Java Web Security Antipatterns

JavaOne 2015

Dominik Schadow | bridgingIT


Failed with nothing but
the best intentions

Architect

Implement

Maintain

Architect

Skipping threat modeling

Software that is secure by design

Know the web application

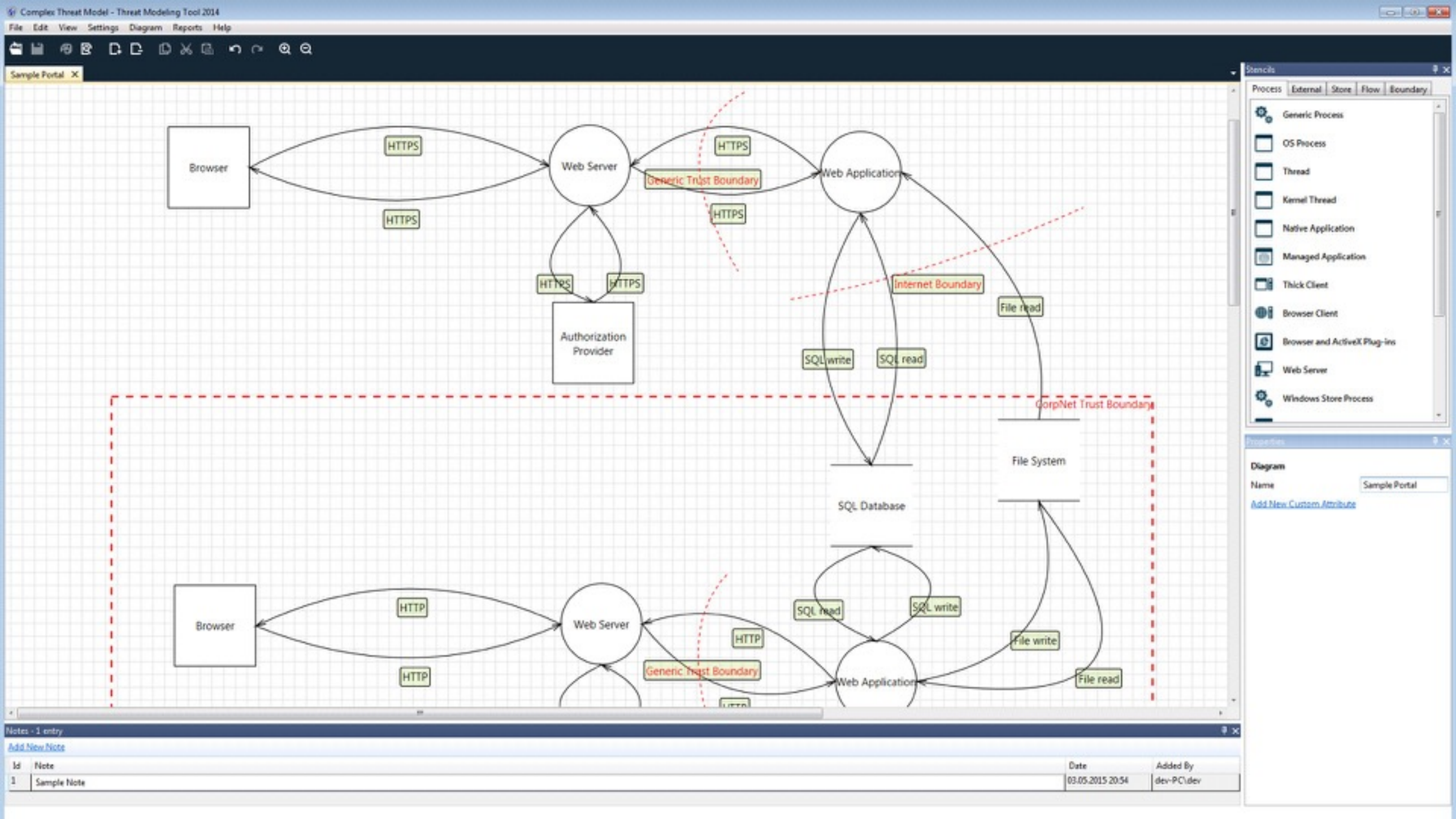
Know all external entities

Know all data flows

Identify all risks

Threat model

Avoid design flaws


Complex Threat Model - Threat Modeling Tool 2014
File Edit View Settings Diagram Reports Help
Sample Portal
Browser
HTTPS
HTTPS
Web Server
HTTPS
Generic Trust Boundary
HTTPS
Web Application
HTTPS
HTTPS
Authorization Provider
Internet Boundary
File read
SQL write
SQL read
CorpNet Trust Boundary
File System
SQL Database
Browser
HTTP
HTTP
Web Server
HTTP
Generic Trust Boundary
SQL read
SQL write
File write
File read
Web Application
Stencils
Process External Store Flow Boundary
Generic Process
OS Process
Thread
Kernel Thread
Native Application
Managed Application
Thick Client
Browser Client
Browser and ActiveX Plug-ins
Web Server
Windows Store Process
Properties
Diagram
Name
Sample Portal
Add New Custom Attribute
Notes - 1 entry
Add New Note
Id
Note
Date
Added By
1
Sample Note
03.05.2015 20:54
dev-PC\dev

SeaSponge

mozilla.github.io/seasponge/#/draw

SeaSponge | File | Zoom-In | Zoom-Out | Web Application Sample | Refresh Sharable Link | Share link | Help

Model Information

Diagram Information

New Threat

Threat Information

High SQL Injection

Stencils

Stencil Properties - Generic Trust Boundary

Boundary

Delete Element

Title

Database

Category

Boundary

Tags

Comma separated tags

Icon

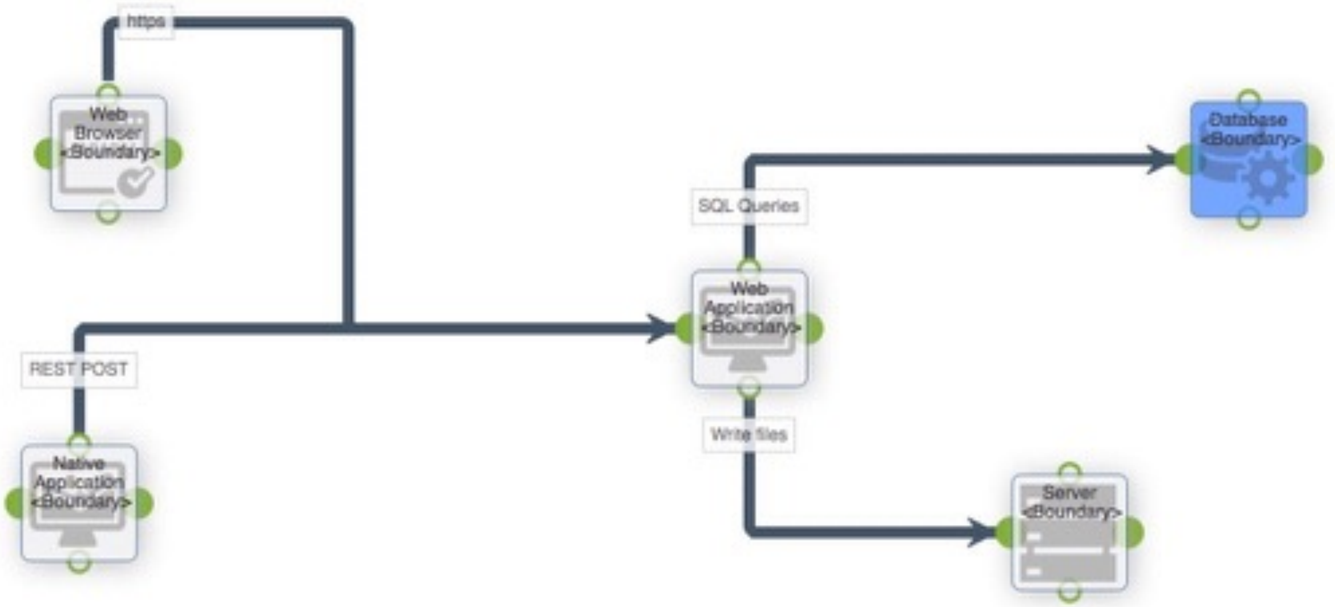
images/icons/database11.svg

Code Type

Managed

Running As

Web Application Sample | + Diagram | Reports



„Good engineering involves thinking about how things can be made to work; the security mindset involves thinking about how things can be made to fail. It involves thinking like an attacker […].“

Bruce Schneier

www.schneier.com/blog/archives/2008/03/the_security_mi_1.html

Implement

User passwords stored

-  as plaintext
-  encrypted
-  trivially hashed

Password Retriever

Forgotten your password? No problem! Just enter your email address and postcode below and click the "Retrieve" button. Your password will display on the screen. An email with your password will be also sent to you, please make sure the email address entered is the same as you used when you created your account.

Email *

Display password on screen directly

Retrieve

Important: for your security, please change your password after you get the old one back. To change your password, please Go Here

Passw0rd$

d281fdbe0555b913d1c29f99143a3ad7bc66cf83

2e2c68bc1e9187cc6919fcb8564f1483

AKNtqLC_DZM32Jk7pgF4FpRVapo6QFEdROpsflwHkw
2q6rfK2mev4fAQFlRXbH2DecJTYLvF3LMD

Passw0rd$

SHA1

~~d281fdbe0555b913d1c29f99143a3ad7bc66cf83~~

2e2c68bc1e9187cc6919fcb8564f1483

AKNtqLC_DZM32Jk7pgF4FpRVapo6QFEdROpsflwHkw

2q6rfK2mev4fAQFlRXbH2DecJTYLvF3LMD

Passw0rd$

SHA1

~~d281fdbe0555b913d1c29f99143a3ad7bc66cf83~~

MD5

~~2e2c68bc1e9187cc6919fcb8564f1483~~

AKNtqLC_DZM32Jk7pgF4FpRVapo6QFEdROpsflwHkw

2q6rfK2mev4fAQFlRXbH2DecJTYLvF3LMD

Passw0rd$

SHA1

~~d281fdbe0555b913d1c29f99143a3ad7bc66cf83~~

MD5

~~2e2c68bc1e9187cc6919fcb8564f1483~~

AES

~~AKNtqLC_DZM32Jk7pgF4FpRVapo6QFEdROpsflwHkw~~

~~2q6rfK2mev4fAQFlRXbH2DecJTYLvF3LMD~~

Slow down brute force attacks

PBKDF2

Iterations against brute force attacks

Available in plain Java

Demo

bcrypt

Iterations against brute force attacks

Integrated in Spring Security

```
@Configuration
@EnableWebMvcSecurity
public class WebSecurityConfig extends
 WebSecurityConfigurerAdapter {
 @Bean
 public PasswordEncoder passwordEncoder() {
  return new BCryptPasswordEncoder(10);
 }
}
```

```
@Configuration
@EnableWebMvcSecurity
public class WebSecurityConfig extends
 WebSecurityConfigurerAdapter {
 @Bean
 public PasswordEncoder passwordEncoder() {
  return new BCryptPasswordEncoder(10);
 }
}
```

```
@Configuration
@EnableWebMvcSecurity
public class WebSecurityConfig extends
 WebSecurityConfigurerAdapter {
  @Bean
  public PasswordEncoder passwordEncoder() {
   return new BCryptPasswordEncoder(10);
  }
}
```

```
@Configuration
@EnableWebMvcSecurity
public class WebSecurityConfig extends
 WebSecurityConfigurerAdapter {
 @Bean
 public PasswordEncoder passwordEncoder() {
  return new BCryptPasswordEncoder(10);
 }
}
```

scrypt

Memory against brute force attacks

Best protection against dictionary attacks

Increase # iterations with faster hardware

Set period of time to change passwords

User logs in successfully

Calculate new salt

Calculate new hash

Update hash & salt

Period of time expired

Set not changed passwords to null

User tries to log in

Force password reset process

Enforce length limit on password fields

```
<h:inputSecret id="password" maxlength="1024">
 <f:validateLength minimum="10" maximum="1024"/>
</h:inputSecret>
```

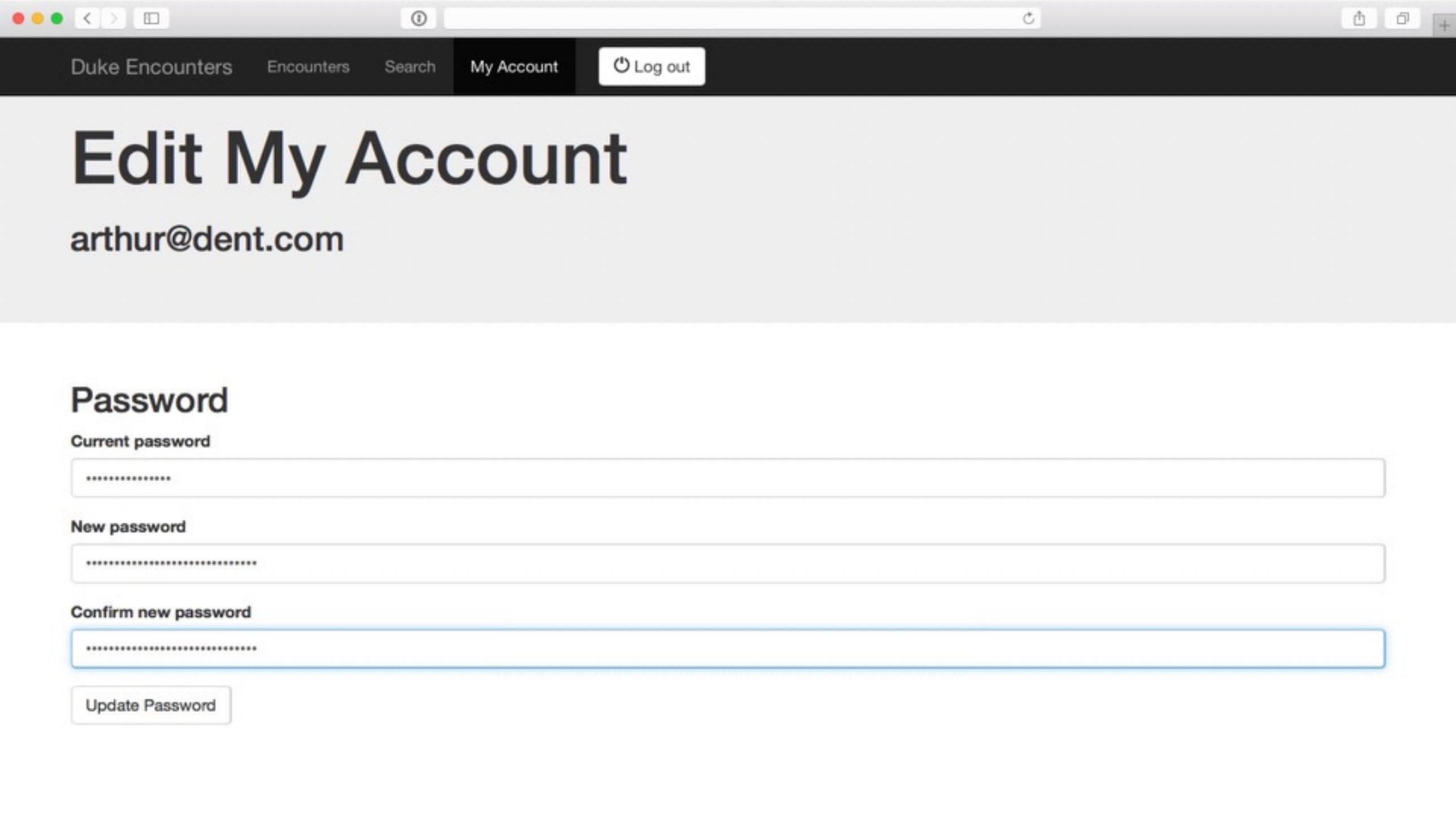
```
<h:inputSecret id="password" maxlength="1024">
 <f:validateLength minimum="10" maximum="1024"/>
</h:inputSecret>
```

```
private byte[] hash(PBEKeySpec keySpec) {
  return secretKeyFactory.generateSecret
      (keySpec).getEncoded();
}
```

```
private byte[] hash(PBEKeySpec keySpec) {
   return secretKeyFactory.generateSecret
        (keySpec).getEncoded();
}
```

Implement

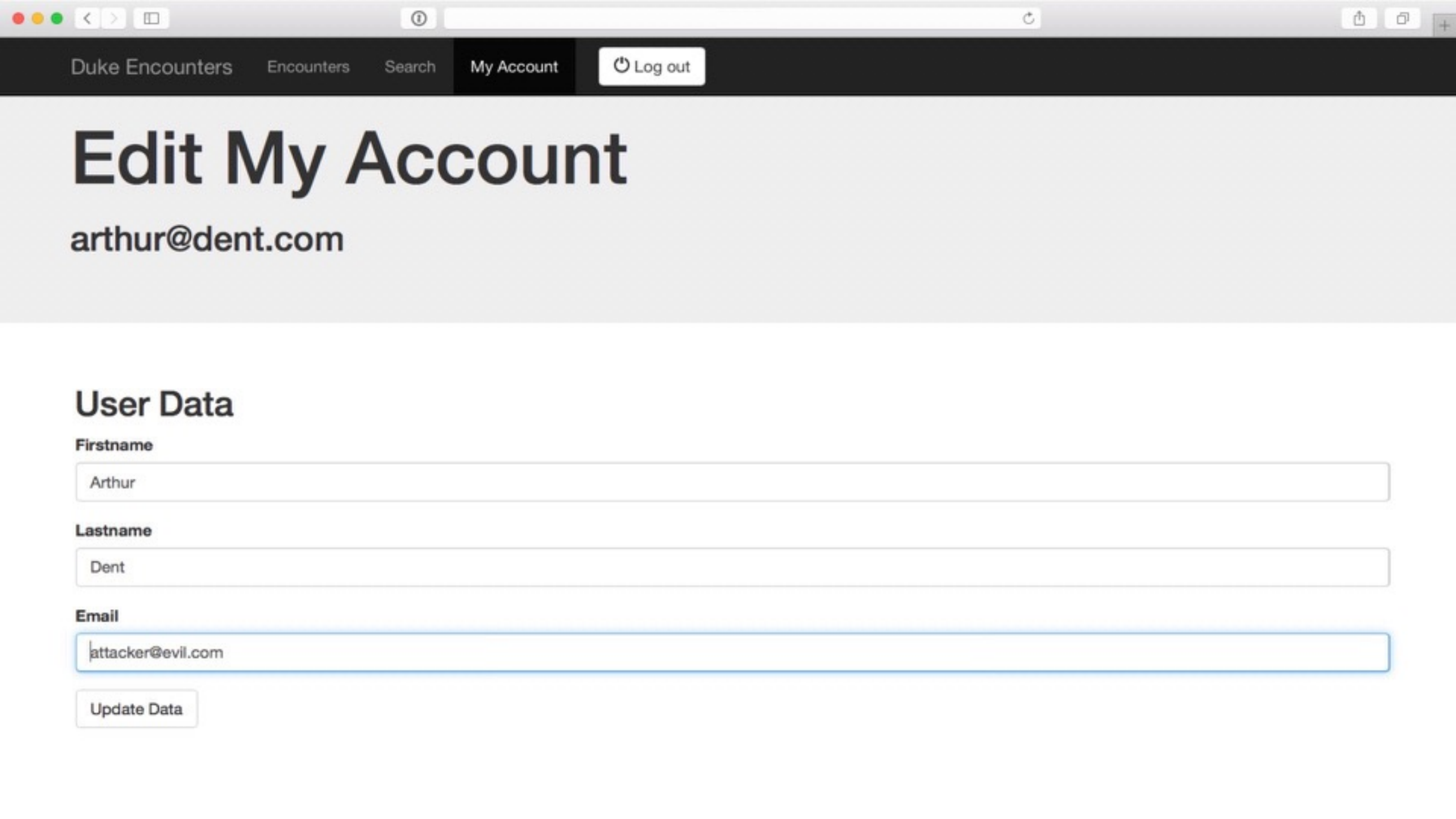
- Changing password
- Changing email address


Duke Encounters
Encounters
Search
My Account
Log out
Edit My Account
arthur@dent.com
Password
Current password
New password
Confirm new password
Update Password

Prevent unintended password change

Cross-Site Request Forgery vulnerability

Session id knowledge


Duke Encounters
Encounters
Search
My Account
Log out
Edit My Account
arthur@dent.com
User Data
Firstname
Arthur
Lastname
Dent
Email
attacker@evil.com
Update Data

Duke Encounters Encounters Search My Account Log out

Edit My Account

arthur@dent.com

Account Data

Email

attacker@evil.com

Current password

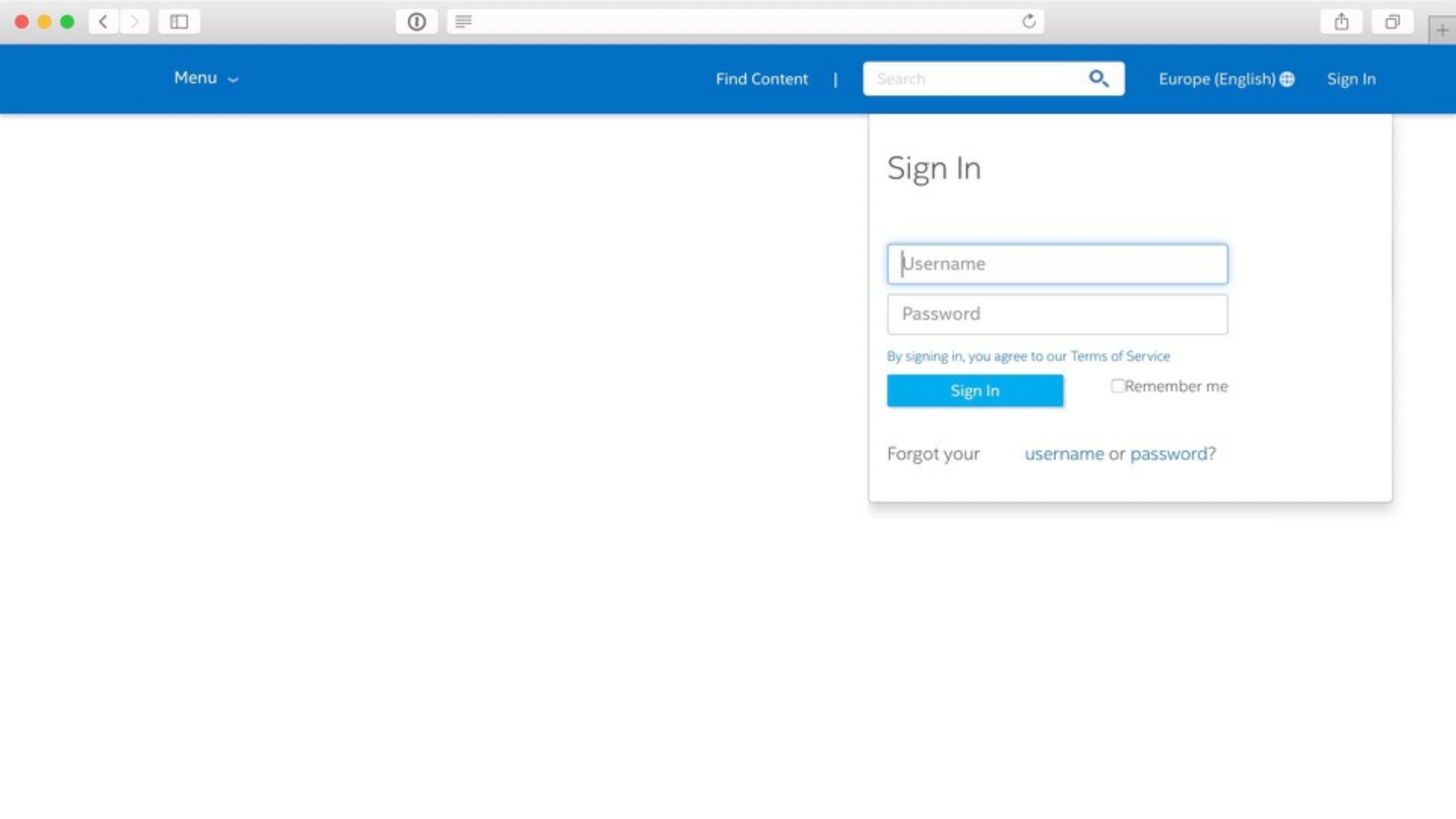
Update Account

Implement

- Disabling pasting passwords
- Delivering log-in form via HTTP

Disabling pasting into password fields

- **Does not** stop any attack
- **Does not** provide any more security
- **Does** frustrate users


Menu
Find Content
Search
Europe (English)
Sign In
Sign In
Username
Password
By signing in, you agree to our Terms of Service
Sign In
Remember me
Forgot your username or password?

HTTP log in page puts security in jeopardy

Link to dedicated HTTPS log in page

Force HTTPS for the whole page

```
@WebFilter(urlPatterns = {"/*"})
public class HSTS implements Filter {
  public void doFilter(…) {
    HttpServletResponse response =
     (HttpServletResponse) res;
    response.addHeader(
      "Strict-Transport-Security",
      "max-age=31556926");
    chain.doFilter(req, response);
  }
  // …
}
```

```
@WebFilter(urlPatterns = {"/*"})
public class HSTS implements Filter {
  public void doFilter(…) {
    HttpServletResponse response =
     (HttpServletResponse) res;
    response.addHeader(
      "Strict-Transport-Security",
      "max-age=31556926");
    chain.doFilter(req, response);
  }
  // …
}
```

```
@WebFilter(urlPatterns = {"/*"})
public class HSTS implements Filter {
  public void doFilter(…) {
    HttpServletResponse response =
     (HttpServletResponse) res;
    response.addHeader(
      "Strict-Transport-Security",
      "max-age=31556926");
    chain.doFilter(req, response);
  }
  // …
}
```

```
@WebFilter(urlPatterns = {"/*"})
public class HSTS implements Filter {
  public void doFilter(…) {
    HttpServletResponse response =
     (HttpServletResponse) res;
    response.addHeader(
      "Strict-Transport-Security",
      "max-age=31556926");
    chain.doFilter(req, response);
  }
  // …
}
```

```
@WebFilter(urlPatterns = {"/*"})
public class HSTS implements Filter {
  public void doFilter(…) {
    HttpServletResponse response =
     (HttpServletResponse) res;
    response.addHeader(
      "Strict-Transport-Security",
      "max-age=31556926");
    chain.doFilter(req, response);
  }
  // …
}
```

HSTS stops any insecure communication

Requires HTTPS connection

No effect on HTTP connections

All resources via HTTPS

Includes scripts, images, …

Requires valid certificate

No self-signed certificates any more

Implement

Not logging security events

Logging forensics after an event

Log in and log out is a security event

OWASP Security Logging

SECURITY_SUCCESS
Successful security check (e.g. successful login)

SECURITY_FAILURE
Failed security check (e.g. failed login)

SECURITY_AUDIT
Record security events for audit (e.g. account edited)

Use an always active log level or

separate log file

```
log.warn(
  SecurityMarkers.SECURITY_AUDIT,
  "User {} has edited his account“,
  username);
```

```
log.warn(
   SecurityMarkers.SECURITY_AUDIT,
   "User {} has edited his account“,
   username);
```

```
log.warn(
   SecurityMarkers.SECURITY_AUDIT,
   "User {} has edited his account“,
   username);
```

Implement

- Skipping session configuration
- Keeping session id after log-in

```
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-war-plugin</artifactId>
 <version>2.6</version>
 <configuration>
  <failOnMissingWebXml>
    false
  </failOnMissingWebXml>
 </configuration>
</plugin>
```

```
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-war-plugin</artifactId>
 <version>2.6</version>
 <configuration>
  <failOnMissingWebXml>
    false
  </failOnMissingWebXml>
 </configuration>
</plugin>
```

web.xml is a rich source for security configuration

```
<web-app ... version="3.1">
 <session-config>
  <!-- idle timeout after session expires -->
  <session-timeout>30</session-timeout>
  <cookie-config>
   <!-- prevent session id script access -->
   <http-only>true</http-only>
   <!-- transfer cookie via https only -->
   <secure>true</secure>
  </cookie-config>
  <!-- session id in cookie, not URL -->
  <tracking-mode>COOKIE</tracking-mode>
 </session-config>
</web-app>
```

```
<web-app ... version="3.1">
 <session-config>
  <!-- idle timeout after session expires -->
  <session-timeout>30</session-timeout>
  <cookie-config>
   <!-- prevent session id script access -->
   <http-only>true</http-only>
   <!-- transfer cookie via https only -->
   <secure>true</secure>
  </cookie-config>
  <!-- session id in cookie, not URL -->
  <tracking-mode>COOKIE</tracking-mode>
 </session-config>
</web-app>
```

```
<web-app ... version="3.1">
 <session-config>
  <!-- idle timeout after session expires -->
  <session-timeout>30</session-timeout>
  <cookie-config>
   <!-- prevent session id script access -->
   <http-only>true</http-only>
   <!-- transfer cookie via https only -->
   <secure>true</secure>
  </cookie-config>
  <!-- session id in cookie, not URL -->
  <tracking-mode>COOKIE</tracking-mode>
 </session-config>
</web-app>
```

```
<web-app ... version="3.1">
 <session-config>
  <!-- idle timeout after session expires -->
  <session-timeout>30</session-timeout>
  <cookie-config>
   <!-- prevent session id script access -->
   <http-only>true</http-only>
   <!-- transfer cookie via https only -->
   <secure>true</secure>
  </cookie-config>
  <!-- session id in cookie, not URL -->
  <tracking-mode>COOKIE</tracking-mode>
 </session-config>
</web-app>
```

```
<web-app ... version="3.1">
 <session-config>
  <!-- idle timeout after session expires -->
  <session-timeout>30</session-timeout>
  <cookie-config>
   <!-- prevent session id script access -->
   <http-only>true</http-only>
   <!-- transfer cookie via https only -->
   <secure>true</secure>
  </cookie-config>
  <!-- session id in cookie, not URL -->
  <tracking-mode>COOKIE</tracking-mode>
 </session-config>
</web-app>
```

User usually receives a session id when entering web application


4E01EF46D8446D1C
10CB5C08EDA69DD1

Session hijacking

Attacker steals the session id

Session fixation

Attacker dictates the session id

Have an always visible logout button

```
<form th:action="@{/logout}" method="post">
 <button type="submit">Log out</button>
</form>
```

```
<form th:action="@{/logout}" method="post">
 <button type="submit">Log out</button>
</form>
```

```
<form action="/logout" method="post">
  <input type="hidden"
    name="${_csrf.parameterName}"
    value="${_csrf.token}"/>
    <input type="submit" value="Logout"/>
</form>
```

```
<form action="/logout" method="post">
  <input type="hidden"
    name="${_csrf.parameterName}"
    value="${_csrf.token}"/>
    <input type="submit" value="Logout"/>
</form>
```

Limit session duration

web.xml

Force HTTPS

HSTS

Change session id after log in

```
@WebServlet
public class Login extends HttpServlet {
 protected void doPost(HttpServletRequest
  request, HttpServletResponse response) {
  // ...
  request.changeSessionId();
  // ...
 }
}
```

```
@WebServlet
public class Login extends HttpServlet {
 protected void doPost(HttpServletRequest
  request, HttpServletResponse response) {
  // ...
  request.changeSessionId();
  // ...
 }
}
```

```
@WebServlet
public class Login extends HttpServlet {
 protected void doPost(HttpServletRequest
  request, HttpServletResponse response) {
  // ...
  request.changeSessionId();
  // ...
 }
}
```

Invalidate session after log out

```
@WebServlet
public class Logout extends HttpServlet {
 protected void doPost(HttpServletRequest
  request, HttpServletResponse response) {
  // ...
  request.getSession().invalidate();
  // ...
 }
}
```

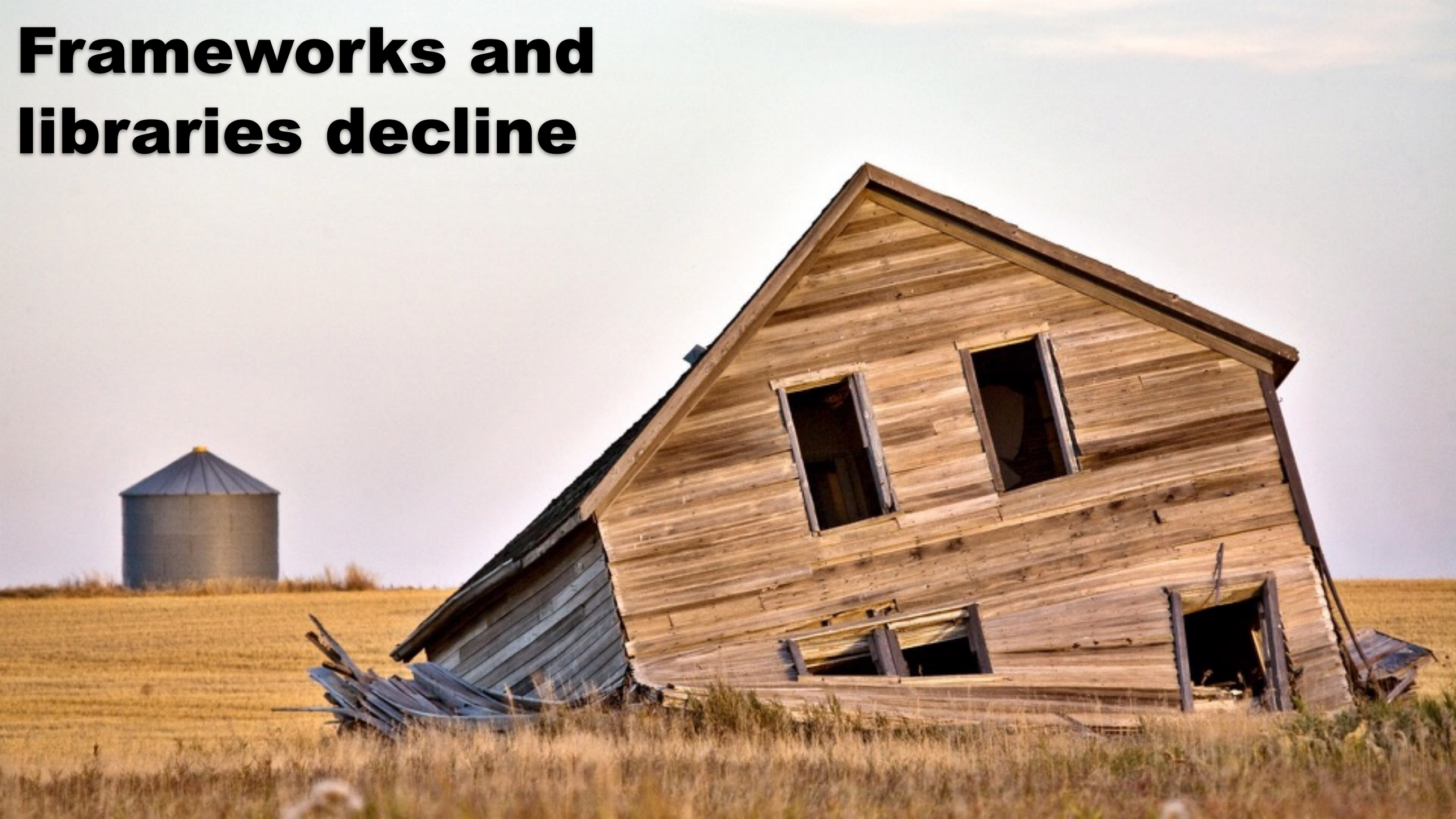
```
@WebServlet
public class Logout extends HttpServlet {
 protected void doPost(HttpServletRequest
  request, HttpServletResponse response) {
  // ...
  request.getSession().invalidate();
  // ...
 }
}
```

Demo

Maintain

Using outdated libraries

Frameworks and libraries decline



```
Marvin:duke-encounters dos$ dependency-check --project DukeEncounters --scan target/dependency/
[INFO] Checking for updates
[INFO] NVD CVE requires several updates; this could take a couple of minutes.
[INFO] Download Started for NVD CVE - 2002
[INFO] Download Started for NVD CVE - 2004
[INFO] Download Started for NVD CVE - 2003
[INFO] Download Complete for NVD CVE - 2004  (8273 ms)
[INFO] Download Started for NVD CVE - 2005
[INFO] Processing Started for NVD CVE - 2004
[INFO] Download Complete for NVD CVE - 2003  (9816 ms)
[INFO] Download Started for NVD CVE - 2006
[INFO] Processing Complete for NVD CVE - 2004  (2867 ms)
[INFO] Processing Started for NVD CVE - 2003
[INFO] Processing Complete for NVD CVE - 2003  (697 ms)
[INFO] Download Complete for NVD CVE - 2005  (12099 ms)
[INFO] Processing Started for NVD CVE - 2005
[INFO] Download Started for NVD CVE - 2007
[INFO] Download Complete for NVD CVE - 2002  (21154 ms)
[INFO] Download Started for NVD CVE - 2008
[INFO] Processing Complete for NVD CVE - 2005  (2565 ms)
[INFO] Processing Started for NVD CVE - 2002
[INFO] Processing Complete for NVD CVE - 2002  (1349 ms)
[INFO] Download Complete for NVD CVE - 2007  (16334 ms)
[INFO] Processing Started for NVD CVE - 2007
[INFO] Download Started for NVD CVE - 2009
[INFO] Download Complete for NVD CVE - 2006  (26902 ms)
[INFO] Download Started for NVD CVE - 2010
[INFO] Processing Complete for NVD CVE - 2007  (2594 ms)
[INFO] Processing Started for NVD CVE - 2006
[INFO] Processing Complete for NVD CVE - 2006  (2445 ms)
```

DEPENDENCY-CHECK

Project: DukeEncounters

Scan Information (show all):

- *dependency-check version*: 1.3.1
- *Report Generated On*: Okt 17, 2015 at 09:52:41 MESZ
- *Dependencies Scanned*: 96
- *Vulnerable Dependencies*: 1
- *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
tomcat-embed-core-8.0.28.jar	cpe:/a:apache:tomcat:8.0.28 cpe:/a:apache_tomcat:apache_tomcat:8.0.28	org.apache.tomcat.embed:tomcat-embed-core:8.0.28	High	4	LOW	16

Dependencies

```
<reporting>
 <plugins><plugin>
  <groupId>org.owasp</groupId>
  <artifactId>dependency-check-maven</artifactId>
  <version>1.3.1</version>
  <reportSets>
   <reportSet>
    <reports>
     <report>aggregate</report>
    </reports>
    </reportSet>
   </reportSets>
  </plugin></plugins>
</reporting>
```

Post-build Actions

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

100% 0%

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
	5	2	3	5

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

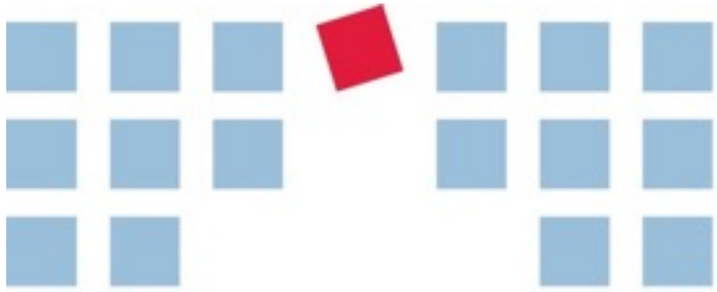
Summary

Plan security with threat modeling

Think (like an attacker) during implementation

Keep 3rd party libraries up-to-date

Enjoy secure programming



bridging IT

Koenigstr. 42
70173 Stuttgart
Germany

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

Blog blog.dominikschadow.de
Twitter @dschadow

Demo Projects
github.com/dschadow/JavaSecurity

HTTP Strict Transport Security RFC
tools.ietf.org/html/rfc6797

Microsoft Threat Modeling Tool
www.microsoft.com/en-us/sdl/adopt/threatmodeling.aspx

Mozilla SeaSponge
air.mozilla.org/mozilla-winter-of-security-seasponge-a-tool-for-easy-threat-modeling

OWASP Dependency Check
www.owasp.org/index.php/OWASP_Dependency_Check

OWASP Security Logging
www.owasp.org/index.php/OWASP_Security_Logging_Project

Spring Security
projects.spring.io/spring-security

Pictures
www.dreamstime.com