Security Vulnerability Notice

SE-2014-02-GOOGLE-5

[Google App Engine Java security sandbox bypasses, Issue 40]

DISCLAIMER

INFORMATION PROVIDED IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW NEITHER SECURITY EXPLORATIONS, ITS LICENSORS OR AFFILIATES, NOR THE COPYRIGHT HOLDERS MAKE ANY REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OR THAT THE INFORMATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS, OR OTHER RIGHTS. THERE IS NO WARRANTY BY SECURITY EXPLORATIONS OR BY ANY OTHER PARTY THAT THE INFORMATION CONTAINED IN THE THIS DOCUMENT WILL MEET YOUR REQUIREMENTS OR THAT IT WILL BE ERROR-FREE.

YOU ASSUME ALL RESPONSIBILITY AND RISK FOR THE SELECTION AND USE OF THE INFORMATION TO ACHIEVE YOUR INTENDED RESULTS AND FOR THE INSTALLATION, USE, AND RESULTS OBTAINED FROM IT.

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL SECURITY EXPLORATIONS, ITS EMPLOYEES OR LICENSORS OR AFFILIATES BE LIABLE FOR ANY LOST PROFITS, REVENUE, SALES, DATA, OR COSTS OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, PROPERTY DAMAGE, PERSONAL INJURY, INTERRUPTION OF BUSINESS, LOSS OF BUSINESS INFORMATION, OR FOR ANY SPECIAL, DIRECT, INDIRECT, INCIDENTAL, ECONOMIC, COVER, PUNITIVE, SPECIAL, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND WHETHER ARISING UNDER CONTRACT, TORT, NEGLIGENCE, OR OTHER THEORY OF LIABILITY ARISING OUT OF THE USE OF OR INABILITY TO USE THE INFORMATION CONTAINED IN THIS DOCUMENT, EVEN IF SECURITY EXPLORATIONS OR ITS LICENSORS OR AFFILIATES ARE ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS.

Security Explorations discovered one additional security vulnerability in Google App Engine for Java. A table below, presents its technical summary:

ISSUE #	TECHNICAL DETAILS	
40	Origin	Class Sweeper
	Cause	no security checks related to class linking and methods resolution
	Impact	the ability to subclass and call methods of restricted classes
	Type	partial GAE security bypass vulnerability

Issue 40 stems from the fact that no security checks are implemented in GAE that would correspond to the JRE security checks aimed at prohibiting access to restricted classes[1]. GAE implements additional restricted classes namespace on top of the JRE, but it does not implement security checks in all locations where such classes could be referenced. More specifically, it does not implement the necessary security checks related to the class linking and methods resolution. As a result, user defined classes could be linked with restricted GAE classes (they could subclass from them and call their methods via `invokevirtual` / `invokespecial` / `invokestatic` bytecode instructions).

Issue 40 could be combined together with a previously reported Issue 37 to achieve a complete GAE Java security sandbox escape. The following exploitation scenario is implemented in our Proof of Concept code (POC30) to illustrate that:

1) Issue 37 is used to create an instance of a `java.net.URLClassLoader` class (UCL loader),
2) UCL loader is used to create an instance of `MyCPU` class, which is a subclass of `com.google.apphosting.runtime.ClassPathUtils` class, the instantiation process proceeds through a finalizer due to the abnormal termination of a `ClassPathUtils` constructor (*SecurityException* gets thrown by the `addApiJars` method),
3) a custom `MyCL` Class Loader instance is created that delegates loading (linking) of restricted GAE classes to UCL loader,
4) Issue 40 is used to create an instance of `MyRCL` class, which is a subclass of `com.google.apphosting.runtime.security.RuntimeClassLoader` class, `MyRCL` class definition and instantiation occurs in the `MyCL` Class Loader namespace with a Class Sweeper in place,
5) a privileged `HelperClas` class is defined in `MyRCL` Class Loader namespace,
6) `HelperClass` class is instantiated and a Security Manager is turned off.

Attached to this report, there is a Proof of Concept code that illustrates the impact of the vulnerability described above. It has been successfully tested in a production GAE environment patched against security issues we reported to Google in Dec 2014 / Jan 2015.

About Security Explorations

[1] such as classes originating from a non-user Class Loader namespace or that are not on the JRE Class Whitelist.

Security Explorations (http://www.security-explorations.com) is a security start-up company from Poland, providing various services in the area of security and vulnerability research. The company came to life in a result of a true passion of its founder for breaking security of things and analyzing software for security defects. Adam Gowdiak is the company's founder and its CEO. Adam is an experienced Java Virtual Machine hacker, with over 50 security issues uncovered in the Java technology over the recent years. He is also the hacking contest co-winner and the man who has put Microsoft Windows to its knees (vide MS03-026). He was also the first one to present successful and widespread attack against mobile Java platform in 2004.