Cloud

Cortex Cloud and Prisma Cloud help identify vulnerable PHP versions across code, registry images and runtime instances. Cortex Cloud's Supply Chain Attacks Portal helps teams identify affected assets and take action to prevent vulnerable components from being reintroduced. Additionally, Cortex CLI integrates into CI/CD pipelines to scan images and enforce policies that block vulnerable images before they reach registries and runtime environments.

Cortex Xpanse

Cortex Xpanse can identify exposed devices on the public internet and escalate these findings to defenders. Customers can enable alerting on this risk by ensuring that the PHP Attack Surface Rule is enabled. Identified findings can either be viewed in the Threat Response Center or in the incident view of Expander. These findings are also available for Cortex XSIAM customers who have purchased the ASM module.

Next-Generation Firewall

Next-Generation Firewall with the following security subscriptions can help block the attacks:

Advanced Threat Prevention Plus with Frontier Virtual Patching since 10:21 p.m. PT on October 20, 2026, via Threat ID 543004 .

Advanced Threat Prevention with best practices since October 21, 2026 via the following Threat Prevention signatures 543004 , 97622 , 97623 , 97624 , and 97625 .

Additional Resources

Impact of Critical Vulnerabilities in PHP on Palo Alto Networks Products — Advisory, Palo Alto Networks

Millions of PHP Instances Exposed: What Organizations Need To Do Right Now — Lee Klarich, Palo Alto Networks

Security advisory: Critical remote code execution vulnerabilities, update now - PHP

Updated October 21, 2026, at 9:30 a.m. PT to add Palo Alto Networks security advisory, update links in additional resources, and update product protection section.

Updated October 21, 2026, at 1:00 p.m. PT to update the product protection section.

Updated October 22, 2026, at 4:30 p.m. PT to add additional product protection information for Advanced Threat Prevention, Cortex Cloud, and Prisma Cloud and make light copyedits.

Updated October 23, 2026, at 12:30 p.m. PT to add details on agentic AI post-exploitation activity and Managed Threat Hunting queries.

Updated 23 October, 2026 at 12:34 PM PDT --> Back to top

Tags CVE-2026-109618 CVE-2026-109619 PHP2Shell

Threat Research Center

Next: EMudhra Intermediate CA Certificates Compromised

Table of Contents

Related Articles

Related Vulnerabilities Resources

High Profile Threats October 20, 2026

Active Exploitation of May 2026 PAN-OS CVE-2026-0257 (Updated October 20)

CVE-2026-0257

Vulnerability

[Read now](#)

High Profile Threats September 30, 2026

Threat Brief: NetScaler Zero Days CVE-2026-88771 and CVE-2026-88772 Exploited in the Wild (Updated September 30)

Citrix

CVE-2026-88771

CVE-2026-88772

[Read now](#)

Insights August 4, 2026

The Frontier AI Vulnerability Burst: Industrializing Autonomous Zero-Day Discovery in Open-Source Software AI

Frontier AI

Vulnerability Exploitation

[Read now](#)

Threat Research July 30, 2026

Chinese-Speaking Threat Actor Harnesses AI Models for Autonomous Cyberattacks

ChatGPT

Claude code

CVEs

[Read now](#)

Threat Research July 17, 2026

Three Steps to the Terminal: A Siemens ROX II Zero-Day Trilogy

Command injection

CVE-2025-40947

CVE-2025-40948

[Read now](#)

High Profile Threats May 6, 2026

Threat Brief: Exploitation of PAN-OS Captive Portal Zero-Day for Unauthenticated Remote Code Execution

CVE-2026-0300

EarthWorm

PAN-OS

[Read now](#)

High Profile Threats May 5, 2026

Copy Fail: What You Need to Know About the Most Severe Linux Threat in Years

Containers

CVE-2026-31431

Kubernetes

[Read now](#)

Threat Research April 16, 2026

A Deep Dive Into Attempted Exploitation of CVE-2023-33538

Botnet

Command injection

CVE-2023-33538

[Read now](#)

High Profile Threats February 19, 2026

VShell and SparkRAT Observed in Exploitation of BeyondTrust Critical Vulnerability (CVE-2026-1731)

Bash

CVE-2026-1731

PowerShell

[Read now](#)