AWS's Log4Shell Hot Patc
Container Escape and Priv

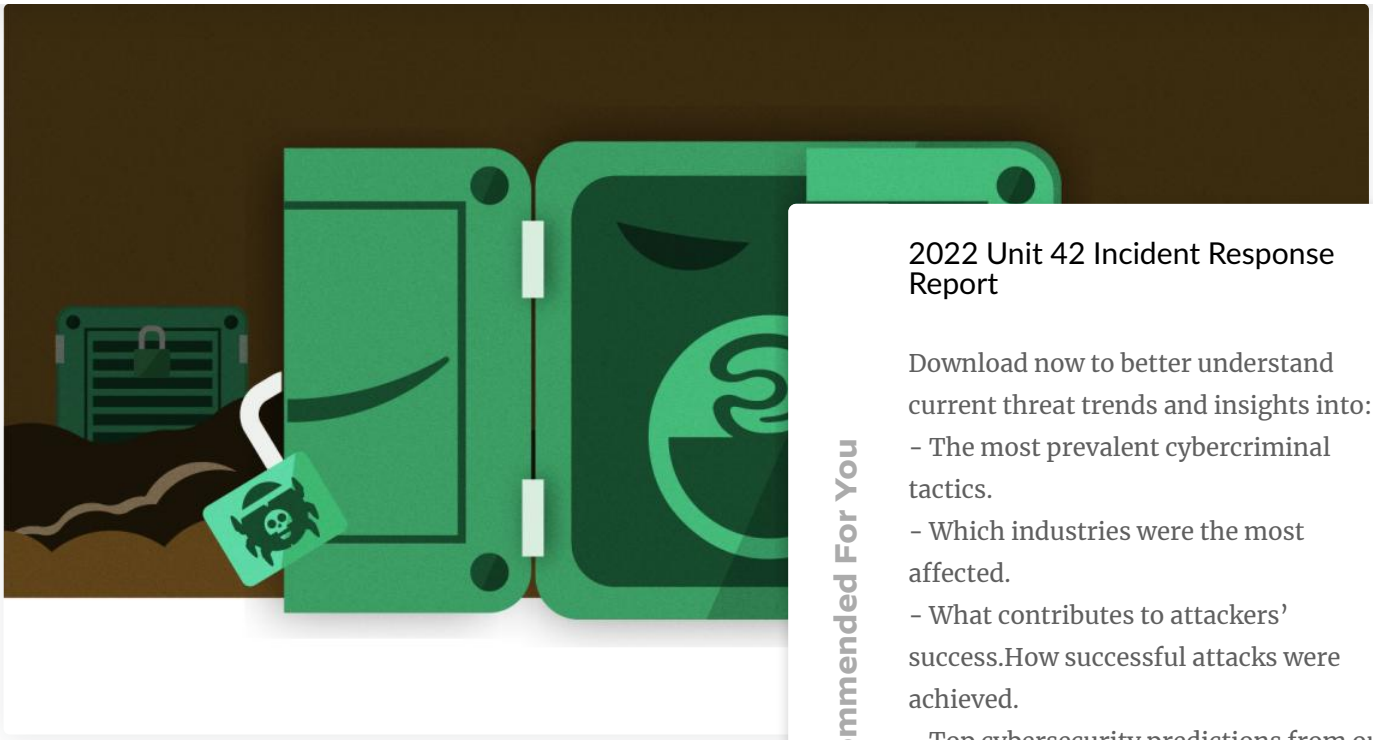
By Yuval Avrahami
April 19, 2022 at 3:00 PM
Category: Cloud, Vulnerability
Tags: Apache Log4j, AWS, container escape, containers, CVE-20
CVE-2022-0070, CVE-2022-0071, log4j, privilege escalation

This post is also available in: 日本語 (Japanese)

Executive Summary

Following Log4Shell, AWS released several hot patch solutions that monitor for vulnerable Java applications and Java containers and patch them on the fly. Each solution suits a different environment, covering standalone servers, Kubernetes clusters, Elastic Container Service (ECS) clusters and Fargate. The hot patches aren't exclusive to AWS environments and can be installed onto any cloud or on-premises environment.

Unit 42 researchers identified severe security issues within these patching solutions and partnered with



Containers can escape regardless of whether they run Java
host runs Bottlerocket, AWS's hardened Linux distribution f
namespaces or as a non-root user are affected as well. Unit
3101, CVE-2022-0070 and CVE-2022-0071 to track the vu

AWS released a fixed version for each hot patch solution or

1. Version 1.1-16 of the `log4j-cve-2021-44228-hotpatch` package, which bundles the hot patch service.

2. Version 1.1-16 of the `kubernetes-log4j-cve-2021-44228-node-agent` Daemonset, which installs the updated package.

3. Version 1.02 of Hotdog, a hot patch solution for Bottlerocket hosts based on Open Container Initiative (OCI) hooks.

Unit 42 advises anyone who installed any of these hot patches to upgrade to a fixed version. Note that starting from Dec. 17, 2021, JDK packages (Java installations) on Amazon Linux automatically installed the `log4j-cve-2021-44228-hotpatch` package. Alternatively, users who are confident their applications are patched against Log4Shell can disable the hot patch service following the instructions in the Mitigations section below.

Related Unit 42 topics	Container escape, privilege escalation, cloud, Apache log4j

Table of Contents

Overview of AWS Log4Sh

Log4Shell proved itself as one of the worst vulnerabilities of
issue at scale, AWS open-sourced several hot patch solution
Hot patching is the process of injecting a fix to a vulnerable
short-term solution until a new, fixed version of the applicat

AWS released three hot patching solutions that detect processes and containers running vulnerable Java applications and patch them on the fly:

1. A hot patch service bundled in an RPM package. Starting from Dec. 17, 2021, this service is automatically installed with Amazon Linux JDK (Java) packages. Fargate customers could've asked for this service to be installed on the hosts running their containers.

2. A hot patch Daemonset for Kubernetes clusters, which installs the aforementioned hot patch service on all nodes.

3. Hotdog, a hot patch solution bundled as a set of OCI hooks. Hotdog is primarily intended for Bottlerocket hosts.

These solutions cover most compute environments, from Kubernetes clusters to ECS clusters, Fargate containers and standalone servers. They aren't exclusive to AWS environments, and can be installed onto other cloud environments or on-premises.

Root Cause Analysis

AWS's hot patch solutions continuously search for Java processes and patch them against Log4Shell on the fly. Any process running a binary named “java“ – inside or outside of a container – is considered a candidate for the hot patch.

To patch Java processes inside containers, the hot patch sol
example, they run the container's "java" binary twice: once t
inject the hot patch. The issue was that they invoked contai
them. That is, the new processes would run without the limi
processes.

For example, the "java" binary was invoked in the container
(excluding the user namespace). But aside from that, it was s
without the isolation technologies that normally confine cor
also ran as the root user regardless of the container's user.

A malicious container therefore could have included a malic
installed hot patch solution into invoking it with elevated pr
then abuse its elevated privileges to escape the container ar
hot patch solutions now properly containerize container bin

Aside from containers, the hot patch service also patched h
malicious unprivileged process could have created and run a
hot patch service into executing it with elevated privileges. The fixed hot patch service now spawns “java” binaries with the same privileges as the Java process being patched.

Container Escape Demo

To verify the vulnerability is exploitable, we built a proof of concept (PoC) container image. When deployed to a cluster or VM that runs a vulnerable version of a hot patch solution, the container exploits the vulnerabilities to escape and gain root code execution on the underlying host. It then sends a reverse shell to an attacker-controlled server.

In the demo video below, a user installed the hot patch Daemonset to an EKS cluster. The demo then simulates a supply chain attack by showing what happens when the user inadvertently runs a malicious container image that exploits the hot patch.

Video 1. CVE-2021-3100 exploit demo.

While the demo showcases a supply chain attack, existing c
network payload) can also exploit the issues to escape and t
decided not to share the exploit's implementation details at
weaponizing it.

Impact

Given the urgency surrounding Log4Shell, users may have deployed hot patches at scale, inadvertently putting container environments at risk. Even after Java applications were patched against Log4Shell, users may have kept the hot patch running for defense-in-depth as there isn't a strong incentive to remove it.

Containers are often used as a security boundary between applications running on the same machine. A container escape allows an attacker to extend a campaign beyond a single application and compromise neighboring services. In Kubernetes clusters, a single container escape is unfortunately sometimes enough to take over the entire cluster.

The issues are exploitable regardless of the container configuration, so even environments that enable advanced isolation techniques like running containers in user namespaces or as a non-root user are affected.

Mitigations

AWS released a fix for each hot patch solution. Once a host runs a fixed version, container escape and privilege escalation are no longer possible.

1. In Kubernetes clusters, you can install the fixed hot patch
provided by AWS. Note that only deleting the hot patch Da
service from your nodes. **Updated April 25**: Currently, there
Debian-based hosts (Debian and Ubuntu). See this GitHub t
fixed Daemonset version for Debian-based hosts was releas
Debian `log4j-cve-2021-44228-hotpatch` package is

2. On standalone hosts, you can upgrade by running `yum u`
`hotpatch`.

3. Hotdog users need to upgrade to the latest version.

Alternatively, if you're confident that your environment is pa
the hot patch service on a host by running `sudo touch /`
`hotpatch.kill`. To disable Hotdog, run `apiclient set`
`enabled=false`.

Prisma Cloud customers can identify affected hosts under t
the hot patch packages and alerts customers on VMs runnir
vulnerabilities, use the Amazon Linux Security Advisories (A
2021-1554, ALAS-2021-1732, ALAS-2022-1580 and ALAS-2022-1773.

Figure 1. Prisma Cloud detects and alerts on vulnerable log4

Palo Alto Networks Prisma Cloud, Cortex XDR and Next-Ge
follow-on attacker activities and disrupt command and control communications like the reverse shell used in the demo.

Safely Interacting With Containers

CVE-2021-3100, CVE-2021-3101, CVE-2022-0070 and CVE-2022-0071 add to a long list of container escape vulnerabilities that arise from a host process directly interacting with a running container. Simple tasks like copying files or spawning a new containerized process can have surprising outcomes when the container is malicious.

If you're building software around containers, defer to an established container runtime like runc for operations involving a container's processes or filesystem. Although they have also had their share of vulnerabilities, container runtimes are by far the most vetted and mature programs for safely interacting with containers.

as soon as possible. Multitenant container environments and clusters running untrusted images are especially at risk.

If you're still patching against Log4Shell, prioritize that effort first. While the presented issues can lead to severe attacks against container environments, Log4Shell has rightfully earned its spot as one of the worst vulnerabilities of all time and is still being actively exp

We'd like to thank AWS for their partnership and coordinati
efficiently. As Log4Shell exploitation peaked, AWS's hot pat
attacks. With these vulnerabilities fixed, it's now possible to
while also keeping container environments secure.

Additional Resources

- Unit 42 analysis of Log4Shell
- AWS advice on mitigating Log4Shell in container environ
- Prisma Cloud Mitigations for Log4Shell

Disclosure Timeline

- **Dec. 14:** AWS releases hot patch package with support f
- **Dec. 20**: Unit 42 researchers identify the issue.
- **Dec. 21**: Advisory sent to AWS.
- **Dec. 22**: AWS acknowledges the issue.
- **Dec. 23**: AWS releases fixes and advisories for affected components.
- **Dec. 27**: Unit 42 reports bypasses for the initial fixes to AWS.
- **Feb. 9**: Unit 42 researchers meet with AWS security to discuss fixes.
- **April 1**: AWS shares fixed versions for Unit 42 review.
- **April 4**: Unit 42 points out a few remaining issues.
- **April 19**: AWS releases final fixes and advisories; Unit 42 discloses the vulnerabilities publicly.

Updated May 5, 2022, at 11:10 a.m. PT.

Sign up to receive the latest news, cyber threat intelligence and research from us

Email address

Subscribe

I'm not a robot

reCAPTCHA
Privacy - Terms

By submitting this form, you agree to our Terms of Use and acknowle

Recommended For You

2022 Unit 42 Incident Response Report

Download now to better understand current threat trends and insights into:
- The most prevalent cybercriminal tactics.
- Which industries were the most affected.
- What contributes to attackers' success.How successful attacks were achieved.
- Top cybersecurity predictions from our incident responders.
- Actionable recommendations to get ahead of future threats.

Get the report

Popular Resources

Resource Center

Blog

Communities

Tech Docs

Unit 42

Sitemap

Legal Notices

Privacy

Terms of Use

Documents

Account