

# Real World Java Web Security



Java User  
Group Ostfalen

Dominik Schadow | [bridgingIT](#)















# Who thinks about ...

... architecture while coding?

... architecture before coding?

# Who thinks about ...

... security while coding?

... security before coding?

# Who wants to ...

... develop secure applications?







# OWASP TOP 10 2013

A01 Injection

A02 Broken Authentication and Session Management

A03 Cross-Site Scripting (XSS)

A04 Insecure Direct Object References

A05 Security Misconfiguration

A06 Sensitive Data Exposure

A07 Missing Function Level Access Control

A08 Cross-Site Request Forgery (CSRF)

A09 Using Components with Known Vulnerabilities

A10 Unvalidated Redirects and Forwards







# OWASP 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



# Verify for Security Early and Often

Know the web application

Know all external entities

Identify all data flows

**Identify all risks**



# Threat

Anything that threatens the application, its data  
or any other asset



# Common security flaws

Forget to authenticate a user

Broken authorization

No auditing functionality

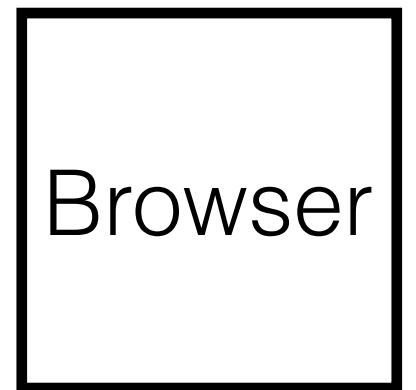
Using components with known vulns



# Data Flow Diagrams

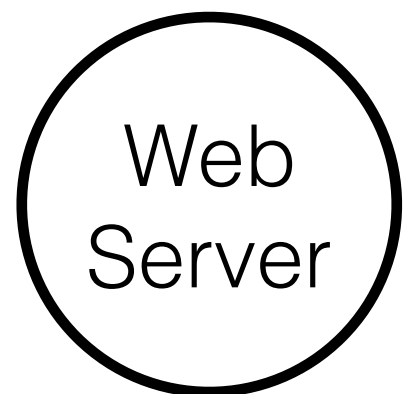
**External Entity**

People or code outside your control



**Process**

Any running code



**Data Store**

Things that store data



**Data Flow**

Communication between processes or processes and data stores





Browser



Web  
Server



App  
Server



Database

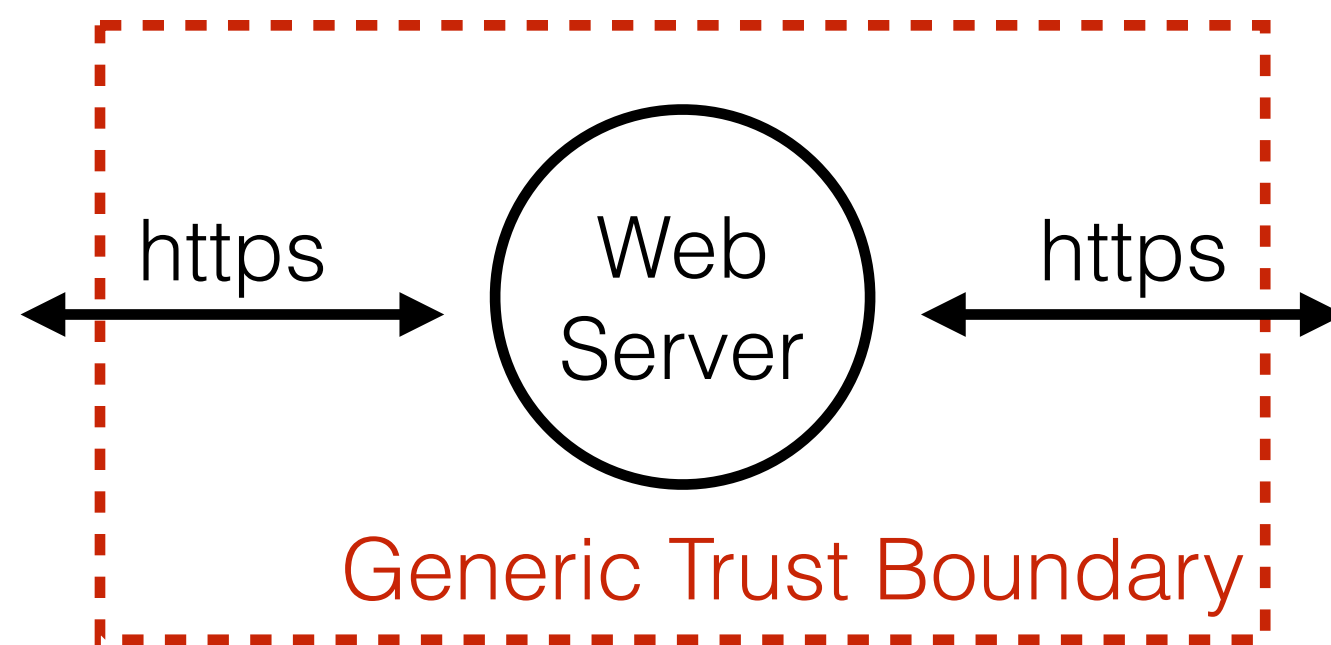


# Trust Boundaries

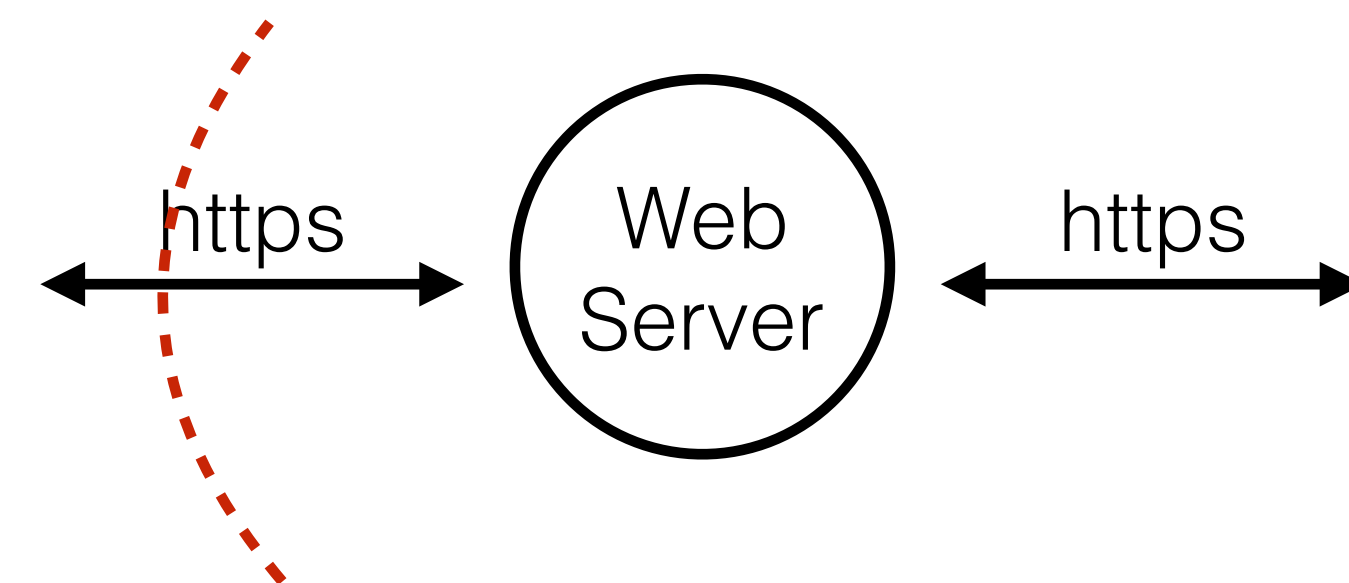
## Trust Boundary

Where entities with different privileges interact

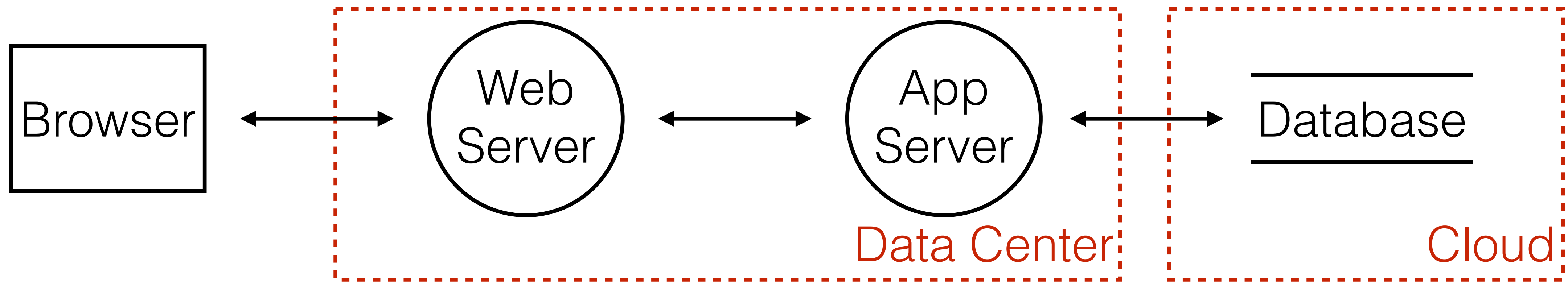
Generic Trust Boundary



Generic Trust Boundary









# Where are the threats?

**Follow the data flow**

Start with data crossing boundaries



# STRIDE

## **S**poofing

Pretending to be something or somebody else  
Violated property: **Authentication**

## **T**ampering

Modifying something on disk, network or memory  
Violated property: **Integrity**

## **R**epudiation

Claiming that someone didn't do something  
Violated property: **Non-Repudiation**



# STRIDE

## **I**nformation **D**isclosure

Providing information to someone not authorized  
Violated property: **Confidentiality**

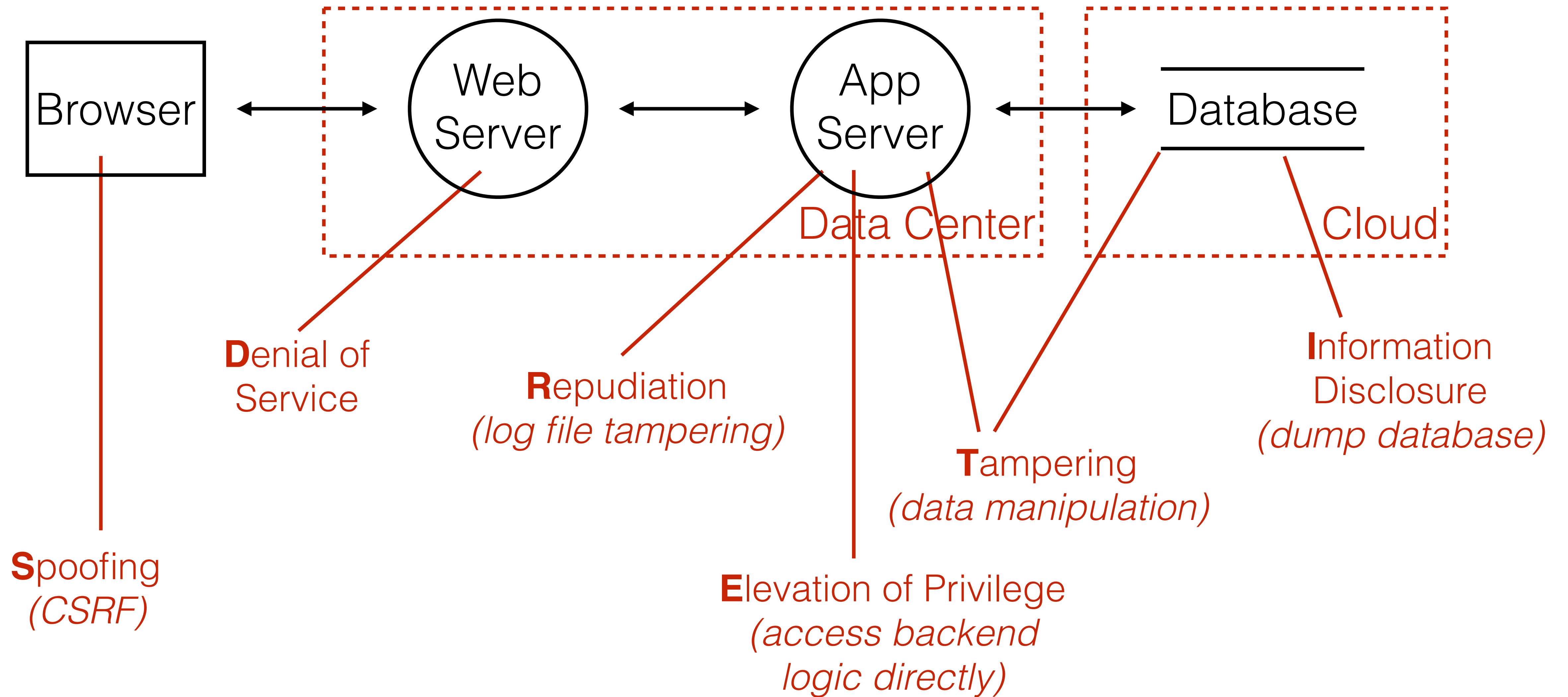
## **D**enial of **S**ervice

Absorbing resources needed to provide service  
Violated property: **Availability**

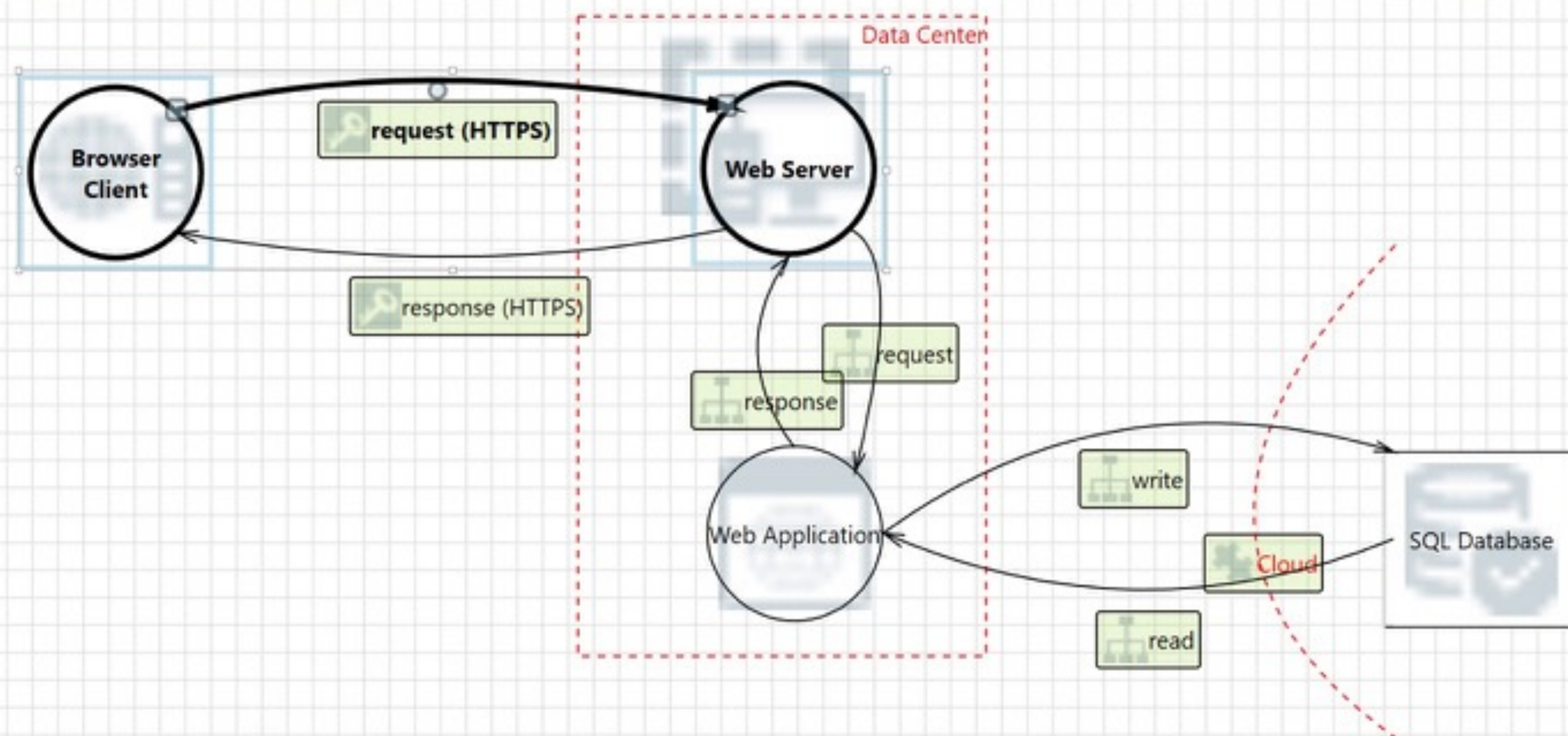
## **E**levation **o**f Privilege

Doing something someone is not authorized to do  
Violated property: **Authorization**









## Threat List

ID	Title	Category	Description	Justification	Interaction	Diagram	Changed By	Last Modified	State	Priority
1	Spoofing the Browser Client Proc...	Spoofing	Browser Client...		request (HTTPS)	Duke Encount...		28.02.2016 14:0...	Not Started	High
2	Cross Site Scripting	Tampering	The web server...		request (HTTPS)	Duke Encount...		28.02.2016 14:0...	Not Started	High
3	Potential Data Repudiation by We...	Repudiation	Web Server cla...		request (HTTPS)	Duke Encount...		28.02.2016 14:0...	Not Started	High
4	Potential Process Crash or Stres fo...	Denial Of Servi...	Web Server cra...		request (HTTPS)	Duke Encount...		28.02.2016 14:0...	Not Started	High

44 Threats Displayed, 44 Total

## Threat Properties

ID: 2    Diagram: Duke Encounters    Status:     Last Modified: 28.02.2016 14:07:53

Title:

Category:

Description:

Threat Properties:



# **Demo**

## **Threat Modeling**



<b>Threat Target</b>	<b>Mitigation Strategy</b>	<b>Mitigation Technique</b>	<b>Priority</b>	<b>Issue ID</b>
Repudiating actions	Log	Logging all security relevant actions in an audit log	2	1001
Spoofing a user	Identification and authentication	Password policy, token, password reset process	1	1002
Network flooding	Elastic cloud	Dynamic cloud resources (servers and databases) to provide service	3	1006
Tampering network packets	Cryptography	HTTPS/TLS	1	1007






**Fight the  
identified  
threats by  
priority**





**Maintain your threat models**



A row of red triangular handrails on a train platform, receding into the distance. The handrails are attached to a metal pole and are spaced out along the platform. The background is a blurred train platform with a brick wall and some lights.

# Leverage Security Frameworks and Libraries



# Frameworks and libraries decline





# The unfortunate reality of insecure libraries

- Up to 80% of code in today's applications comes from libraries and frameworks
- 113 million downloads analyzed for the 31 most popular Java libraries and frameworks
- 26% had known vulnerabilities (29 million)
- Most vulnerabilities are undiscovered

Jeff Williams & Arshan Dabirsiaghi  
The Unfortunate Reality of Insecure Libraries  
Aspect Security (March 2012)



```
Marvin:JavaSecurity dos$ dependency-check --project JavaSecurity --scan ./**/*.jar
```

```
[INFO] Checking for updates
```

```
[INFO] Download Started for NVD CVE - Modified
```

```
[INFO] Download Complete for NVD CVE - Modified (595 ms)
```

```
[INFO] Processing Started for NVD CVE - Modified
```

```
[INFO] Processing Complete for NVD CVE - Modified (1468 ms)
```

```
[INFO] Begin database maintenance.
```

```
[INFO] End database maintenance.
```

```
[INFO] Check for updates complete (15456 ms)
```

```
[INFO] Analysis Starting
```

```
[INFO] Creating the CPE Index
```

```
[INFO] CPE Index Created (2092 ms)
```

```
[INFO] Analysis Complete (63657 ms)
```

```
Marvin:JavaSecurity dos$
```



```
## preparation
```

```
mvn dependency:copy-dependencies
```

```
## single project
```

```
dependency-check --project SampleProject  
--scan target/dependency
```

```
## multiple projects
```

```
dependency-check --project MultipleProjects  
--scan ./**/*.jar
```



```
## preparation
```

```
mvn dependency:copy-dependencies
```

```
## single project
```

```
dependency-check --project SampleProject  
--scan target/dependency
```

```
## multiple projects
```

```
dependency-check --project MultipleProjects  
--scan ./**/*.jar
```



```
## preparation
```

```
mvn dependency:copy-dependencies
```

```
## single project
```

```
dependency-check --project SampleProject  
--scan target/dependency
```

```
## multiple projects
```

```
dependency-check --project MultipleProjects  
--scan ./**/*.jar
```



```
## preparation
```

```
mvn dependency:copy-dependencies
```

```
## single project
```

```
dependency-check --project SampleProject  
--scan target/dependency
```

```
## multiple projects
```

```
dependency-check --project MultipleProjects  
--scan ./**/*.jar
```





# 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.

## Project: MultipleProjects

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

- *dependency-check* version: 1.3.6
- *Report Generated On*: Mai 27, 2016 at 12:31:02 MESZ
- *Dependencies Scanned*: 172
- *Vulnerable Dependencies*: 4
- *Vulnerabilities Found*: 4
- *Vulnerabilities Suppressed*: 0
- ...

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

Dependency	CPE	GAV	Highest Severity	CVE Count	CPE Confidence	Evidence Count
<a href="#">jstl-1.2.jar</a>	cpe:/a:apache:standard_taglibs:1.2.1	<a href="#">javax.servlet:jstl:1.2</a>	High	1	LOW	18
<a href="#">commons-beanutils-1.8.3.jar</a>	cpe:/a:apache:commons_beanutils:1.8.3	<a href="#">commons-beanutils:commons-beanutils:1.8.3</a>	High	1	LOW	24
<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	21
<a href="#">xalan-2.7.0.jar</a>	cpe:/a:apache:xalan-java:2.7.0	<a href="#">xalan:xalan:2.7.0</a>	High	1	HIGHEST	28

## Dependencies

### jstl-1.2.jar

**File Path:** JavaSecurity/access-control-spring-security/target/dependency/jstl-1.2.jar  
**MD5:** 51e15f798e69358cb893e38c50596b9b  
**SHA1:** 74aca283cd4f4b4f3e425f5820cda58f44409547

Evidence





```
<reporting>
  <plugins><plugin>
    <groupId>org.owasp</groupId>
    <artifactId>dependency-check-maven</artifactId>
    <version>1.3.6</version>
    <reportSets>
      <reportSet>
        <reports>
          <report>aggregate</report>
        </reports>
      </reportSet>
    </reportSets>
  </plugin></plugins>
</reporting>
```



# Jenkins integration

- Dependency Check might take too long for an automatic build after every push
- Extend only the nightly build job with Dependency Check
- Don't set up an individual NVD per job, use a centralized one and update it separately



# Create a NVD update only job

## Build Triggers

- Trigger builds remotely (e.g., from scripts)
- Build after other projects are built
- Build periodically

Schedule

@weekly

Would last have run at Saturday, February 13, 2016 10:02:02 PM CET; would next run at Saturday, February 20, 2016 10:02:02 PM CET.

- Build when a change is pushed to GitHub
- Poll SCM

## Build Environment

- Delete workspace before build starts
- Send files or execute commands over SSH before the build starts
- Send files or execute commands over SSH after the build runs
- Install custom tools

## Build

Invoke OWASP Dependency-Check NVD update only

Data directory

/Users/dos/owasp-nvd

- Enable verbose logging
- Skip if triggered by SCM changes
- Skip if triggered by upstream changes

Delete



# Reference the database in every build

Post Steps

Run only if build succeeds  Run only if build succeeds or is unstable  Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Invoke OWASP Dependency-Check analysis

Path to scan

Output directory

Data directory

Suppression file

ZIP extensions

Post-build Actions

- Use Maven artifacts as scan path (deprecated)
- Disable NVD auto-update
- Enable verbose logging
- Generate optional HTML reports
- Skip if triggered by SCM changes
- Skip if triggered by upstream changes

Publish OWASP Dependency-Check analysis results

Dependency-Check results

Fileset includes setting that specifies the generated raw Dependency-Check XML report files, such as \*\*/dependency-check-report.xml. Basedir of the fileset is the workspace root. If no value is set, then the default \*\*/dependency-check-report.xml is used. Be sure not to include any non-report files into this pattern.

Run always  
By default, this plug-in runs only for stable or unstable builds, but not for failed builds. If this plug-in should run even for failed builds then activate this check box.

Detect modules  
Determines if Ant or Maven modules should be detected for all files that contain warnings. Activating this option may increase your build time since the detector scans the whole workspace for 'build.xml' or 'pom.xml' files in order to assign the correct module names.

Health thresholds

Configure the thresholds for the build health. If left empty then no health report is created. If the actual number of warnings is between the provided thresholds then the build health is interpolated.

Health priorities  Only priority high  Priorities high and normal  All priorities

Determines which warning priorities should be considered when evaluating the build health.

Status thresholds (Totals)	All priorities	Priority high	Priority normal	Priority low
<input type="text" value="10"/>	<input type="text" value="4"/>	<input type="text" value="6"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

If the number of total warnings is greater than one of these thresholds then a build is considered as unstable or failed, respectively. I.e., a value of 0 means that the build status is changed if there is at least one warning found. Leave this field empty if the state of the build should not depend on the number of warnings.

Compute new warnings (based on the last successful build unless another reference build is chosen below)

Default Encoding

Default encoding when parsing or showing files. Leave this field empty to use the default encoding of the platform.

Trend graph [You can define the default values for the trend graph in a separate view.](#)



# Dependency-Check Result

## Warnings Trend

All Warnings	New Warnings	Fixed Warnings
4	<u>4</u>	<u>4</u>

## Summary

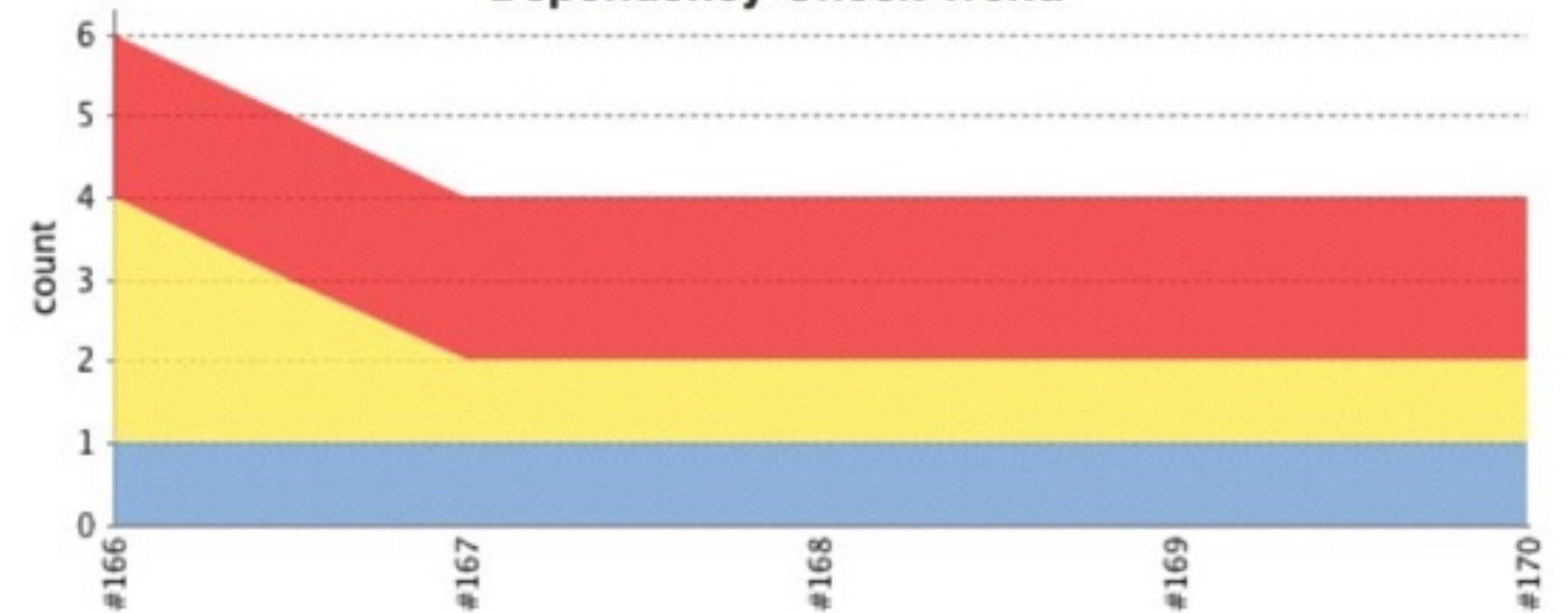
Total	High Priority	Normal Priority	Low Priority
4	<u>2</u>	<u>1</u>	<u>1</u>

## Details

- CWEs
- CVEs
- Warnings
- Details
- New
- Fixed
- High
- Medium
- Low

Category	Total	Distribution
<a href="#">CWE-20 Improper Input Validation</a>	1	
<a href="#">CWE-22 Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')</a>	1	
<a href="#">CWE-79 Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')</a>	1	
Total	4	

Dependency-Check Trend





**Analyze your code**





Preferences

Other Settings > FindBugs-IDEA For current project

Search

- Appearance & Behavior
- Keymap
- Editor
- Plugins
- Version Control
- Build, Execution, Deployment
- Languages & Frameworks
- Tools
- Other Settings
  - Checkstyle
  - FindBugs-IDEA**
  - HOCON
  - SonarLint General Settings
  - SonarLint Project Settings

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://github.com/h3xstream/find-sec-bugs>

Preferences

Other Settings > FindBugs-IDEA For current project

General Report Filter **Detector** Annotate Share

! Disabled detectors will not participate in the analysis.  
'Grayed out' detector will run, however they will not report any result to the UI.

Search Filter Provider

- Provider
  - Find Security Bugs
    - BadHexadecimalConversionDetector**
    - BroadcastDetector
    - CipherWithNoIntegrityDetector
    - CommandInjectionDetector
    - ConstantPasswordDetector
    - CookieReadDetector
    - CrifLogInjectionDetector
    - CustomInjectionDetector
    - CustomMessageDigestDetector
    - DesUsageDetector
    - EsapiEncryptorDetector
    - ExternalConfigurationControlDetector
    - ExternalFileAccessDetector
    - FileUploadFilenameDetector
    - GeolocationDetector
    - GoogleApiKeyDetector
    - HazelcastSymmetricEncryptionDetector
    - HttpResponseSplittingDetector
    - InsecureCookieDetector

Description

Identify Bad hexadecimal concatenation

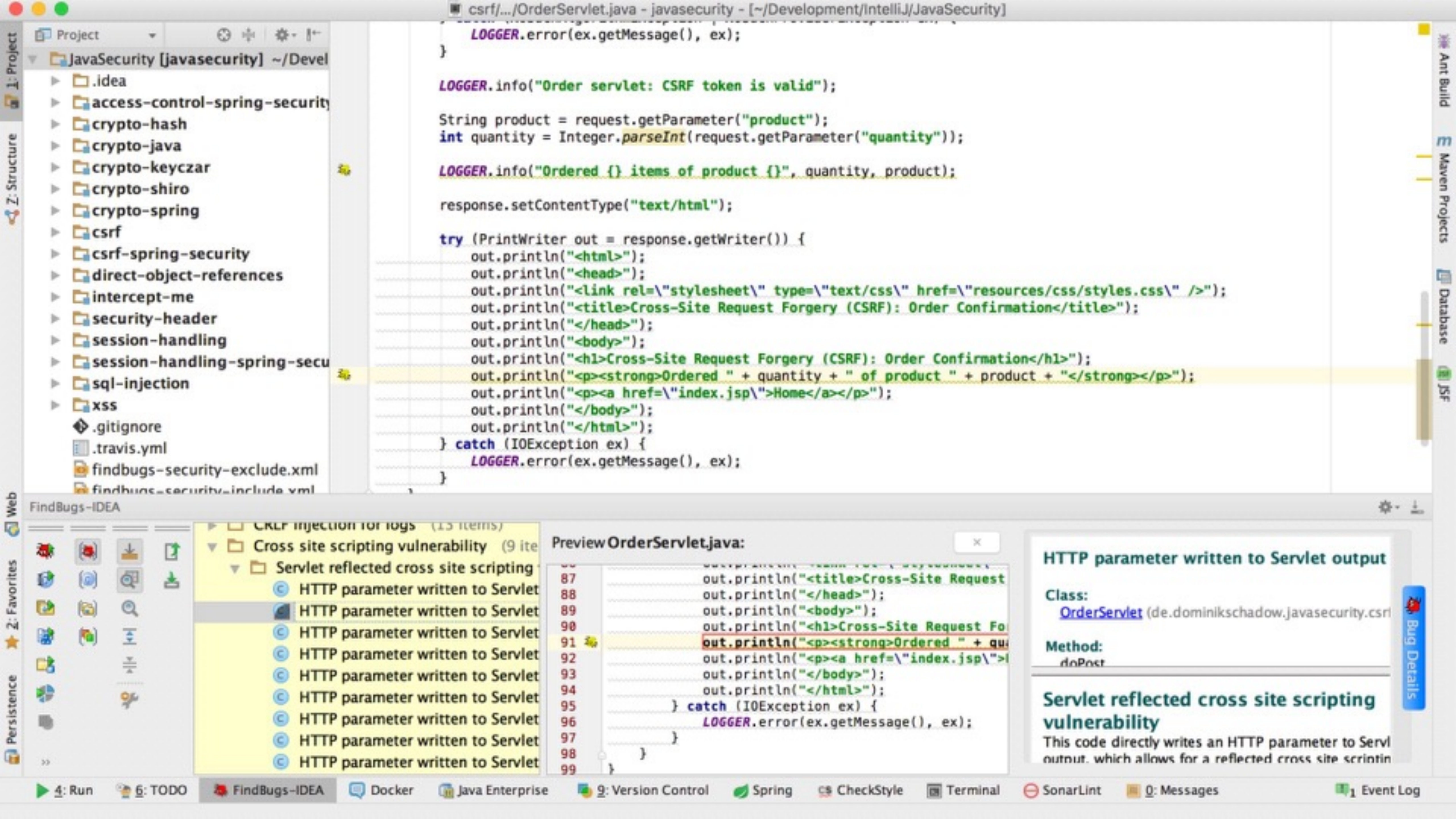
Plugin	Find Security Bugs
Plugin Id	com.h3xstream.findsecbugs
Detector Name	BadHexadecimalConversionDetector

Reported patterns:

SECURITY	SECBHC	BAD_HEXA_CONVERSION	Bad hexadecimal concatenation
----------	--------	---------------------	-------------------------------

Cancel Apply OK





```
    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);
}
```

- CRLP Injection for logs (15 items)
- Cross site scripting vulnerability (9 items)
  - Servlet reflected cross site scripting
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - HTTP parameter written to Servlet
    - 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 }
99 }
```

**HTTP parameter written to Servlet output**

Class:  
[OrderServlet](#) (de.dominikschadow.javasecurity.csrf)

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



**Demo**

**FindSecurityBugs**

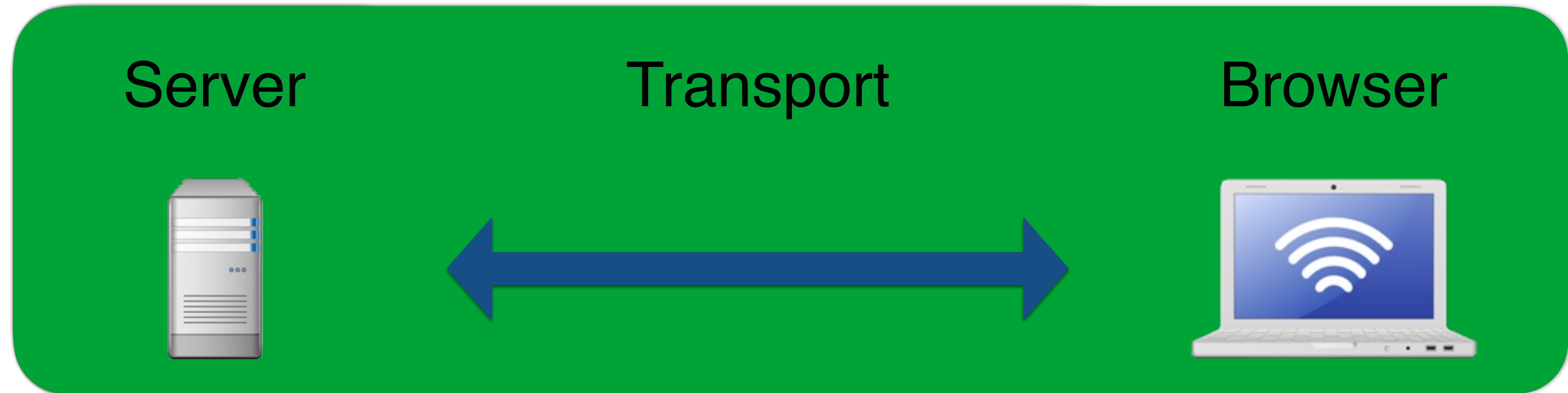


# **Beyond the top 10**





# Defense in depth





# Security response headers

X-Frame-Options

Content Security Policy (CSP)

HTTP Strict Transport Security (HSTS)

HTTP Public Key Pinning (HPKP)



# Individual Servlet filter for every header

- ❑ Intercepts all responses
  - ❑ Always identical configuration
  - ❑ Easier to test
- ❑ A single place to configure each policy
- ❑ Easier to integrate in other web applications (ok: copy)



```
@WebFilter(filterName = "CacheControlFilter", urlPatterns = {"/*"})
public class CacheControlFilter implements Filter {
    public void doFilter(ServletRequest sReq, ServletResponse sRes,
        FilterChain fc) {
        HttpServletResponse response = (HttpServletResponse) sRes;
        response.setHeader("Cache-Control",
            "no-cache, must-revalidate, max-age=0, no-store");
        fc.doFilter(servletRequest, response);
    }

    public void init(FilterConfig filterConfig){}

    public void destroy() {}
}
```



```
@WebFilter(filterName = "CacheControlFilter", urlPatterns = {"/*"})
public class CacheControlFilter implements Filter {
    public void doFilter(ServletRequest sReq, ServletResponse sRes,
        FilterChain fc) {
        HttpServletResponse response = (HttpServletResponse) sRes;
        response.setHeader("Cache-Control",
            "no-cache, must-revalidate, max-age=0, no-store");
        fc.doFilter(servletRequest, response);
    }

    public void init(FilterConfig filterConfig){}

    public void destroy() {}
}
```



```
@WebFilter(filterName = "CacheControlFilter", urlPatterns = {"/*"})
public class CacheControlFilter implements Filter {
    public void doFilter(ServletRequest sReq, ServletResponse sRes,
        FilterChain fc) {
        HttpServletResponse response = (HttpServletResponse) sRes;
        response.setHeader("Cache-Control",
            "no-cache, must-revalidate, max-age=0, no-store");
        fc.doFilter(servletRequest, response);
    }

    public void init(FilterConfig filterConfig){}

    public void destroy() {}
}
```



```
response.addHeader(  
    "Policy name",  
    "Policy value"  
);
```



**Browser must  
understand  
header**

**Additional  
security  
layer**



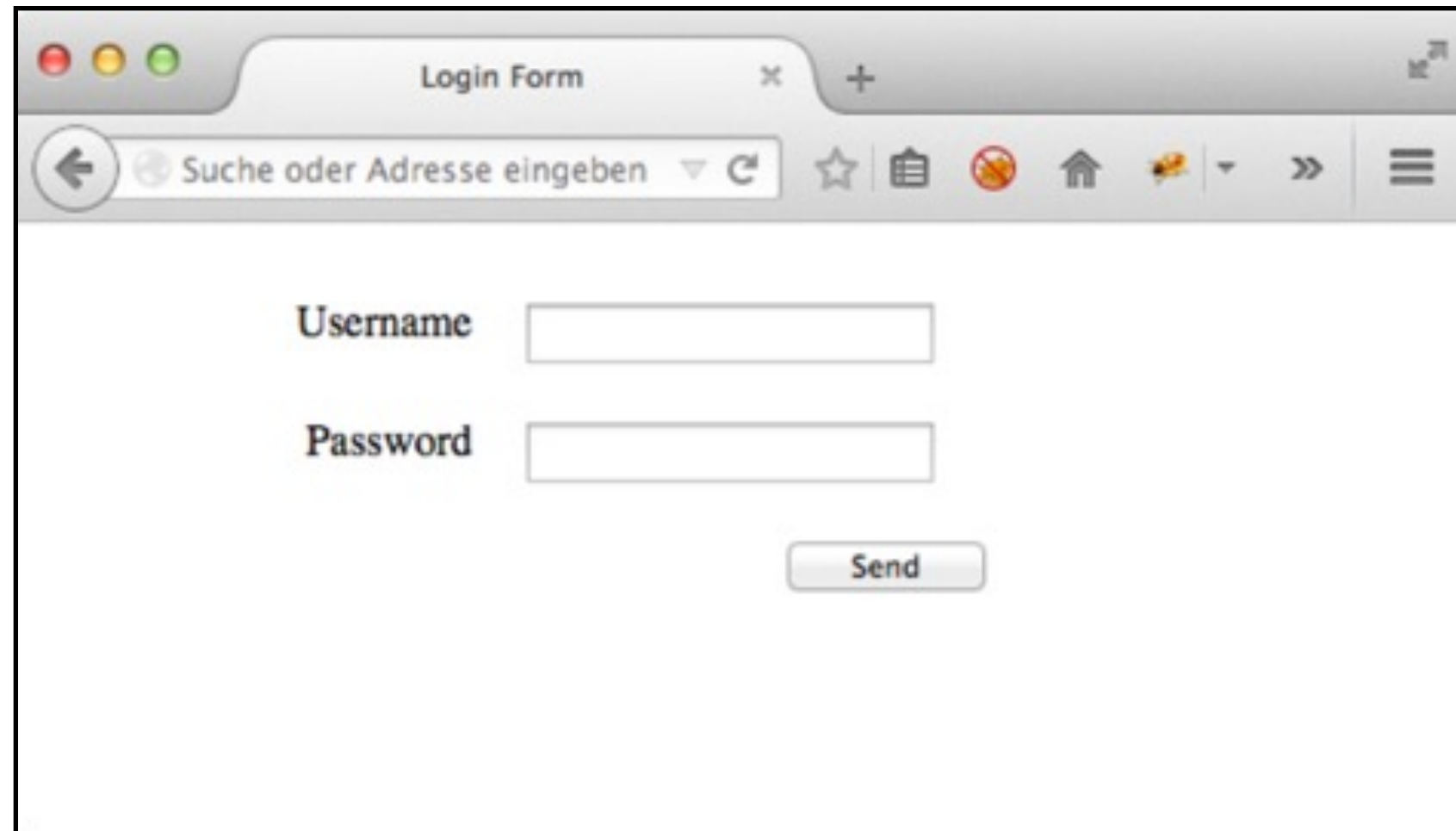


# X-Frame-Options

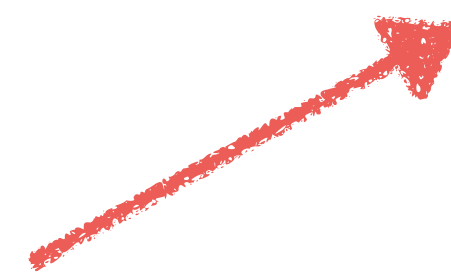
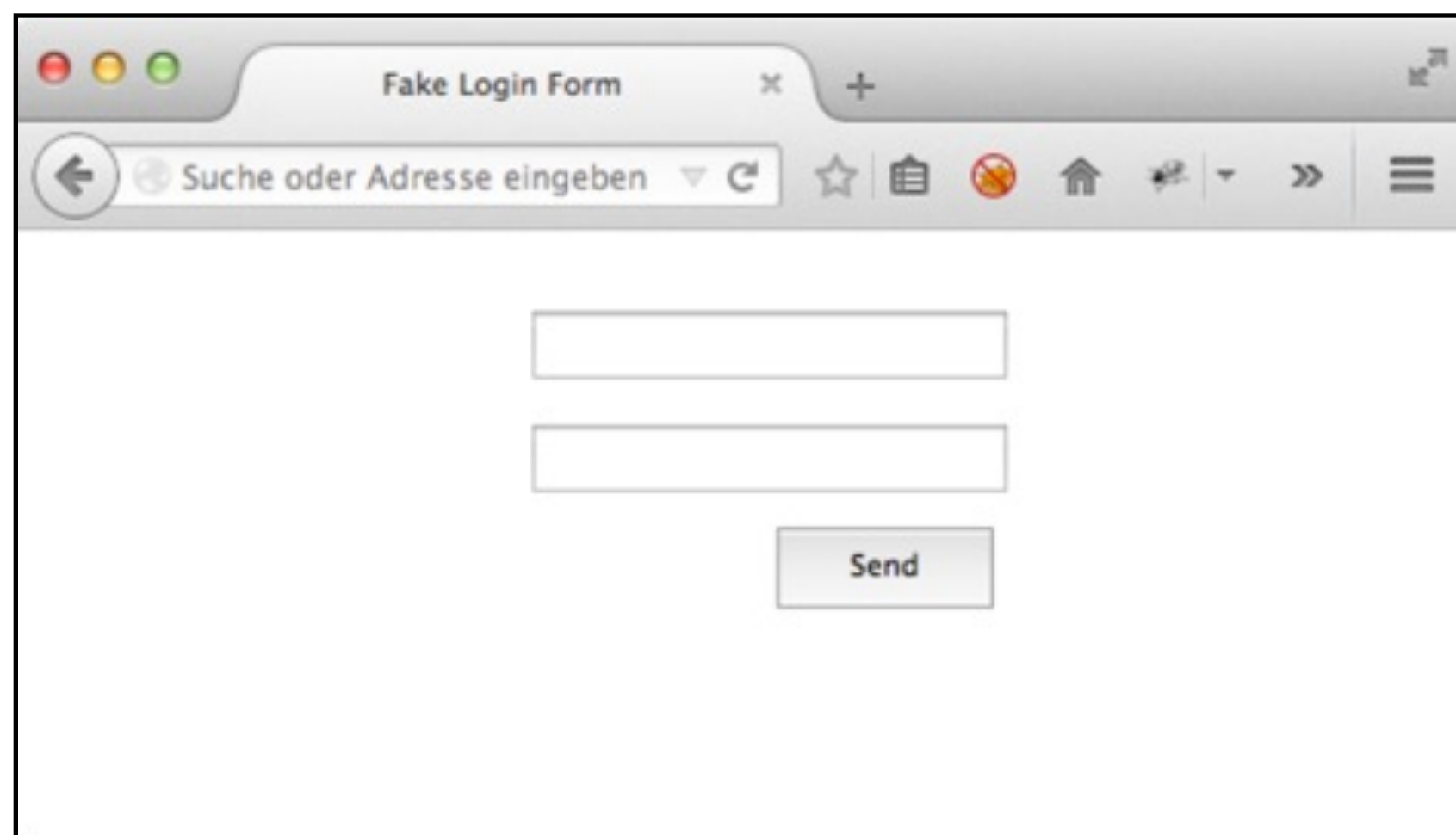
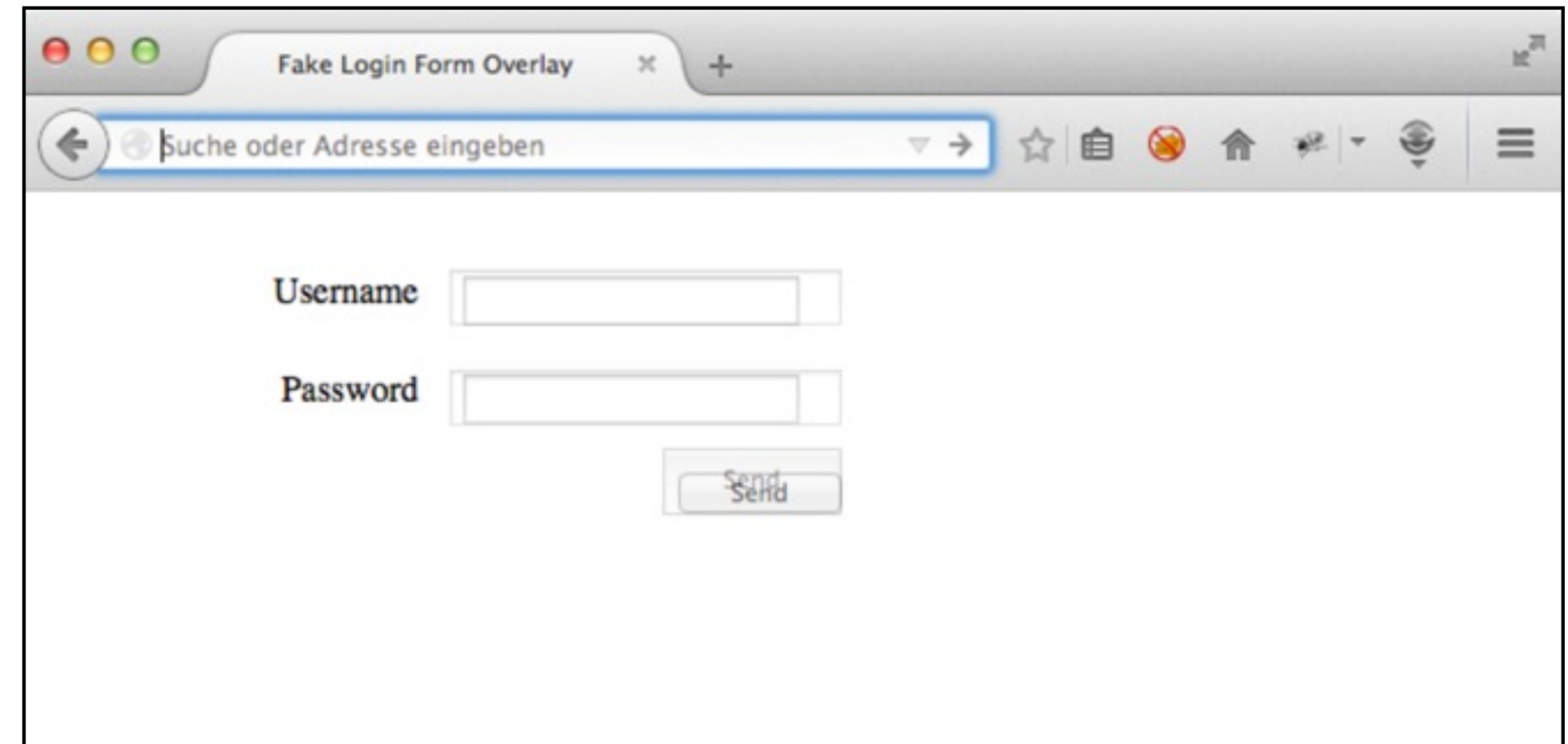
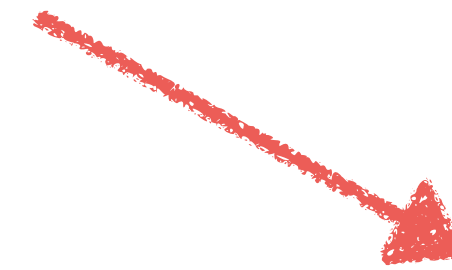
Prevents UI redressing attacks



# UI redressing attacks in a nutshell



*iframe*



*div*



```
response.setHeader(  
    "X-Frame-Options",  
    "DENY"  
);
```

```
"SAME-ORIGIN"
```

```
"ALLOW-FROM [uri]"
```



# X-Frame-Options browser compatibility



ALLOW-FROM only supported  
in Firefox and Internet Explorer

# Content Security Policy (CSP)

Whitelist all content  
Prevents content injection  
(Cross-Site Scripting)



```
response.setHeader(  
    "Content-Security-Policy",  
    "default-src 'self'"  
);
```

# Content Security Policy Directives

<b>default-src</b>	default if specific directive is not set
<b>object-src</b>	Sources in object, embed or applet tags
<b>script-src</b>	Script sources (includes XSLT)
connect-src	XMLHttpRequest, WebSocket, ...
font-src	Font sources
child-src	Sources embeddable as frames/ iframes
img-src	Image sources
media-src	Video and audio sources
style-src	CSS sources (does not include XSLT)



```
response.addHeader(  
    "Content-Security-Policy",  
    "default-src 'none';  
    script-src 'self';  
    image-src 'self';  
    font-src 'self' https://fonts.googleapis.com;  
    style-src 'self' https://fonts.googleapis.com"  
);
```

# Violation Report

```
{  
  "document-uri": "http://.../reporting.jsp?  
    name=%3Cscript%3Ealert(%27XSS%27)%3C/script%3E",  
  "referrer": "http://www.sample.com/security-header/  
    index.jsp",  
  "blocked-uri": "self",  
  "violated-directive": "default-src http://www.sample.com",  
  "source-file": "http://.../reporting.jsp?  
    name=%3Cscript%3Ealert(%27XSS%27)%3C/script%3E",  
  "script-sample": "alert('XSS')",  
  "line-number": 10  
}
```



# CSP Level 1 browser compatibility

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
8			45					4.3	
9			46					4.4	
10		43	47			8.4		4.4.4	
11	13	44	48	9	34	9.2	8	47	47
	14	45	49	9.1	35	9.3			
		46	50		36				
		47	51						

# Adding CSP to your application

- ❑ Don't use inline styles or scripts
- ❑ Start with `default-src: 'none'`  
(or `default-src: 'self'`)
- ❑ Configure other directives to make your application work
- ❑ Specify a report URI and improve the CSP header
- ❑ Use a generator, browser developer tools and an analyzer



# One header to rule them all

- CSP Level 2 will replace other headers in the future
- Be aware that older browsers understand the current (old) headers but will never understand CSP Level 2

# Content Security Policy 2 extensions

## **frame-ancestors**

Allow resource frame embedding (obsoletes X-Frame-Options header)

## **reflected-xss**

(De-)activate user agent XSS heuristics (obsoletes X-XSS-Protection header)

## **upgrade-insecure-request**

Load everything over HTTPS, even if URL specifies HTTP (page is loaded via HTTPS)

## **block-all-mixed-content**

Prevent browser from loading any assets using HTTP when using HTTPS



```
response.addHeader(  
    "Content-Security-Policy",  
    "default-src 'self';  
    frame-ancestors 'none' "  
);
```

# Upgrade requests to HTTPS

- ❑ Forces browsers to upgrade any link automatically to HTTPS
- ❑ Does not block the request, but upgrades it
- ❑ Implemented as CSP directive or meta tag



```
<meta http-equiv="Content-  
Security-Policy" content="upgrade-  
insecure-requests">
```

# CSP Level 2 browser compatibility

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
			29						
			45						
			48					4.3	
8			49			8.4		4.4	
9		7 45	50	9	36	9.2		4.4.4	
11	13	7 46	51	9.1	37	9.3	8	50	50
	14	7 47	52	TP	38				
		7 48	53		39				
		7 49	54						



[Home](#) > [Tools](#) > [CSP Builder](#)

## Here is your Policy!

**Content-Security-Policy: default-src 'none' ; script-src 'self' ; style-src 'self' ; img-src 'self' ; font-src 'self' ; frame-ancestors 'none' ; form-action 'self' ; upgrade-insecure-requests; block-all-mixed-content; reflected-xss block;**

### Import a policy

### Build your CSP

1) Default Source

2) Script Source

3) Style Source

4) Image Source

5) Font Source

6) Connect Source

7) Media Source

8) Object Source

9) Child Source

Default Source [View Info](#)

None

All

Self

Data

Unsafe Inline

Unsafe Eval

Space separated list of hosts.

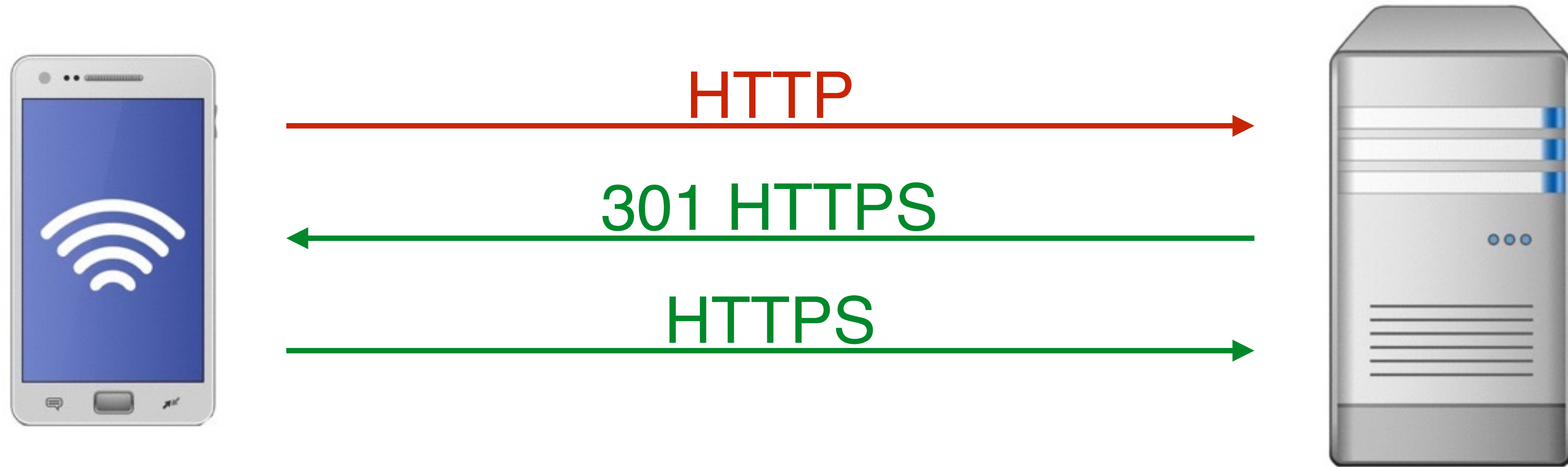
# HTTP Strict Transport Security (HSTS)

Force HTTPS

Prevent TLS stripping

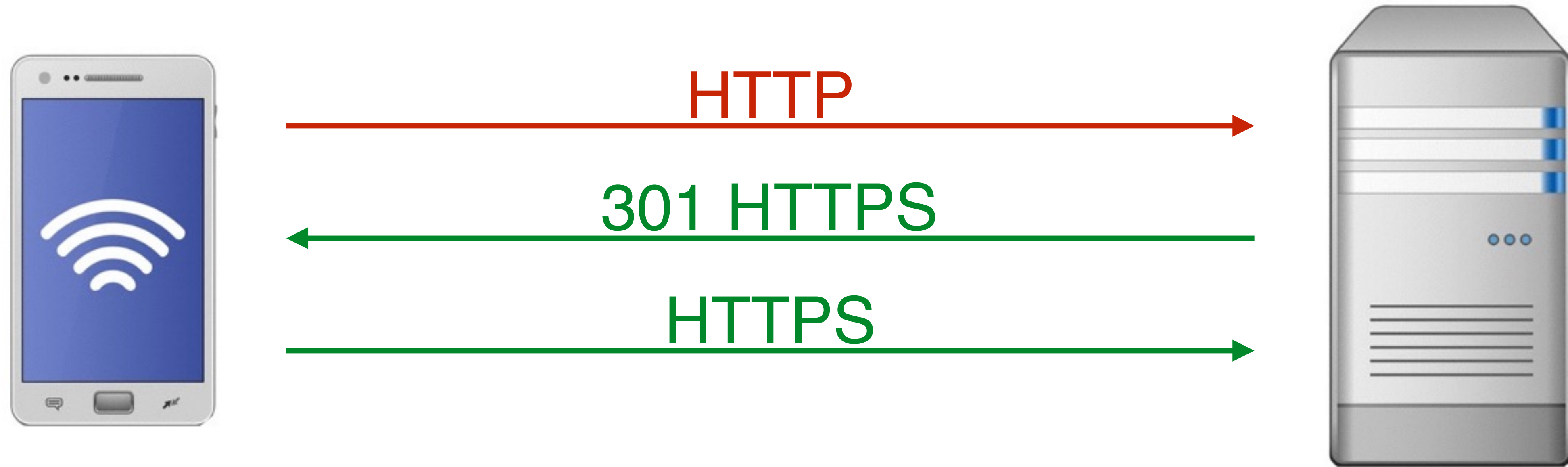


# Without HSTS, first call



Server is issuing a 301 redirect

# Without HSTS, second call



Server is issuing a 301 redirect

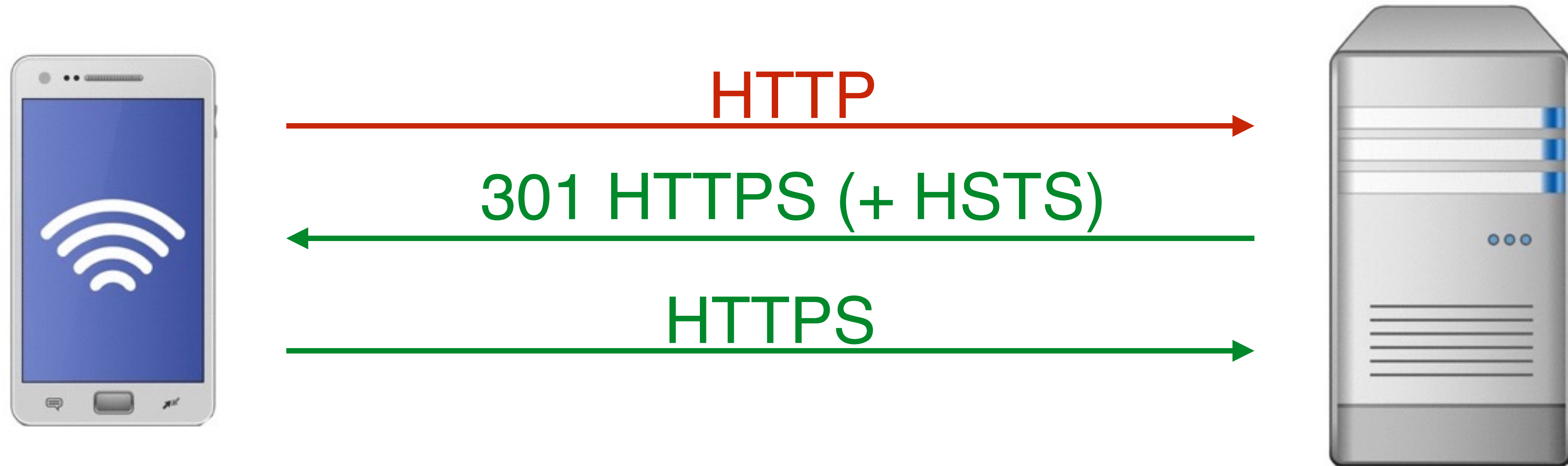


```
response.setHeader(  
    "Strict-Transport-Security",  
    "max-age=31556926"  
);
```

```
"max-age=31556926; includeSubDomains"
```

```
"max-age=31556926; includeSubDomains; preload"
```

# With HSTS, first call



Server is issuing a 301 redirect



# With HSTS, second call



Browser is issuing a 307 internal redirect

# HSTS requirements

- The configured duration should not expire, there will be an initial unprotected request otherwise
- All resources must be available via HTTPS, includes any (external) scripts, images, ...
- Valid certificate required, no self-signed certificates any more
- Requires a HTTPS connection, not active on HTTP connections



# HSTS preload list

- Preload list for HSTS hosts hard coded into Chrome
  - Included in Firefox, Internet Explorer and Safari
- Requires the complete HSTS header

```
Strict-Transport-Security "max-age=31556926;  
includeSubDomains; preload"
```
- Submit your page at <https://hstspreload.appspot.com>

# HSTS browser compatibility

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
8			45					4.3	
9			46					4.4	
10		43	47			8.4		4.4.4	
11	13	44	48	9	34	9.2	8	47	47
	14	45	49	9.1	35	9.3			
		46	50		36				
		47	51						

<http://caniuse.com/#feat=stricttransportsecurity>



# HTTP Public Key Pinning (HPKP)

Fixes the broken CA system

Be careful, invalid hash  
prevents page access

```
response.setHeader(  
    "Public-Key-Pins",  
    "pin-sha256='eSC+HM0...wuKgUzr4=';  
    pin-sha256='7HIpact...oQYcRhJ3Y=';  
    max-age=5184000;  
    includeSubdomains;  
    report-uri='https://...'" );  
);
```





Home > Tools > [HPKP Hash Generator](#)

## Create your HPKP hash

https://www.google.de

Hash

Here is your PKP hash for www.google.de: pin-sha256="e5C+HMY06A72RnZr9RiGuLhQX7IgtOvXoNBwuKgUzr4="

Here is your PKP hash for Google Internet Authority G2: pin-sha256="7HlpactklAq2Y49orFOOQKurWxmmSFZhBCoQYcRhj3Y="

Here is your PKP hash for GeoTrust Global CA: pin-sha256="h6801m+z8v3zbgkRHpq6L29Esgfzhj89C15yUCOQmqU="

Here is your PKP hash for Equifax Secure Certificate Authority: pin-sha256="/1aAzXOlcD2gSBegdf1GJQanNQbEuBoVg+9UIHJSZHY="

### About

report-uri.io is a project created by [Scott Helme](#)

### Contact

Email: [info@report-uri.io](mailto:info@report-uri.io)

### Tweet

Send us a tweet [@reporturi](#)

# HPKP browser compatibility

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Chrome for Android
8			45					4.3	
9			46					4.4	
10		43	47			8.4		4.4.4	
11	13	44	48	9	34	9.2	8	47	47
	14	45	49	9.1	35	9.3			
		46	50		36				
		47	51						

<http://caniuse.com/#feat=publickeypinning>





Most headers are only active  
in the current response

HSTS  
+  
HPKP



# Scan your site now

 Hide results

## Security Report Summary



Site: <https://www.google.de/>

IP Address: 2607:f8b0:4005:803::2003

Report Time: 13 Feb 2016 16:01:56 UTC

Headers:

✓ X-XSS-Protection ✓ X-Frame-Options ✗ Strict-Transport-Security ✗ Content-Security-Policy  
✗ Public-Key-Pins ✗ X-Content-Type-Options

## Raw Headers

HTTP/1.1	200 OK
Date	Sat, 13 Feb 2016 16:01:56 GMT
Expires	-1
Cache-Control	private, max-age=0
Content-Type	text/html; charset=UTF-8
P3P	CP="This is not a P3P policy! See <a href="https://www.google.com/support/accounts/answer/1516577?hl=en">https://www.google.com/support/accounts/answer/1516577?hl=en</a> for more info."
Server	gws
X-XSS-Protection	1; mode=block
X-Frame-Options	SAMEORIGIN
Set-Cookie	CONSENT=WP_250320; expires=Fri, 01-Jan-2038 00:00:00 GMT; path=/; domain=.google.de
Alternate-Protocol	443:quic,p=1
Alt-Svc	quic="www.google.com:443"; ma=2592000; v="30,29,28,27,26,25",quic=":443"; ma=2592000; v="30,29,28,27,26,25"
Accept-Ranges	none
Vary	Accept-Encoding
Transfer-Encoding	chunked

## Missing Headers





## Page, Header & Cookie Security Analyser








Analysis results for:

 <https://blog.dominikschadow.de/>



Click the icons in the tables below for a more detailed explanation.

### HTTP security headers

Name	Value	Setting secure
x-content-type-options	nosniff	
x-frame-options	deny	
cache-control	no-cache, must-revalidate, max-age=0, no-store, no-cache, must-revalidate	
content-security-policy	default-src 'self'; img-src *; font-src *; style-src 'self' https://fonts.googleapis.com 'unsafe-inline'; frame-ancestors 'none'	
strict-transport-security	max-age=31556926	
x-xss-protection	1; mode=block	
access-control-allow-origin	Header not returned	

**Recx** is a HTTP header and cookie security analyzer plugin for Google Chrome

Alternatives are the **Security Headers** extensions for Chrome and Firefox

# **Demo**

## **security-header**

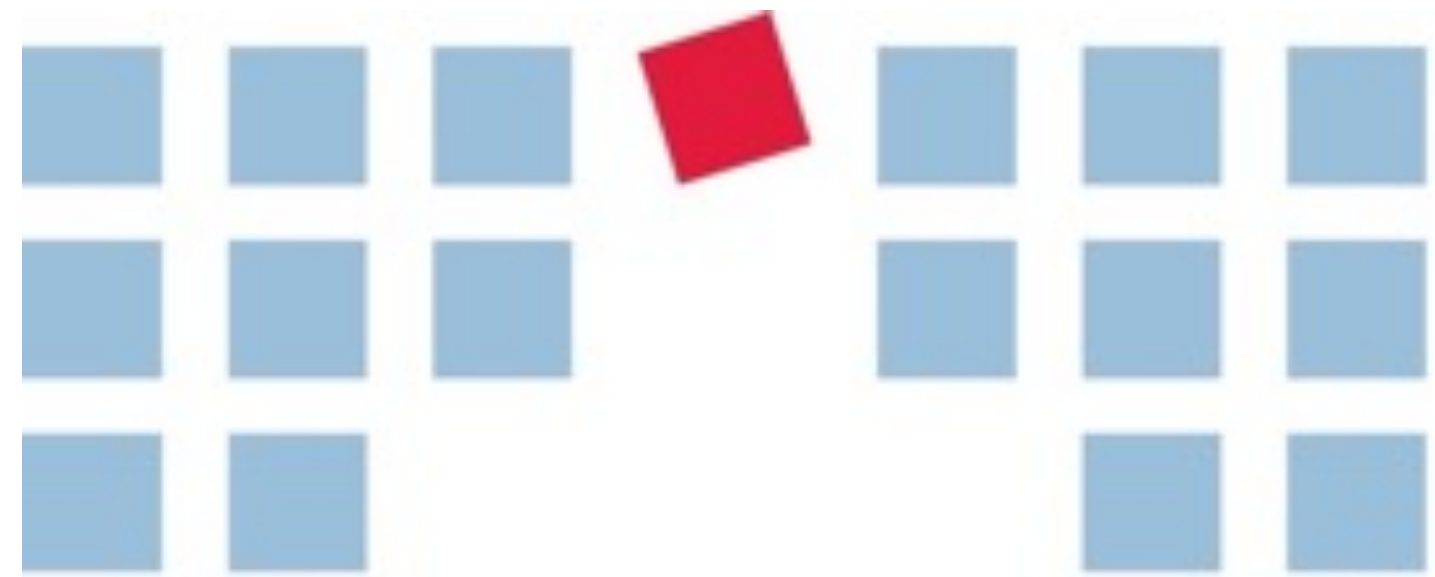


# Summary

Use the OWASP Proactive Controls  
as a real world guide

Start small and secure your  
development part first

Spread the word



# bridging IT

Marienstraße 17  
70178 Stuttgart

dominik.schadow@bridging-it.de  
www.bridging-it.de

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

## Demo Projects

[github.com/dschadow/JavaSecurity](https://github.com/dschadow/JavaSecurity)

## Microsoft Threat Modeling Tool

[www.microsoft.com/en-us/sdl/adopt/threatmodeling.aspx](http://www.microsoft.com/en-us/sdl/adopt/threatmodeling.aspx)

## OWASP Dependency Check

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

## OWASP TOP 10

[www.owasp.org/index.php/Category:OWASP\\_Top\\_Ten\\_Project](http://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project)

## OWASP TOP 10 Proactive Controls

[www.owasp.org/index.php/OWASP\\_Proactive\\_Controls](http://www.owasp.org/index.php/OWASP_Proactive_Controls)

## Recx Security Analyser

[www.recx.co.uk/products/chromeplugin.php](http://www.recx.co.uk/products/chromeplugin.php)

## Spring Security

[projects.spring.io/spring-security](http://projects.spring.io/spring-security)

## Pictures

[www.dreamstime.com](http://www.dreamstime.com)

