Background:

Over the past few days, the Zscaler ThreatLabz team has been closely monitoring the reports of potential RCEs in Spring Cloud Framework and Spring Cloud Function.

Spring is an open-source lightweight Java platform which many developers use as their application development framework. As part of the Spring echo system, Spring Cloud is a component using which one can write cloud agnostics code or develop applications and make them working on well known cloud services such as AWS, Azure, Alibaba etc. The Spring Cloud Function is one of the subcomponents of Spring Cloud Function which enables developers to do serverless programming.

Leveraging from our internal research and the published articles, github posts and POCs, at the moment our understanding is that there could potentially be more than one issue in Spring Cloud Framework and sub-component Spring Cloud Function.

At the moment, we are discussing two issues/vulnerabilities here.

1. CVE-2022-22963 - Spring Expression Resource Access Vulnerability which can provide access to the critical systems/resources to the unauthenticated adversary. At the moment it is categorized as a medium severity issue.
2. CVE-2022-22965 aka Spring4Shell or SpringShell - Spring Framework RCE via Data Binding on JDK 9+. This vulnerability is categorized as Critical.

What are the issues?

1. CVE-2022-22963

Spring Expression Resource Access Vulnerability was found in Spring Cloud Function versions 3.1.6 and 3.2.2 or prior. The adversaries can exploit this vulnerability by sending a crafted HTTP request packet with the specific HTTP header named, spring.cloud.function.routing-expression, in the HTTP request packet. This parameter is treated as SpEL expression when the routing is in use. Vulnerable version of the Spring Cloud Function, while parsing this specific HTTP header, spring.cloud.function.routing-expression, along with the malformed SpEL expression, can allow an adversary to gain access to the critical resources on servers, systems which can allow adversary to perform further malicious activities. The vulnerability has been classified as “medium” at the moment. However, the actual impact of exploiting this vulnerability is unknown.

Vulnerable versions for CVE-2022-22963: Spring Cloud Function versions between 3.1.6 or prior and 3.2.2 or prior seem to be vulnerable to the Expression Resource Access Vulnerability.

Spring Foundation Version Patched for CVE-2022-22963: The Spring Foundation version 3.1.7 and 3.2.3 have been released to patch this critical vulnerability.

Zsclaer strongly recommends upgrading to these versions depending on what current version of Spring Foundation is deployed. 2. CVE-2022-22965 [Spring4Shell OR SpringShell]

There is a severe Remote Code Execution vulnerability in Spring Core JDK9+. This vulnerability can allow an unauthenticated attacker to execute arbitrary code on the target system. As per the sources available publicly, exploiting this vulnerability in certain configurations is relatively easier because it can be exploited with just simple crafted HTTP requests when sent to a vulnerable server. While in other configurations, it may not be that easy for an adversary to exploit and may require him/her to perform additional research.

To exploit this vulnerability, the vulnerable system should have DataBinder enabled, and exploitation depends majorly on the Servlet container of the application. As per the researcher who confirmed the exploitation, when Spring is deployed to Apache Tomcat, the WebAppClassLoader is accessible, which allows an attacker to call getters and setters and can write a malicious file to disk. However, if Spring is deployed using the Embedded Tomcat Servlet Container, the classloader is a LaunchedURLClassLoader which has limited access.

The details on the vulnerability and possible exploitation through a proof-of-concept is described here.

Mitigations

- CVE-2022-22963 Users of affected versions should upgrade to 3.1.7, 3.2.3. Releases that have fixed this issue include: Spring Cloud Function - 3.1.7 - 3.2.3
- CVE-2022-22965 Users of affected versions should apply the following mitigation: 5.3.x users should upgrade to 5.3.18+, 5.2.x users should upgrade to 5.2.20+. There are other mitigation steps for applications that cannot upgrade to the above versions. Releases that have fixed this issue include: Spring Framework - 5.3.18+ - 5.2.20+

Best Practices/Guidelines To follow:

- Limit the impact from a potential compromise by restricting lateral movement with identity-based micro-segmentation (Zscaler Workload Segmentation) and a Zero Trust architecture.
- Safeguard crown jewel applications by limiting lateral movement using Zscaler Private Access, especially with application security modules turned on.
- Route all server traffic through Zscaler Private Access with additional application security module enabled and Zscaler Internet Access, which will provide the right visibility to identify and stop malicious activity from compromised systems/servers.
- Restrict traffic to the critical infrastructure from the allowed list of known-good destinations.
- Ensure you are inspecting all SSL traffic.
- Turn on Advanced Threat Protection to block all known command-and-control domains. This will provide additional protection in case the adversary exploits this vulnerability to implant malware.
- Extend command-and-control protection to all ports and protocols with the Advanced Cloud Firewall (Cloud IPS module), including emerging C2 destinations. Again, this will provide additional protection in case if the adversary exploits this vulnerability to implant malware.
- Use Advanced Cloud Sandbox to prevent unknown malware delivered as part of a second stage payload.

Zscaler Coverage:

Zscaler's ThreatLabZ team has deployed protection for all known POCs for these vulnerabilities.

Advanced Threat Protection

- HTML.Exploit.Spring4Shell
- HTML.Exploit.CVE-2022-22963

Zscaler Private Access w/ Application Security

- HTML.Exploit.CVE-2022-22963
- HTML.Exploit.Spring4Shell

Additional References:

1. https://www.lunasec.io/docs/blog/spring-rce-vulnerabilities/#exploit-scenario-overview 2. https://www.praetorian.com/blog/spring-core-jdk9-rce/ 3. https://github.com/spring-projects/spring-framework/commit/7f7fb58dd0dae86d22268a4b59ac7c72a6c22529 4. https://spring.io/blog/2022/03/29/cve-report-published-for-spring-cloud-function 5. https://tanzu.vmware.com/security/cve-2022-22963 6. https://www.cyberkendra.com/2022/03/rce-0-day-exploit-found-in-spring-cloud.html 7. https://www.cyberkendra.com/2022/03/springshell-rce-0-day-vulnerability.html

About ThreatLabz ThreatLabz is the security research arm of Zscaler. This world-class team is responsible for hunting new threats and ensuring that the thousands of organizations using the global Zscaler platform are always protected. In addition to malware research and behavioral analysis, team members are involved in the research and development of new prototype modules for advanced threat protection on the Zscaler platform, and regularly conduct internal security audits to ensure that Zscaler products and infrastructure meet security compliance standards. ThreatLabz regularly publishes in-depth analyses of new and emerging threats on its portal, research.zscaler.com.

- Security Research
- Insights and Research
-

Authors

Jithin Nair

Amit Banker

Recommended for You

Analysis of BlackGuard - A New Info Stealer Malware Being Sold In A Russian Hacking Forum

Conti Ransomware Attacks Persist With an Updated Version Despite Leaks

Lapsus$ Attack on Okta: How to Evaluate the Impact to your Organization

Midas Ransomware: Tracing the Evolution of Thanos Ransomware Variants