



Workshop: Advanced Java Application Code Review

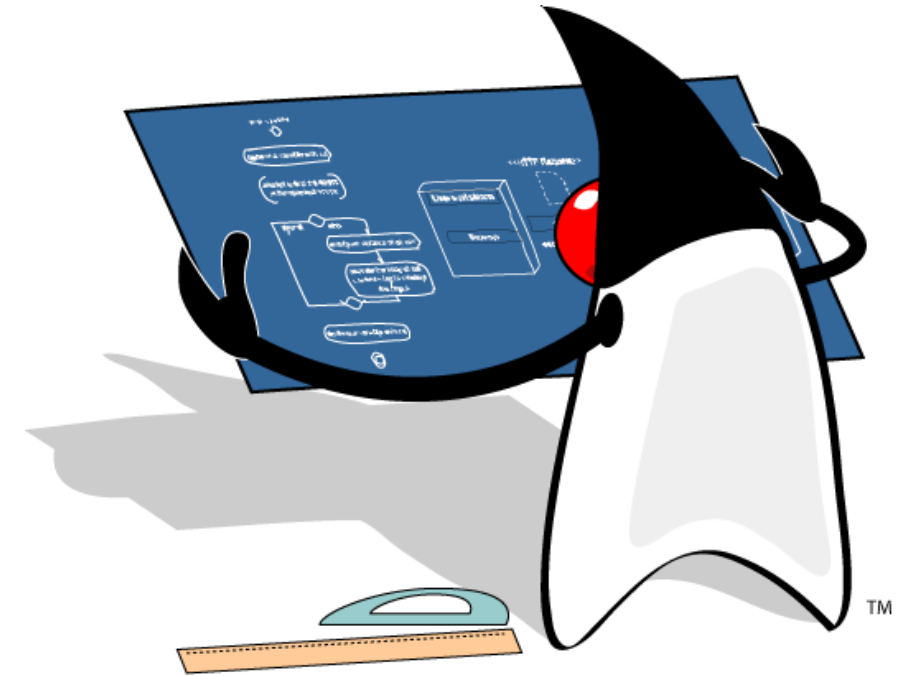
Philippe Arteau
Security Researcher at GoSecure

44CON

14/09/2016

What is this Training About?

- Agenda (2 hours training)
 - Introduction to SonarQube
 - **Code review** exercises divided in 6 modules
 - Some **exploitation**..

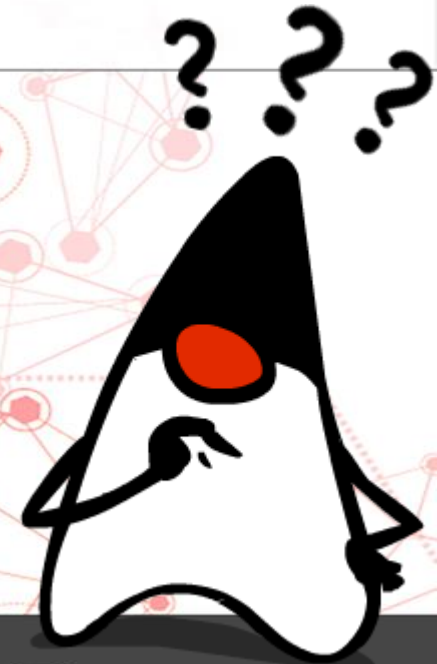


Agenda

- Warm up (Path traversal, Cryptography, XSS)
- Xml eXternal Entity (XXE)
- Template injection
- Deserialisation vulnerabilities
- Expression Language (EL) injection
- Hibernate Query Language injection

Module 1

Warm up



GoSECURE

SonarQube interface

44CON

Advanced Java Application Code Review
Trainer: [Philippe Arteau \(GoSecure\)](#)

Welcome to this code review workshop !

Warm-up

- [Crypto \[Open\]](#)
- [XSS \[Open\]](#)
- [Forgot password: Logic flaw \[Open\]](#)

Modules

- [RssViewer \[Open\]](#)
- [EmailCampaign \[Open\]](#)
- [RemoteAdmin \[Open\]](#)

Reference material
All the references are here -->
• [Workshop CheatSheet \(Slides\)](#)

PROJECTS

QG	NAME	VERSION	LOC	BUGS	VUL
★	✓ EmailCamp	0.0.1-SNAPSHOT	221	0	
★	✓ RemoteAdmin	0.0.1-SNAPSHOT	76	0	
★	✓ RssViewer	0.0.1-SNAPSHOT	120	0	
★	✓ VulnerableApp	0.0.1-SNAPSHOT	90	0	

4 results

PROJECTS

Vulnerabilities

VulnerableApp	11
RemoteAdmin	2
RssViewer	1
EmailCamp	0

VULNERABLEAPP

Issue Tag Cloud

Hotspots by Vulnerabilities [More](#)

CryptoFunctions.java	4
xss5.jsp	3
ResetPassword.java	1
xss0.jsp	1
xss1.jsp	1
xss2.jsp	1

SonarQube™ technology is powered by SonarSource SA
Version 5.6.1 - [LGPL v3](#) - [Community](#) - [Documentation](#) - [Get Support](#) - [Plugins](#) - [Web API](#)

Quick
links

Applications
analyzed

SonarQube interface

September 8, 2016 9:08 PM Version 0.0.1-SNAPSHOT

VulnerableApp

Issues Measures Code Dashboards

19 * @param password Value to hash
20 * @return Hash of the password
21 * @throws NoSuchAlgorithmException
22 * @throws UnsupportedEncodingException
23 */
24 public static String hashPassword(String password) throws NoSuchAlgorithmException, UnsupportedEncodingException {
25
26 MessageDigest md = MessageDigest.getInstance("MD5");
27
28 MD5 is not a recommended cryptographic hash function ---
29 Vulnerability Major Open Not assigned Comment cryptography, cwe, owasp
30
31 byte[] hash = md.digest(password.getBytes("UTF-8"));
32
33 StringBuilder sb = new StringBuilder(2*hash.length);
34 for(byte b : hash) {
35 sb.append(String.format("%02x", b & 0xff));
36 }
37 return sb.toString();
38 }
39 }

Open the short description

Open the detailed explanation and resolution

Security - MD2, MD4 and MD5 are weak hash functions

Vulnerability Major cryptography, cwe, owasp-a6 Available Since September 8, 2016 findseccbugs:WEAK_MESSAGE_DIGEST_MD5

The algorithms MD2, MD4 and MD5 are not a recommended MessageDigest. **PBKDF2** should be use to hash password for example.

"The security of the MD5 hash function is severely compromised. A collision attack exists that can find collisions within seconds on a computer with a 2.6 GHz Pentium 4 processor (complexity of $2^{24.1}$).[1] Further, there is also a chosen-prefix collision attack that can produce a collision for two inputs with specified prefixes within hours, using off-the-shelf computing hardware (complexity 2^{39}).[2]"

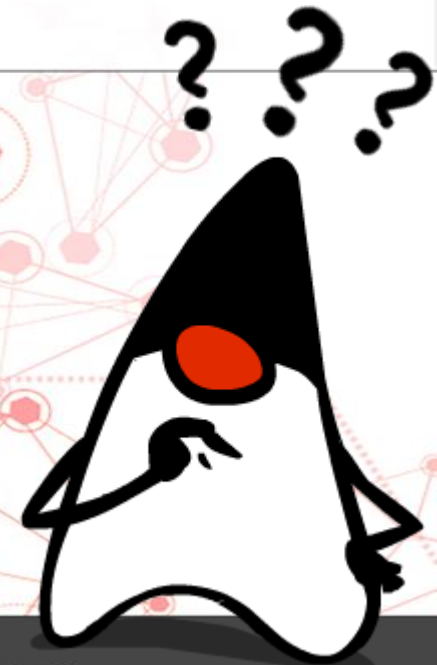
- [Wikipedia: MD5 - Security](#)

"SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, and SHA-512/256:
The use of these hash functions is acceptable for all hash function applications."

- [NIST: Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths p.15](#)

Module 2

Xml eXternal Entity (XXE)



GoSECURE

Xml eXternal Entity (XXE)

Example of simple payload and out-of-bound:

- <http://blog.h3xstream.com/2014/06/identifying-xml-external-entity.html>

Example with jar: protocol trick

- http://www.agarri.fr/kom/archives/2013/11/27/compromising_an_unreachable_solr_server_with_cve-2013-6397/index.html

XML Schema, DTD, and Entity Attacks

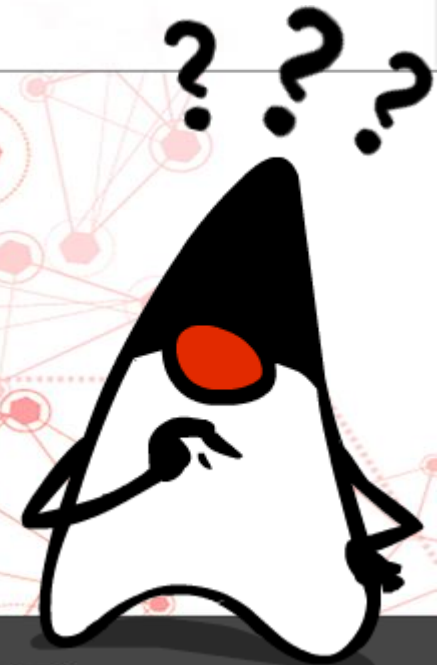
- <http://vsecurity.com/download/papers/XMLDTDEntityAttacks.pdf>

More examples

- <http://web-in-security.blogspot.com/2016/03/xxe-cheat-sheet.html>

Module 3

Template Injection



GoSECURE

Template Injection

Server-Side Template Injection

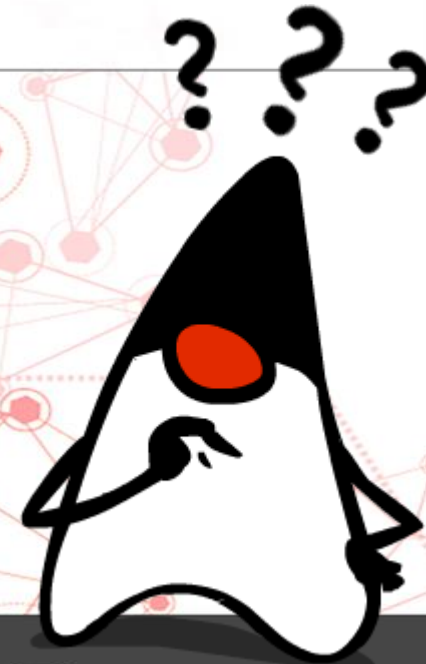
- <http://blog.portswigger.net/2015/08/server-side-template-injection.html>

Server-Side Template Injection (White-paper version)

- <https://www.blackhat.com/docs/us-15/materials/us-15-Kettle-Server-Side-Template-Injection-RCE-For-The-Modern-Web-App-wp.pdf>

Module 4

Deserialization vulnerabilities



GoSECURE

Deserialization Vulnerabilities

Proof of concept for WebLogic, WebSphere, JBoss, ...

- <https://foxglovesecurity.com/2015/11/06/what-do-weblogic-websphere-jboss-jenkins-opennms-and-your-application-have-in-common-this-vulnerability/>

AppSecCali : first presentation on the subject with a working gadget demo

- <http://frohoff.github.io/appseccali-marshalling-pickles/>

Great article to understand the risk and differentiate safe and unsafe implementations

- <http://www.ibm.com/developerworks/library/se-lookahead/>

YSoSerial (tool needed for the exercise)

- <https://github.com/frohoff/ysoserial>

Deserialization Vulnerabilities

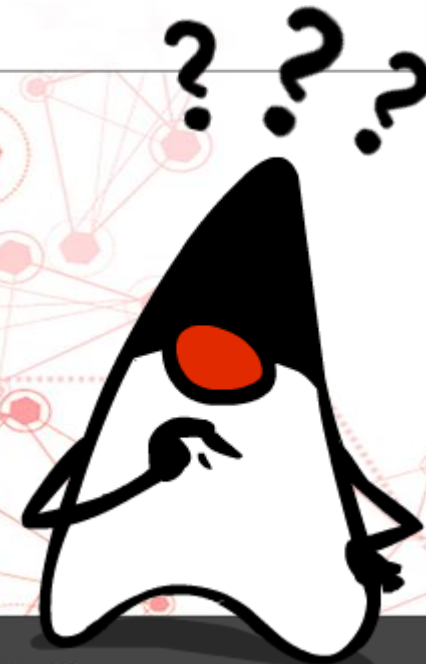
YSoSerial

- List the gadgets available
 - `java -jar ysoserial-0.0.5-all.jar`
- Generate a payload
 - `java -jar ysoserial-0.0.5-all.jar [GADGET_NAME] "YOUR_COMMAND" > payload.bin`



Module 5

Expression Language (EL) injection



GoSECURE

Expression Language Injection

Spring Expression Language (SpEL) and ScriptEngine/Rhino examples

- <http://blog.h3xstream.com/2014/11/remote-code-execution-by-design.html>

Java EL Injection (Mostly used in JSP)

- <http://sectooladdict.blogspot.com/2014/12/el-30-injection-java-is-getting-hacker.html>

Java EL Injection (Mostly used in JSP)

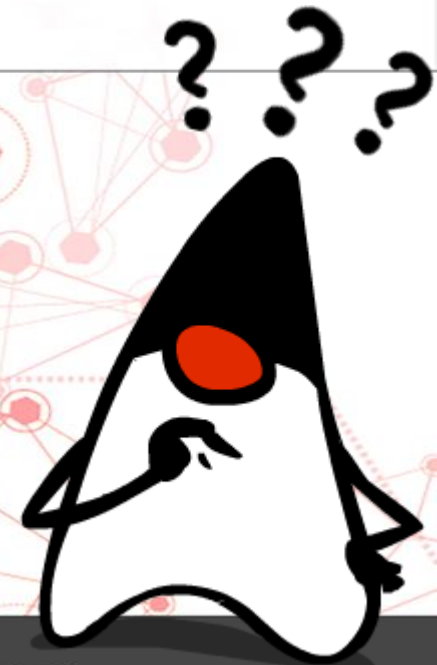
- <http://blog.mindedsecurity.com/2015/11/reliable-os-shell-with-el-expression.html>

Spring Expression Language (SpEL)

- <http://danamodio.com/appsec/research/spring-remote-code-with-expression-language-injection/>

Module 6

Hibernate Query Language (HQL)



Hibernate Query Language

Introduction to HQL

- <http://blog.h3xstream.com/2014/02/hql-for-pentesters.html>

Reference for “advanced” exploitation (Zeronights 2015)

- <http://www.slideshare.net/0ang3el/orp2pwn-exploiting-injections-in-hibernate-orm>

HQLmap: tool for automation

- <https://github.com/PaulSec/HQLmap>



Questions?

Contact

✉ parteau@gosecure.ca

🌐 blog.gosecure.ca

🐦 [@h3xstream](https://twitter.com/h3xstream)