Java 7 Applet 0day Exploit

Gondvv (CVE-2012-4681) a.k.a Java Facepalm

Marc Adam

CS558: Internet Security
Boston University

Feb 19, 2013

1. Overview
   - Attacks
   - Java Applets
2. Technical Details
   - Java Classes Overview
   - SunToolkit getField Method
   - Vulnerabilities
   - Exploit
3. Mitigation and Patch

Overview: Attack

Timeline:
August 14, 2012: Oracle released Update 6;
August 26, 2012: Fireeye found exploit on domain in Taiwan;
August 27, 2012: Jduck publicly released Gondvv 0day Exploit;
August 30, 2012: Oracle released Update 7;
January 13, 2013: Mashable writes article about Java Exploit;

Ever since, Exploit is sold in Black Market for over 5000$.

Overview: Java Applets Security

Java Applet: Application loaded by the browser, but executed by the JVM as different processes.

Java Applets have limited security compared to normal applications
Some packages cannot be loaded to the JVM for security purposes

SecurityManager manages all the security in any Java application
It looks at the call stack to determine privileges:

1. Look at the whole call stack
2. Look at the immediate caller in the call stack

Technical Details: Java Classes Overview

- Statement is the Superclass for all Java methods/functions;
- Statement is executed within a ProtectionDomain
  - private AccessControlContext
  - public getContext()
- In a Java Applet, getContext() returns a limited ProtectionDomain ⟹ limited permissions
- If I can modify AccessControlContext, I escalate privileges!

SunToolkit getField Method

SunToolkit: Abstract Window Toolkit used by Java desktop apps;
SunToolkit has getField Method

getField method can get any private Field
getField() also calls setAccessible(true) on Field, making it public

$\Longrightarrow$ Use getField to modify private AccessControlContext!

But!

SunToolkit is restricted, it isn't allowed in Java Applets

So How Do I load it?

2 bugs: findClass & methodFinder

findClass() calls Class.forName() that loads the class
methodFinder() calls getMethod() that loads methods from a class

Belong to java.lang.ClassLoader & com.beans.finder are JRE core
JRE core class $\implies$ trusted.

Create Expression: Class.forName(”SunToolkit”)
Execute Expression $\implies$ Get call stack:

Expression.execute(...)

Statement.invokeInternal(...)

...

...

findClass(...) ← JRE trusted function

Class.forName(”SunToolkit”) ← does weak security check

Full Exploit

Gondvv Exploit:

- Create Statement instance for System.setSecurityManager(null) method using reflection
- Create a custom AccessControlContext with full permissions
- With first bug, get reference to sun.awt.SunToolkit class
- With second bug, invoke getField public static method on sun.awt.SunToolkit
- use getField() to set ACC for create Statement (step 1)
- Execute Statement

Congrats, you've disabled SecurityManager on the Java Applet!

Patch

- This exploit allows any attacker to run arbitrary code, and would allow them to do anything that Java can do
- Java vulnerable since october 2011
- Exploit code is online on: https://compilr.com/liuyuer/20124681/Gondvv.java

- Oracle released update after 3 days of exploit going public (Oracle usually has a 2-3 months patch cycle)
  - Removed getField and MethodFinder functions
  - Now checks package access based on AppletSecurity
- If you still haven't, patch Java to update 11!

References

- Exploit Explanation: http://immunityproducts.blogspot.com/2012/08/java-0day-analysis-cve-2012-4681.html
- Java Security: http://security.stackexchange.com/questions/19565/why-do-some-java-apis-bypass-standard-securitymanager-checks
- Release Notes: http://www.oracle.com/technetwork/java/javase/7u7-relnotes-1835816.html
- Security Alert: http://www.oracle.com/technetwork/topics/security/alert-cve-2012-4681-1835715.html
- Forbes Article Oracle fixing bugs: http://www.forbes.com/sites/andygreenberg/2012/08/30/oracle-quietly-releases-fix-for-serious-java-security-bug-months-after-it-was-reported/
- Blog article on Exploit: http://thexploit.com/sec/java-facepalm/
- Fix: http://immunityproducts.blogspot.com/2012/08/java-patched-at-least-4-bugs.html
- Mashable Article: http://mashable.com/2013/01/13/java-exploit/

References

- Java Security Guidelines:
  http://www.oracle.com/technetwork/java/seccodeguide-139067.html
- Statement Class:
  http://docs.oracle.com/javase/1.4.2/docs/api/java/sql/Statement.html
- Expression Class:
  http://docs.oracle.com/javase/1.4.2/docs/api/java/beans/Expression.html

Questions?