



11. – 14.12.2017  
Frankfurt am Main

Dominik Schadow

# Sicherheit beim Build

#ittage

Camera 009

Camera 011

Camera 051

Camera 071

Camera 039

Camera 046

Camera 055



A woman with blonde hair tied back is shown from behind, wearing a dark blue button-down shirt. She is standing in front of a grid of nine surveillance camera feeds. The cameras are labeled with IDs: Camera 009 (top left), Camera 011 (top middle), Camera 051 (top right), Camera 071 (middle left), Camera 039 (middle right), Camera 046 (bottom left), and Camera 055 (bottom right). The background shows office cubicles and desks. The overall aesthetic is that of a security monitoring interface.

Verify your  
security activities

Build chain integration

Security issues discovered as early as possible

Static and dynamic application security testing

# What to look for?



# OWASP Top 10 Proactive Controls 2016

C01 Verify for Security Early and Often

C02 Parameterize Queries

C03 Encode Data

C04 Validate All Inputs

C05 Implement Identity and Authentication Controls

C06 Implement Appropriate Access Controls

C07 Protect Data

C08 Implement Logging and Intrusion Detection

C09 Leverage Security Frameworks and Libraries

C10 Error and Exception Handling

# OWASP Top 10 Proactive Controls 2016

## C01 Verify for Security Early and Often

C02 Parameterize Queries

C03 Encode Data

C04 Validate All Inputs

C05 Implement Identity and Authentication Controls

C06 Implement Appropriate Access Controls

C07 Protect Data

C08 Implement Logging and Intrusion Detection

C09 Leverage Security Frameworks and Libraries

C10 Error and Exception Handling

A close-up photograph of a person's hand reaching towards a branch of cherries. The hand is positioned to grab a dark red cherry from a cluster. The background is filled with green leaves, some of which appear yellowed or autumnal. The lighting is warm, suggesting sunlight filtering through the leaves.

Catch the low hanging  
fruits (a.k.a. *the easy stuff*)



# Jenkins

# sonarqube

OWASP ZAP

OWASP Dependency Check

FindBugs / Find Security Bugs (SpotBugs)

# Scan for vulnerabilities in web applications



Open source web proxy and dynamic  
application security testing tool

[Updates](#)**Available**[Installed](#)[Advanced](#)**Install** ↓**Name****Version**[Official OWASP ZAP Jenkins Plugin](#)

The Official OWASP ZAP Jenkins Plugin extends the functionality of the ZAP security tool into a CI Environment.

1.1.0

[Install without restart](#)[Download now and install after restart](#)



Scan your application  
for anything you like

# Demo

# Tips & Tricks

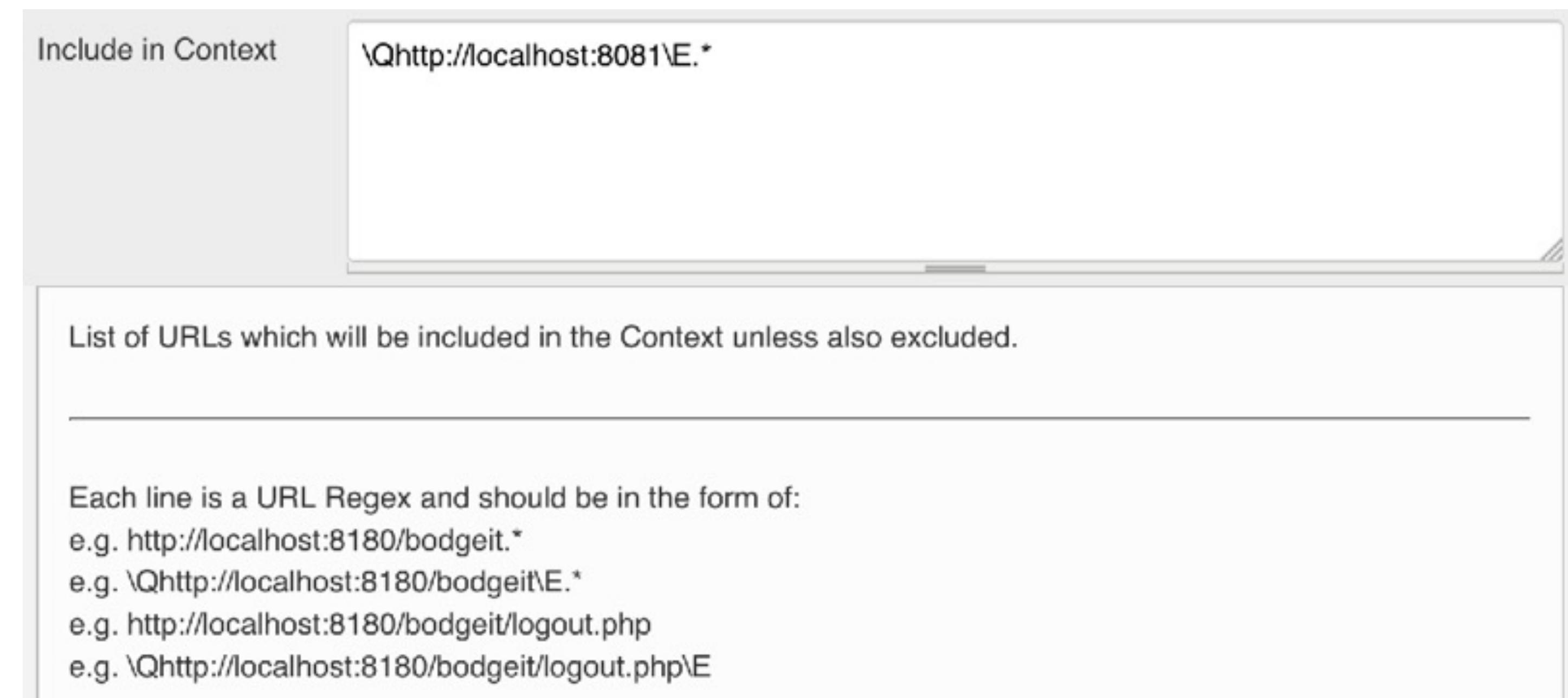
- ❑ Always start with a clean application (and a test database)
- ❑ Launch web application with extensive logging the first time  
(e.g. via Spring Boot profile)

- ❑ Context configuration

sometimes tricky,

`\Q[URL]\E.*` often the

best solution



A photograph of a large, dense hedge maze. The hedges are dark green and well-maintained, forming intricate, winding paths. A narrow, light-colored dirt path leads through the center of the maze, curving to the right. The lighting suggests a sunny day with shadows cast by the hedges.

Spider clicks on any link\*

*\* Including the ,delete all‘ link to wipe all db entries*

# OWASP Top 10 Proactive Controls 2016

**C01 Verify for Security Early and Often**

**C02 Parameterize Queries**

**C03 Encode Data**

**C04 Validate All Inputs**

**C05 Implement Identity and Authentication Controls**

**C06 Implement Appropriate Access Controls**

**C07 Protect Data**

**C08 Implement Logging and Intrusion Detection**

**C09 Leverage Security Frameworks and Libraries**

**C10 Error and Exception Handling**

Works on my machine

Standard Mode

Sites

Header: Text Body: Text

```
POST http://localhost:8080/logout HTTP/1.1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.12; rv:53.0) Gecko/20100101 Firefox/53.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Content-Type: application/x-www-form-urlencoded
Content-Length: 42
Referer: http://localhost:8080/user/user
Cookie: 1SESS10NTD=8F66EFC476B3FA09AF7693362C5A611F5
_csrf=cb2bb499-ebcd-49bf-ae93-49b1996eebdd
```

History Search Alerts Output

Filter: OFF

ID	Req. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Body	Highest Alert	Note	Tags
37	09/06/17 11:01:12	GET	http://localhost:8080/	200		12 ms	994 bytes			SetCookie
38	09/06/17 11:01:12	GET	http://localhost:8080/webjars/bootstrap/css/boo...	200		7 ms	121,200 bytes			Upload, Comment
39	09/06/17 11:01:13	GET	http://localhost:8080/admin/admin	302		6 ms	0 bytes			
40	09/06/17 11:01:13	GET	http://localhost:8080/login	200		6 ms	535 bytes	Low		Form, Password, Hidd...
41	09/06/17 11:01:17	POST	http://localhost:8080/login	302		114 ms	0 bytes			SetCookie
42	09/06/17 11:01:17	GET	http://localhost:8080/admin/admin	200		14 ms	1,285 bytes			Form, Hidden
43	09/06/17 11:01:17	GET	http://localhost:8080/webjars/bootstrap/css/boo...	200		7 ms	121,200 bytes			Upload, Comment
44	09/06/17 11:01:20	POST	http://localhost:8080/logout	302		5 ms	0 bytes			
45	09/06/17 11:01:20	GET	http://localhost:8080/	200		12 ms	994 bytes			SetCookie
46	09/06/17 11:01:20	GET	http://localhost:8080/webjars/bootstrap/css/boo...	200		7 ms	121,200 bytes			Upload, Comment
47	09/06/17 11:18:34	GET	http://detectportal.firefox.com/success.txt	200 OK		1.38 s	8 bytes	Low		

# Identify libraries with known vulnerabilities



# OWASP Top 10 2017

## A9 – Using Components with Known Vulnerabilities

Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.

[Updates](#)**Available**[Installed](#)[Advanced](#)**Install** ↓**Name****Version**[OWASP Dependency-Check Plugin](#)

Dependency-Check is a utility that identifies project dependencies and checks if there are any known, publicly disclosed, vulnerabilities. This tool can be part of the solution to the OWASP Top 10 2013: A9 - Using Components with Known Vulnerabilities. This plug-in can independently execute a Dependency-Check analysis and visualize results.

2.1.1

[Install without restart](#)[Download now and install after restart](#)

# Demo

```
<?xml version="1.0" encoding="UTF-8"?>
<suppressions xmlns="https://jeremylong.github.io/DependencyCheck/dependency-suppression.1.1.xsd">
    <!-- this is actually one of the project artifacts and not a dependency -->
    <suppress>
        <notes><![CDATA[
file name: sso-with-github.jar
]]></notes>
        <filePath regex="true">.*\bsso-with-github\.jar</filePath>
        <cve>CVE-2010-2542</cve>
    </suppress>
    <!-- no SVG usage -->
    <suppress>
        <notes><![CDATA[
file name: batik-css-1.8.jar
]]></notes>
        <gav regex="true">^org\apache\xmlgraphics:batik-css:.*$</gav>
        <cpe>cpe:/a:apache:batik</cpe>
    </suppress>
    <suppress>
        <notes><![CDATA[
file name: batik-ext-1.8.jar
]]></notes>
        <gav regex="true">^org\apache\xmlgraphics:batik-ext:.*$</gav>
        <cpe>cpe:/a:apache:batik</cpe>
    </suppress>
</suppressions>
```

# OWASP Top 10 Proactive Controls 2016

## C01 Verify for Security Early and Often

C02 Parameterize Queries

C03 Encode Data

C04 Validate All Inputs

C05 Implement Identity and Authentication Controls

C06 Implement Appropriate Access Controls

C07 Protect Data

C08 Implement Logging and Intrusion Detection

C09 Leverage Security Frameworks and Libraries

C10 Error and Exception Handling

Works on my machine

```
$ dependency-check --project JavaSecurity --scan ./**/*jar --suppression dependency-check-suppressions.xml
[INFO] Checking for updates
[INFO] Skipping NVD check since last check was within 4 hours.
[INFO] Check for updates complete (1893 ms)
[INFO] Analysis Started
[INFO] Finished Archive Analyzer (1 seconds)
[INFO] Finished File Name Analyzer (0 seconds)
[INFO] Finished Jar Analyzer (1 seconds)
[INFO] Finished Central Analyzer (8 seconds)
[INFO] Finished Dependency Merging Analyzer (0 seconds)
[INFO] Finished Version Filter Analyzer (0 seconds)
[INFO] Finished Hint Analyzer (0 seconds)
[INFO] Created CPE Index (1 seconds)
[INFO] Finished CPE Analyzer (3 seconds)
[INFO] Finished False Positive Analyzer (0 seconds)
[INFO] Finished Cpe Suppression Analyzer (0 seconds)
[INFO] Finished NVD CVE Analyzer (1 seconds)
[INFO] Finished Vulnerability Suppression Analyzer (0 seconds)
[INFO] Finished Dependency Bundling Analyzer (0 seconds)
[INFO] Analysis Complete (16 seconds)
```



# DEPENDENCY-CHECK

Dependency-Check is an open source tool performing a best effort analysis of 3rd party dependencies; false positives and false negatives may exist in the analysis performed by the tool. Use of the tool and the reporting provided constitutes acceptance for use in an AS IS condition, and there are NO warranties, implied or otherwise, with regard to the analysis or its use. Any use of the tool and the reporting provided is at the user's risk. In no event shall the copyright holder or OWASP be held liable for any damages whatsoever arising out of or in connection with the use of this tool, the analysis performed, or the resulting report.

[How to read the report](#) | [Suppressing false positives](#) | [Getting Help: google group](#) | [github issues](#)

## Project: Java Security

Scan Information ([show all](#)):

- dependency-check version: 3.0.1
- Report Generated On: Oct 20, 2017 at 21:32:45 +02:00
- Dependencies Scanned: 121 (96 unique)
- Vulnerable Dependencies: 7
- Vulnerabilities Found: 10
- Vulnerabilities Suppressed: 4
- ...

Display: [Showing Vulnerable Dependencies \(click to show all\)](#)

Dependency	CPE	GAV	Highest Severity	CVE Count	CPE Confidence	Evidence Count
<a href="#">commons-beanutils-core-1.8.3.jar</a>	cpe:/a:apache:commons_beanutils:1.8.3	<a href="#">commons-beanutils:commons-beanutils-core:1.8.3</a> ✓	High	1	Low	25
<a href="#">commons-fileupload-1.3.1.jar</a>	<a href="#">cpe:/a:apache:commons_fileupload:1.3.1</a>	<a href="#">commons-fileupload:commons-fileupload:1.3.1</a> ✓	High	2	Highest	36
<a href="#">ognl-3.0.8.jar</a>	cpe:/a:ognl_project:ognl:3.0.8	<a href="#">ognl:ognl:3.0.8</a> ✓	Medium	1	Low	22
<a href="#">tomcat-coyote-9.0.28</a>	cpe:/a:apache:tomcat-coyote:9.0.28	<a href="#">tomcat-coyote:tomcat-coyote:9.0.28</a>	High	2	Low	12

## Evidence

## Related Dependencies

## Identifiers

- cpe: [cpe:/a:apache:batik:1.8](#) Confidence:Highest [suppress](#)
- maven: [org.apache.xmlgraphics:batik-css:1.8](#) ✓ Confidence:Highest

## Published Vulnerabilities

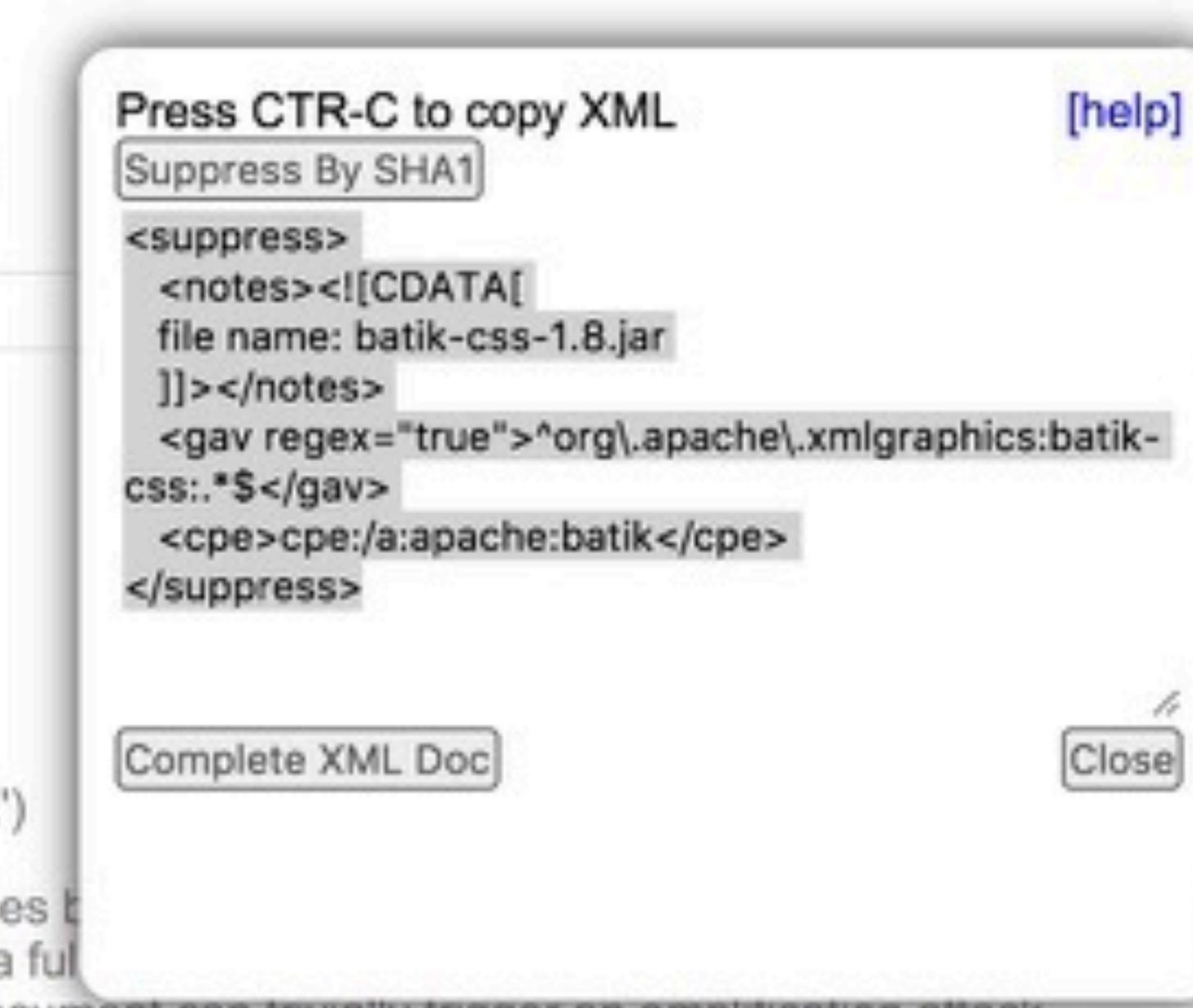
### [CVE-2017-5662](#) [suppress](#)

Severity: High

CVSS Score: 7.9 (AV:N/AC:M/Au:S/C:C/I:N/A:C)

CWE: CWE-611 Improper Restriction of XML External Entity Reference ('XXE')

In Apache Batik before 1.9, files lying on the filesystem of the server which uses the user context in which the exploitable application is running. If the user is root a full availability of the server via denial of service as the references within a xml document can trivially trigger an amplification attack.



## Vulnerable Software & Versions:

- cpe:/a:apache:batik:1.8 and all previous versions

# Find vulnerabilities in source code



[Updates](#)**Available**[Installed](#)[Advanced](#)**Install ↓****Name****Version**[FindBugs Plug-in](#)

This plug-in generates the trend report for FindBugs, an open source program which uses static analysis to look for bugs in Java code.

4.71

[Install without restart](#)[Download now and install after restart](#)

## Build

Root POM

pom.xml



Goals and options

clean package findbugs:findbugs



[Advanced...](#)

## Build Settings

Publish Checkstyle analysis results



Publish FindBugs analysis results



Use rank as priority



Uses the bug rank when evaluating the priority of the warnings (otherwise the FindBugs priority is used).

[Advanced...](#)

# findbugs-security-include.xml

```
<FindBugsFilter>
  <Match>
    <Bug category="SECURITY"/>
  </Match>
</FindBugsFilter>
```

# FindBugs Result

## Warnings Trend

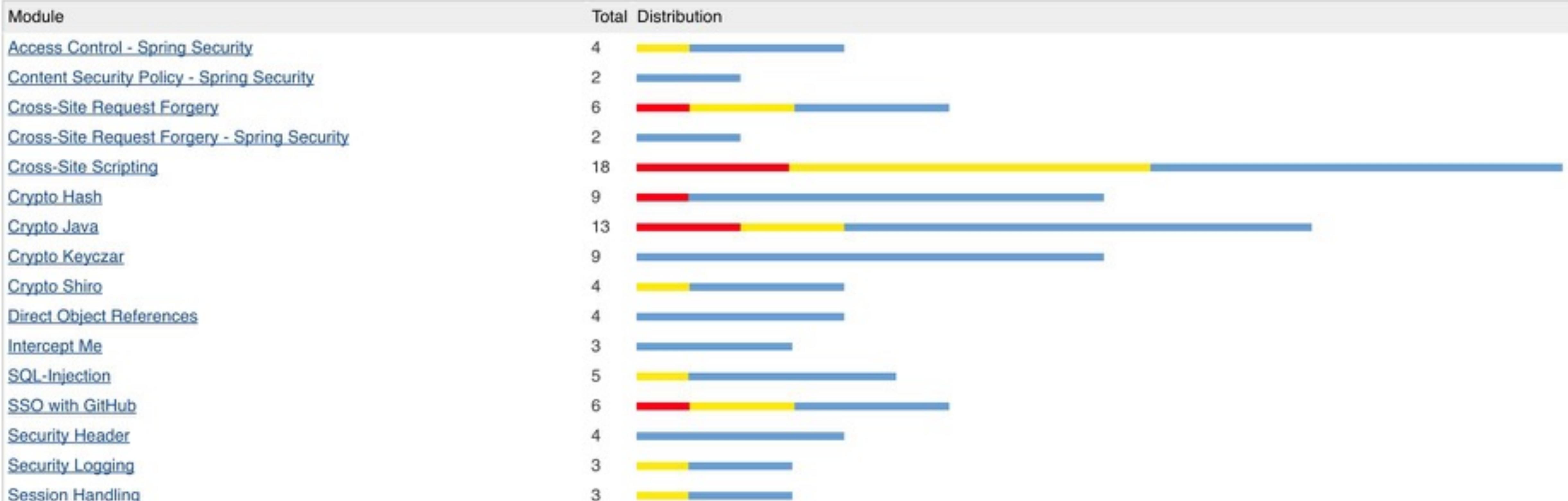
All Warnings	New this build	Fixed Warnings
98	<a href="#">98</a>	0

## Summary

Total	High Priority	Normal Priority	Low Priority
98	<a href="#">8</a>	<a href="#">18</a>	<a href="#">72</a>

## Details

Modules	Packages	Files	Types	Warnings	Details	New	High	Normal	Low



A blurred background image of two relay runners. One runner in red shorts and a white shirt is handing off a yellow baton to another runner in similar attire. The scene is set outdoors with a clear blue sky.

# FindBugs wants to be fast...

List contains false positives  
and negatives

New generation just in time code analysis by  
Fraunhofer IEM/ University Paderborn

## ★ Application Intrusion Detection

10. Juni 2017 14:39 Version 316

[Home](#) Issues Measures Code Activity Administration ▾

### Quality Profiles

Choose which profile is associated with this project on a language-by-language basis. (Note that you will only need to select profiles for multiple languages for multi-language projects.)

Language	Quality Profile
----------	-----------------

CSS	Default: SonarQube Way
-----	------------------------

Java	FindBugs Security Audit
------	-------------------------

JavaScript	Default: Sonar way
------------	--------------------

Less	Default: SonarQube Way
------	------------------------

SCSS	Default: SonarQube Way
------	------------------------

Web	Default: Sonar way
-----	--------------------

XML	Default: Sonar way
-----	--------------------

## Rules

1 / 33 rules

[Reload](#) [New Search](#) [Bulk Change](#)

<input type="text" value="Search"/>
<input checked="" type="checkbox"/> Language
Java 33
JavaScript 9
flex 6
CSS 1
Less 1
py 1
SCSS 1
<input type="text" value="Search"/>
<input checked="" type="checkbox"/> Type
Bug 128
Vulnerability 33
Code Smell 252
<input type="checkbox"/> Tag
<input type="checkbox"/> Repository
<input type="checkbox"/> Default Severity
<input type="checkbox"/> Status
<input type="checkbox"/> Available Since
<input type="checkbox"/> Template
<input checked="" type="checkbox"/> Quality Profile
FindBugs Java
FindBugs + FB-Contrib Java
FindBugs Security Audit Java
FindBugs Security Minimal Java <span style="background-color: #e0f2e0; border: 1px solid #80c0ff;">active</span> <span style="background-color: #ffcccc; border: 1px solid red;">inactive</span>
Sonar way Java
<input type="checkbox"/> Inheritance
<input type="checkbox"/> Activation Severity

"@RequestMapping" methods should be "public"	Java	Vulnerability  spring	T ▾	<a href="#">Activate</a>
"enum" fields should not be publicly mutable	Java	Vulnerability  bad-practice	T ▾	<a href="#">Activate</a>
"File.createTempFile" should not be used to create a directory	Java	Vulnerability  owasp-a9	T ▾	<a href="#">Activate</a>
"HttpServletRequest.getRequestedSessionId()" should not be used	Java	Vulnerability  cwe, owasp-a2, sans-top25-porous	T ▾	<a href="#">Activate</a>
"javax.crypto.NullCipher" should not be used for anything other than testing	Java	Vulnerability  cwe, owasp-a6, sans-top25-porous	T ▾	<a href="#">Activate</a>
"public static" fields should be constant	Java	Vulnerability  cert, cwe	T ▾	<a href="#">Activate</a>
Class variable fields should not have public accessibility	Java	Vulnerability  cwe	T ▾	<a href="#">Activate</a>
Classes should not be loaded dynamically	Java	Vulnerability  cwe, owasp-a1	T ▾	<a href="#">Activate</a>
Cookies should be "secure"	Java	Vulnerability  cwe, owasp-a2, owasp-a6	T ▾	<a href="#">Activate</a>
Credentials should not be hard-coded	Java	Vulnerability  cert, cwe, owasp-a2, sans-top25-porous	T ▾	<a href="#">Activate</a>
Cryptographic RSA algorithms should always incorporate OAEP (Optimal Asymmetric Encryption Padding)	Java	Vulnerability  cwe, owasp-a5, owasp-a6, sans-top25-porous	T ▾	<a href="#">Activate</a>
Default EJB interceptors should be declared in "ejb-jar.xml"	Java	Vulnerability	T ▾	<a href="#">Activate</a>
Exceptions should not be thrown from servlet methods	Java	Vulnerability  cert, cwe, error-handling, owasp-a6	T ▾	<a href="#">Activate</a>
HTTP referers should not be relied on	Java	Vulnerability  cwe, owasp-a2, sans-top25-porous	T ▾	<a href="#">Activate</a>
IP addresses should not be hardcoded	Java	Vulnerability  cert	T ▾	<a href="#">Activate</a>
Member variable visibility should be specified	Java	Vulnerability	T ▾	<a href="#">Activate</a>
Members of Spring components should be injected	Java	Vulnerability  spring	T ▾	<a href="#">Activate</a>
Mutable fields should not be "public static"	Java	Vulnerability  cert, cwe, unpredictable	T ▾	<a href="#">Activate</a>
Mutable members should not be stored or returned directly	Java	Vulnerability  cert, cwe, unpredictable	T ▾	<a href="#">Activate</a>

## Java Security

Issues Measures Code Activity Administration ▾

### Display Mode

Issues

Effort

ordered by creation date 1 / 4 issues

Reload New Search Bulk Change

### Type

Bug

0

Vulnerability

4

Code Smell

0

SQL-Injection / src/.../javasecurity/database/PlainSqlQuery.java

Use a variable binding mechanism to construct this query instead of concatenation. [...](#)

vor 3 Monaten ▾ L45 ⌂ ⌂ ▾

Vulnerability ▾ Critical ▾ Open ▾ Not assigned ▾ 20min effort Comment [cert, cwe, hibernate, owasp-a1, sans-top25-insecure, sql ▾](#)

### Resolution

Unresolved 4

Fixed 0

Crypto Hash / src/.../javasecurity/hash/MD5.java

Remove this hard-coded password. [...](#)

vor 2 Jahren ▾ L47 ⌂ ⌂ ▾

Vulnerability ▾ Critical ▾ Open ▾ Not assigned ▾ 30min effort Comment [cert, cwe, owasp-a2, sans-top25-porous ▾](#)

False Positive 0

Won't fix 0

Removed 7

Crypto Hash / src/.../javasecurity/hash/SHA512.java

Remove this hard-coded password. [...](#)

vor 2 Jahren ▾ L51 ⌂ ⌂ ▾

Vulnerability ▾ Critical ▾ Open ▾ Not assigned ▾ 30min effort Comment [cert, cwe, owasp-a2, sans-top25-porous ▾](#)

Severity

Status

Creation Date

Rule

Tag

Module

Directory

File

Assignee

Crypto Shiro / src/.../javasecurity/hash/SHA512.java

Remove this hard-coded password. [...](#)

vor 3 Jahren ▾ L51 ⌂ ⌂ ▾

Vulnerability ▾ Critical ▾ Open ▾ Not assigned ▾ 30min effort Comment [cert, cwe, owasp-a2, sans-top25-porous ▾](#)

## ★ Java Security

Issues Measures Code Activity Administration ▾

Display Mode

Issues

Effort

Return to List

SQL-Injection / src/.../javasecurity/database/PlainSqlQuery.java

1 / 4 issues

Reload

New Search

Bulk Change

 Type

Bug

0

Vulnerability

4

Code Smell

0

 Resolution

Unresolved 4 Fixed 0

False Positive 0 Won't fix 0

Removed 7

 Severity Status Creation Date Rule Tag Module Directory File Assignee Author Language

```

22 import org.springframework.stereotype.Component;
23
24 import java.util.List;
25 import java.util.Map;
26
27 /**
28  * Servlet using a plain Statement to query the in-memory-database. User input is not modified and used directly in the
29  * SQL query. {@code ' or '1'='1} is a good input to return all statements, {@code ';' drop table customer;--} to delete
30  * the complete table.
31 *
32 * @author Dominik Schadow
33 */
34 @Component
35 public class PlainSqlQuery {
36     private JdbcTemplate jdbcTemplate;
37
38     public PlainSqlQuery(JdbcTemplate jdbcTemplate) {
39         this.jdbcTemplate = jdbcTemplate;
40     }
41
42     public List<Customer> query(String name) {
43         String query = "SELECT * FROM customer WHERE name = '" + name + "' ORDER BY id";
44
45         List<Map<String, Object>> rows = jdbcTemplate.queryForList(query);
46
47         return CustomerRowMapper.mapRows(rows);
48     }
49 }

```

Use a variable binding mechanism to construct this query instead of concatenation. ...

vor 3 Monaten ▾ L45 \$3 T-

 Vulnerability ▾  Critical ▾  Open ▾ Dominik Schadow 20min effort Comment

cert, cwe, hibernate, owasp-a1, sans-top25-insecure, sql ▾

# OWASP Top 10 Proactive Controls 2016

**C01 Verify for Security Early and Often**

**C02 Parameterize Queries**

**C03 Encode Data**

**C04 Validate All Inputs**

C05 Implement Identity and Authentication Controls

C06 Implement Appropriate Access Controls

**C07 Protect Data**

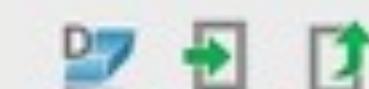
C08 Implement Logging and Intrusion Detection

C09 Leverage Security Frameworks and Libraries

**C10 Error and Exception Handling**

Works on my machine

```
$ mvn clean compile findbugs:findbugs
[INFO] Scanning for projects...
[INFO] -----
[INFO] Reactor Build Order:
[INFO]
[INFO] Java Security
[INFO] Access Control - Spring Security
[INFO] Crypto Hash
[INFO] Crypto Java
[INFO] Crypto Keyczar
[INFO] Crypto Shiro
[INFO] Content Security Policy - Spring Security
[INFO] Cross-Site Request Forgery
[INFO] Cross-Site Request Forgery - Spring Security
[INFO] Direct Object References
[INFO] Intercept Me
[INFO] Security Header
[INFO] Security Logging
[INFO] Session Handling
[INFO] Session Handling - Spring Security
[INFO] SQL-Injection
[INFO] SSO with GitHub
[INFO] Cross-Site Scripting
[INFO]
[INFO] -----
[INFO] Building Java Security 2.0.0
[INFO] -----
[INFO]
[INFO] --- maven-clean-plugin:2.6.1:clean (default-clean) @ javasecurity ---
[INFO]
[INFO] --- maven-enforcer-plugin:1.4.1:enforce (enforce-maven) @ javasecurity ---
[INFO]
[INFO] --- findbugs-maven-plugin:3.0.4:findbugs (default-cli) @ javasecurity ---
```

SearchOther Settings > FindBugs-IDEA For current project[General](#) [Report](#) [Filter](#) [Detector](#) [Annotate](#) [Share](#)

- Compile affected files before analyze
- Analyze affected files after compile
- Analyze affected files after auto make
- Run analyze in background
- Activate toolwindow on run

## Plugins

**Find Security Bugs** (com.h3xstream.findsecbugs)

- Find Security Bugs is a plugin that aims to help security audit.  
<https://find-sec-bugs.github.io>

Project 1: Project

JavaSecurity [javasecurity] ~/Devel

```
LOGGER.error(ex.getMessage(), ex);

LOGGER.info("Order servlet: CSRF token is valid");

String product = request.getParameter("product");
int quantity = Integer.parseInt(request.getParameter("quantity"));

LOGGER.info("Ordered {} items of product {}", quantity, product);

response.setContentType("text/html");

try (PrintWriter out = response.getWriter()) {
    out.println("<html>");
    out.println("<head>");
    out.println("<link rel=\"stylesheet\" type=\"text/css\" href=\"resources/css/styles.css\" />");
    out.println("<title>Cross-Site Request Forgery (CSRF): Order Confirmation</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<h1>Cross-Site Request Forgery (CSRF): Order Confirmation</h1>");
    out.println("<p><strong>Ordered " + quantity + " of product " + product + "</strong></p>");
    out.println("<p><a href=\"index.jsp\">Home</a></p>");
    out.println("</body>");
    out.println("</html>");
} catch (IOException ex) {
    LOGGER.error(ex.getMessage(), ex);
}
```

FindBugs-IDEA

1: Structure 2: Favorites 3: Persistence

CRITICAL INJECTION FOR I/Os (13 items)

- Cross site scripting vulnerability (9 items)
  - Servlet reflected cross site scripting
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet

Preview OrderServlet.java:

```
...
87     out.println("<title>Cross-Site Request
88     out.println("</head>");
89     out.println("<body>");
90     out.println("<h1>Cross-Site Request Fo
91     out.println("<p><strong>Ordered " + qu
92     out.println("<p><a href=\"index.jsp\">
93     out.println("</body>");
94     out.println("</html>");
95 } catch (IOException ex) {
96     LOGGER.error(ex.getMessage(), ex);
97 }
98 }
```

HTTP parameter written to Servlet output

Class:

[OrderServlet](#) (de.dominiksshadow.javasecurity.csri)

Method:

`doPost`

Servlet reflected cross site scripting vulnerability

This code directly writes an HTTP parameter to Servlet output, which allows for a reflected cross site scripting attack.

JavaSecurity > sql-injection > src > main > java > de > dominikshadow > javasecurity > database > PlainSqlQuery

Session Handling (Spring Security)

Project JavaSecurity [javasecurity]

PlainSqlQuery.java

```
PlainSqlQuery query()
    ...
    /**
     * Servlet using a plain Statement to query the in-memory-database. User input is not modified and used directly in the
     * SQL query. (@code ' or '1'='1) is a good input to return all statements, (@code 'drop table customer;--') to delete
     * the complete table.
     *
     * @author Dominik Schadow
     */
    @Component
    public class PlainSqlQuery {
        private JdbcTemplate jdbcTemplate;

        public PlainSqlQuery(JdbcTemplate jdbcTemplate) { this.jdbcTemplate = jdbcTemplate; }

        public List<Customer> query(String name) {
            String query = "SELECT * FROM customer WHERE name = '" + name + "' ORDER BY id";
            List<Map<String, Object>> rows = jdbcTemplate.queryForList(query);

            return CustomerRowMapper.mapRows(rows);
        }
    }
```

SonarLint: Current file Project files Log

Found 1 issue in 1 file

PlainSqlQuery.java (1 issue)

(45, 67) Use a variable binding mechanism to construct this query instead of concatenation.

Rule Locations

### SQL binding mechanisms should be used

Vulnerability Blocker squid:S2077

Applications that execute SQL commands should neutralize any externally-provided values used in those commands. Failure to do so could allow an attacker to include input that changes the query so that unintended commands are executed, or sensitive data is exposed.

This rule checks a variety of methods from different frameworks which are susceptible to SQL injection if not used properly. Frameworks which are covered are Java JDBC, JPA, JDO, Hibernate and Spring. The following specific method signatures are tested.

- org.hibernate.Session.createQuery
- org.hibernate.Session.createSQLQuery
- java.sql.Statement.executeQuery
- java.sql.Statement.execute
- java.sql.Statement.executeUpdate
- java.sql.Statement.executeLargeUpdate
- java.sql.Statement.addBatch
- java.sql.Connection.prepareStatement
- java.sql.Connection.prepareCall
- java.sql.Connection.nativeSQL

Automatic analysis is enabled

TODO FindBugs-IDEA Docker Java Enterprise Version Control Spring Terminal SonarLint

46:64 LF: UTF-8 Git: master No Gitflow Event Log

**END  
SLIDE  
AREA**

O -  
TING

# OWASP Top 10 Proactive Controls 2016

**C01 Verify for Security Early and Often**

**C02 Parameterize Queries**

**C03 Encode Data**

**C04 Validate All Inputs**

**C05 Implement Identity and Authentication Controls**

**C06 Implement Appropriate Access Controls**

**C07 Protect Data**

**C08 Implement Logging and Intrusion Detection**

**C09 Leverage Security Frameworks and Libraries**

**C10 Error and Exception Handling**

# Alternatives and Extensions

## **Arachni**

Web application security scanner

## **Clair**

Static analysis of vulnerabilities in Docker containers

## **Gauntlet**

Provides hooks for a variety of security tools

## **SourceClear**

Cloud based dependency checker

A photograph of a man sleeping peacefully in a bed. In the foreground, a traditional metal alarm clock sits on a surface, its white face showing the time as approximately 7:15. The background is softly blurred, focusing attention on the clock.

Scan early,  
scan often

A white cup of coffee with steam rising, sitting on a saucer, surrounded by coffee beans.

Scans slow down  
your build

A photograph of a person lying in a grassy field, viewed from behind. The person is wearing blue plaid shorts and a pink shirt. They are looking towards a range of mountains in the distance. The sun is low, casting a warm glow over the scene.

Check security in  
nightly build

Let a security champion  
take care of the findings





Don't wait until the  
last minute to clear  
the findings



Don't do duplicate scans

# Filter false positives



Tools do not replace  
other activities





Marienstr. 17  
70178 Stuttgart

dominik.schadow@bridging-it.de  
[www.bridging-it.de](http://www.bridging-it.de)

Blog [blog.dominiksshadow.de](http://blog.dominiksshadow.de)  
Twitter @dschadow

## FindSecBugs

<http://find-sec-bugs.github.io>

## Just-in-time Code-Analysis

<https://blogs.uni-paderborn.de/sse/2017/05/19/issta-jit-analysis>

## OWASP Dependency Check

[https://www.owasp.org/index.php/OWASP\\_Dependency\\_Check](https://www.owasp.org/index.php/OWASP_Dependency_Check)

## OWASP Zed Attack Proxy

[https://www.owasp.org/index.php/OWASP\\_Zed\\_Attack\\_Proxy\\_Project](https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project)

## Pictures

<http://www.dreamstime.com>

