Security Vulnerability Notice

SE-2012-01-ORACLE-13

[Security vulnerabilities in Java SE, Issue 69]

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 a security vulnerability in Java SE Platform, Standard Edition. The vulnerability is due to insecure implementation of new Reflection API at the core VM level. More specifically, we found out that Class Loader constraints are not enforced for Method Handle objects. This is illustrated on a picture below:

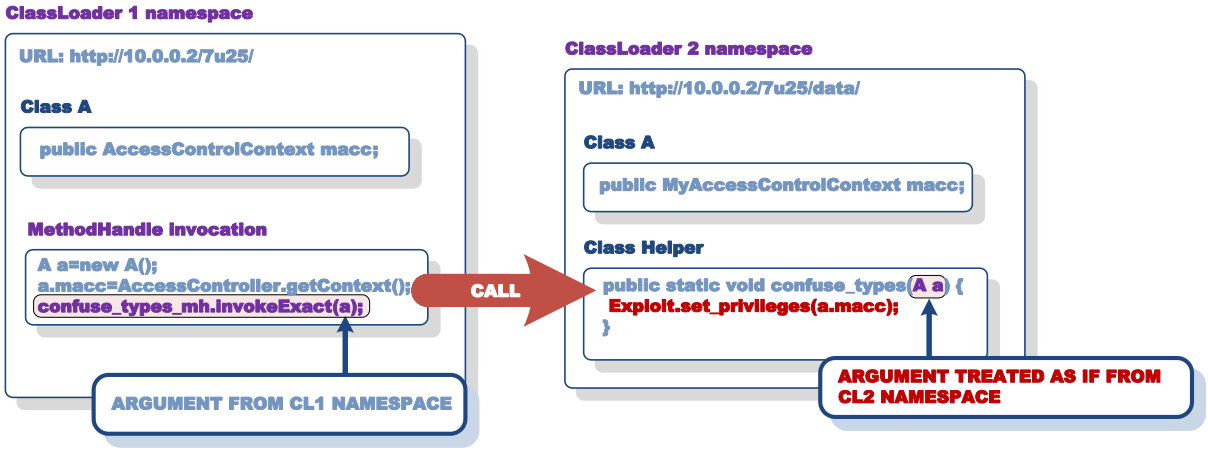

Figure 1 Illustration of the vulnerability.

When Method Handle objects are invoked across two different Class Loader namespaces, no checks are done against the type safety of their argument types. As a result, it is possible to provide a spoofed definition for a given argument type, which will be treated as of a completely different type in a target Class Loader namespace.

In our Proof of Concept code, we create and call a Method Handle instance pointing to the static `confuse_types` method from a custom Class Loader namespace (CL2). As an argument we pass an instance of Class A from our default Class Loader namespace (CL1). Since, no Class Loader constraints are enforced for MethodHandle calls conducted through the new Reflection API, it is possible to spoof the type of an argument provided to `confuse_types` method (Class A). This itself directly leads to a type confusion condition.

It should be possible to conduct similar attack with the use of Method Handle objects denoting fields as well. This however requires more thorough investigation.

What's interesting to note is that the described class spoofing attack had been known for at least 10+ years. Its technical details can be found in a Java VM security paper from 2002 [1]. The more surprising it is to discover that new Reflection API introduced to Java SE 7 didn't implement proper protection against this very classic attack against Java VM.

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 the environment of Java SE 7 Update 25 (JRE version 1.7.0_25-b16) with Internet Explorer 9 and Mozilla Firefox 20.0.1 web browsers.

REFERENCES

[1] Java and Java VM security vulnerabilities and their exploitation techniques, Last Stage of Delirium Research Group, `http://lsd-pl.net/`

About Security Explorations

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.