Security updates

CVE-2015-3253

CVE-2016-6814

CVE-2020-17521

Reporting problems

Security updates

Here you can find information about security patches or updates released for Apache Groovy. Note that unless specified otherwise, no binary or source patches are available. To obtain a security fix, you need to upgrade to the latest maintained version of Apache Groovy.

Releases prior to 2.4.4 were not released under Apache so no official patches for security updates are available for older versions.

CVE-2015-3253 Apache Groovy Information Disclosure

Severity: Important

Vendor: The Apache Software Foundation

Versions Affected:

- Unsupported Codehaus versions of Groovy from 1.7.0 to 2.4.3
- Fixed in version 2.4.4

Impact:

Remote execution of untrusted code, DoS

Description:

When an application has Groovy on the classpath and uses standard Java serialization mechanisms to communicate between servers, or to store local data, it is possible for an attacker to bake a special serialized object that will execute code directly when deserialized. All applications which rely on serialization and do not isolate the code which deserializes objects are subject to this vulnerability.

Mitigation:

Apache Groovy 2.4.4 is the first supported release under the Apache Software Foundation. It is strongly recommended that all users using serialization upgrade to this version. If you cannot upgrade or rely on an older, unsupported version of Groovy, you can apply the following patch on the `MethodClosure` class (`src/main/org/codehaus/groovy/runtime/MethodClosure.java`):

```
  public class MethodClosure extends Closure {
+    private Object readResolve() {
+        throw new UnsupportedOperationException();
+    }
```

Alternatively, you should make sure to use a custom security policy file (using the standard Java security manager) or make sure that you do not rely on serialization to communicate remotely.

Credit:

This vulnerability was discovered by:

- cpnrodzc7 working with HP's Zero Day Initiative

References:

- CVE-2015-3253: Remote execution of untrusted code (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-3253)
- https://groovy-lang.org/security.html (https://groovy-lang.org/security.html)

CVE-2016-6814 Apache Groovy Information Disclosure

Severity: Important

Vendor: The Apache Software Foundation

Versions Affected:

- Unsupported Codehaus versions of Groovy from 1.7.0 to 2.4.3
- Apache Groovy 2.4.4 to 2.4.7
- Fixed in version 2.4.8

Impact:

Remote execution of untrusted code, DoS

Description:

When an application with Groovy on classpath uses standard Java serialization mechanisms, e.g. to communicate between servers or to store local data, it is possible for an attacker to bake a special serialized object that will execute code directly when deserialized. All applications which rely on serialization and do not isolate the code which deserializes objects are subject to this vulnerability. This is similar to CVE-2015-3253 but this exploit involves extra wrapping of objects and catching of exceptions which are now safe guarded against.

Mitigation:

Users of Groovy relying on (de)serialization with the affected versions should apply one of the following mitigations:

- Isolate the code doing the (de)serialization
- Upgrade to Apache Groovy 2.4.8 or later
- Users of older versions of Groovy can apply the following patch to the `MethodClosure` class (`src/main/org/codehaus/groovy/runtime/MethodClosure.java`):

```
public class MethodClosure extends Closure {
+    private void readObject(java.io.ObjectInputStream stream) throws
IOException, ClassNotFoundException {
+        if (ALLOW_RESOLVE) {
+            stream.defaultReadObject();
+        }
+        throw new UnsupportedOperationException();
+    }
```

Credit:

This vulnerability was discovered by:

- Sam Thomas of Pentest Limited working with Trend Micro’s Zero Day Initiative

History:

- 2016-09-20 Original advisory
- 2017-01-12 Updated information on affected versions

References:

- CVE-2016-6814: Remote execution of untrusted code (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-6814)

- https://groovy-lang.org/security.html (https://groovy-lang.org/security.html)

CVE-2020-17521 Apache Groovy Information Disclosure

Severity: Important

Vendor: The Apache Software Foundation

Versions Affected:

Unsupported Codehaus versions of Groovy from 2.0 to 2.4.4. Apache Groovy versions 2.4.4 to 2.4.20, 2.5.0 to 2.5.13, 3.0.0 to 3.0.6, and 4.0.0-alpha-1.

Fixed in versions 2.4.21, 2.5.14, 3.0.7, 4.0.0-alpha-2

Impact:

This vulnerability potentially impacts Unix-like systems, and very old versions of Mac OSX and Windows. On such OS versions, Groovy may create temporary directories within the OS temporary directory which is shared between all users on affected systems. Groovy will create such directories for internal use when producing Java Stubs (very low impact) or on behalf of user code via two extension methods[4,5] for creating temporary directories. If Groovy user code uses either of these extension methods, and stores executable code in the resulting temporary directory, then the risk is high, since this can lead to local privilege escalation. If such Groovy code is making use of the temporary directory to store sensitive information, then the risk is medium, since such information could be exposed or modified.

When analyzing the impact of this vulnerability, here are the important questions to ask:

Is the Groovy code running on a machine with an impacted operating system? Do other users have access to the machine running the Groovy code? Does the Groovy code create temporary directories using Groovy's createTempDir extension methods[4,5]?

If you answer no to any of these questions, you are not affected. If you answered yes, does the Groovy code write or store executable code in the temporary directory? If you answer yes, the risk is high, and can lead to local privilege escalation. Does the Groovy code write sensitive information, like API keys or passwords, into the temporary directory? If you answer yes, the risk is medium, and information may be exposed or modified.

Description:

Groovy was making use of a method in the JDK which is now flagged as not suitable for security-sensitive contexts. In addition, Groovy wasn't checking a flag related to successful creation of the temporary directory which leads to a race condition whereby the vulnerability exists[1].

For the fixed versions, Groovy 2.5 and above is now using a newer JDK method which creates a directory that is only readable by the user running the Groovy code. The same is true for the fixed Groovy 2.4 version except if running on a pre-JDK7 version of the JDK in which case a fallback implementation is used which now checks for successful creation of the temporary directory. This eliminates the high-risk scenario involving the race condition whereby executables or information could be modified, but still leaves the potential for sensitive information leakage. Groovy 2.4/JDK 6 users are recommended to use the `java.io.tmpdir` mitigation.

Mitigation:

Setting the `java.io.tmpdir` system environment variable to a directory that is exclusively owned by the executing user will fix this vulnerability for all operating systems and all Groovy versions.

Users who cannot easily move to the fixed Groovy versions may wish to consider using the JDK's Files#createTempDirectory method instead of the Groovy extension methods.

Credit:

This vulnerability was discovered by Jonathan Leitschuh (https://twitter.com/jlleitschuh (https://twitter.com/jlleitschuh))

Similar Vulnerabilities:

- Jetty - https://github.com/eclipse/jetty.project/security/advisories/GHSA-g3wg-6mcf-8jj6 (https://github.com/eclipse/jetty.project/security/advisories/GHSA-g3wg-6mcf-8jj6)
- JUnit4 - https://github.com/junit-team/junit4/security/advisories/GHSA-269g-pwp5-87pp (https://github.com/junit-team/junit4/security/advisories/GHSA-269g-pwp5-87pp)
- Google Guava - https://github.com/google/guava/issues/4011 (https://github.com/google/guava/issues/4011)
- Apache Ant - https://nvd.nist.gov/vuln/detail/CVE-2020-1945 (https://nvd.nist.gov/vuln/detail/CVE-2020-1945)
- JetBrains Kotlin Compiler - https://nvd.nist.gov/vuln/detail/CVE-2020-15824 (https://nvd.nist.gov/vuln/detail/CVE-2020-15824)

References:

[1] CWE-379: Creation of Temporary File in Directory with Insecure Permissions (https://cwe.mitre.org/data/definitions/379.html (https://cwe.mitre.org/data/definitions/379.html)) [2] "File.createTempFile" should not be used to create a directory (https://rules.sonarsource.com/java/RSPEC-5445/

(https://rules.sonarsource.com/java/RSPEC-5445/)) [3] Groovy CVE list (https://groovy-lang.org/security.html (https://groovy-lang.org/security.html)) [4] https://docs.groovy-lang.org/latest/html/groovy-jdk/java/io/File.html#createTempDir((https://docs.groovy-lang.org/latest/html/groovy-jdk/java/io/File.html#createTempDir()) [5] https://docs.groovy-lang.org/latest/html/groovy-jdk/java/io/File.html#createTempDir(java.lang.String,%20java.lang.String (https://docs.groovy-lang.org/latest/html/groovy-jdk/java/io/File.html#createTempDir(java.lang.String,%20java.lang.String)) [6] related Jira issue: https://issues.apache.org/jira/browse/GROOVY-9824 (https://issues.apache.org/jira/browse/GROOVY-9824)

Reporting problems

The Apache Software Foundation takes a very active stance in eliminating security problems in its software products. If you have questions about how to configure or use Groovy securely, you should send them to the users mailing list (mailing-lists.html). If you find any security problems due to bugs in Groovy software, you should raise issues in the bug tracker (contribute.html#reporting-issues). The Apache Software Foundation has a dedicated security team (http://www.apache.org/security/) which you may contact should the need arise.

Report issues in Jira (reporting-issues.html)
Stack Overflow questions (http://stackoverflow.com/questions/tagged/groovy)
Slack Community (http://www.groovycommunity.com/)

The Groovy programming language is supported by the Apache Software Foundation (https://www.apache.org) and the Groovy community.



Apache, Apache Groovy, Groovy, and the ASF logo are either registered trademarks or trademarks of The Apache Software Foundation.

© 2003-2026 The Apache Software Foundation — Groovy is Open Source: Apache 2 (https://www.apache.org/licenses/LICENSE-2.0.html) License (https://www.apache.org/licenses/), privacy policy (https://privacy.apache.org/policies/privacy-policy-public.html).