Seguridad de aplicaciones Java/JakartaEE con OWASP Top 10

Víctor Orozco - @tuxtor

17 de mayo de 2020

Academik

ACADEMIK

Principios básicos

- No existe un framework generico para hacer ”seguridad 360”
- Seguridad = Balance entre necesidad de negocio/tecnología
- En Java hay n formas de hacer lo mismo
- Requerimientos -> Cifrado, firmas digitales, autenticación, autorización
- Herramientas

¿Java?

- Lenguaje
- VM
- Bibliotecas/API

El conjunto es la plataforma Java

¿Java?

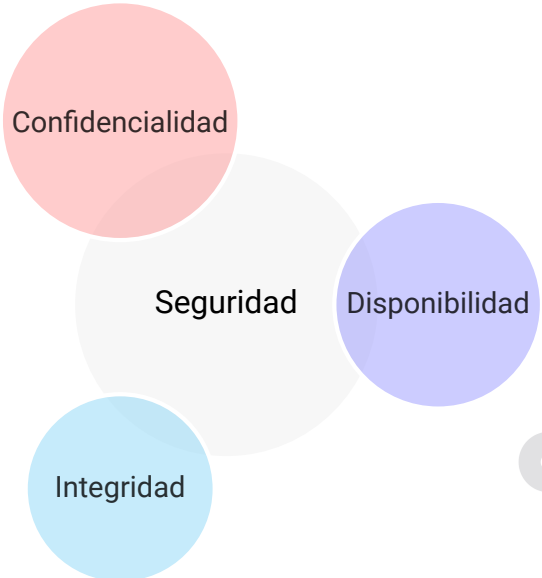
- Lenguaje
- VM
- Bibliotecas/API

El conjunto es la plataforma Java (TM)

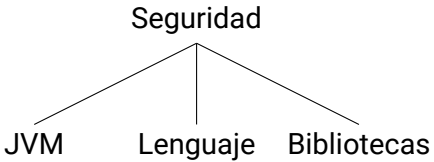
ACADEMIK

Seguridad en Java

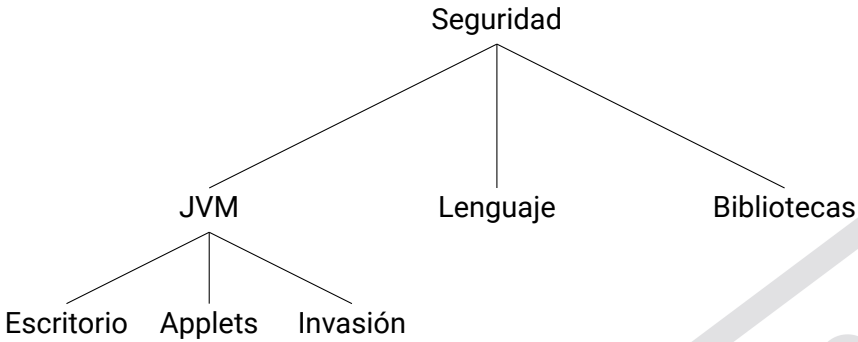
Infosec



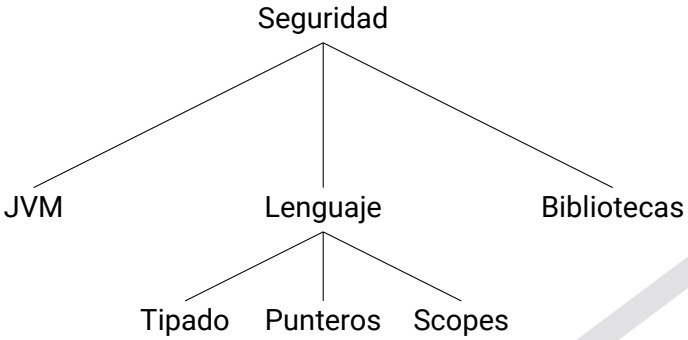
Seguridad



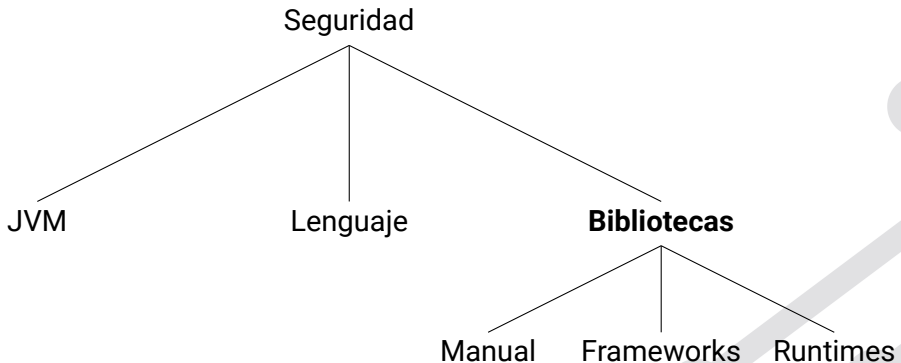
Seguridad



Seguridad



Seguridad



Bibliotecas

¿Cual?

- Apache Shiro
- Spring Security
- OACC
- Keycloak
- JGuard
- JACC
- SoteriaRI
- MicroProfile JWT

...

Render en servidor

- JSF (Icefaces, Primefaces)
- GWT
- JSP
- Servlets
- Vaadin
- Struts
- Spring MVC

Render en servidor

- JSF (Icefaces, Primefaces)
- GWT
- JSP
- Servlets
- Vaadin
- Struts
- Spring MVC

Render en cliente

- Angular
- React
- Knockout (Oracle JET)
- Vue

Tecnologias Web

Render en servidor

- JSF (Icefaces, Primefaces)
- GWT
- JSP
- Servlets
- Vaadin
- Struts
- Spring MVC

Render en cliente

- Angular
- React
- Knockout (Oracle JET)
- Vue

Servicios

- SOAP
- Rest
- RMI

Tecnología

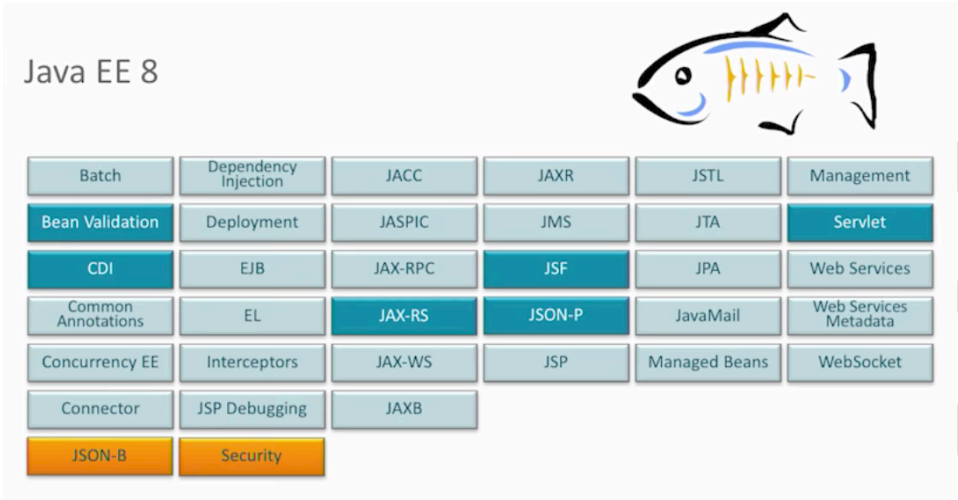

ORACLE
PostgreSQL
MySQL
mongoDB


solaris
redhat
debian
ORACLE
LINUX
CentOS
Java
ENTERPRISE
EDITION
JBoss
by Red Hat
GlassFish
WildFly
Coquoro

JavaEE 8

- Mejor integración de JSF con CDI
- Mejor integración de JMS con CDI
- HTTP/2
- JSON-B
- Security
- JAX-RS Reactivo

JavaEE 8



EE vs OWASP Top 10

Advertencia

- Visto en N desarrollos
- Un punto de inicio
- Informar

JavaEE

- A1-Injection
- A2-Broken Authentication and Session Management
- A3-Sensitive Data Exposure
- A4-XML External Entities
- A5-Broken Access Control
- A6-Security Misconfiguration
- A7-Cross-Site Scripting (XSS)
- A8-Insecure deserialization
- A9-Using Components with Known Vulnerabilities
- A10-Insufficient Logging y Monitoring

`https://owasp.org/www-project-top-ten/`

- **A1-Injection**
- **A2-Broken Authentication and Session Management**
- **A3-Sensitive Data Exposure**
- A4-XML External Entities
- **A5-Broken Access Control**
- **A6-Security Misconfiguration**
- A7-Cross-Site Scripting (XSS)
- **A8-Insecure deserialization**
- **A9-Using Components with Known Vulnerabilities**
- A10-Insufficient Logging y Monitoring

`https://www.owasp.org/index.php/Top_10_2017-Top_10`

JavaEE - A1 - Injection

Problemas y causas comunes

- Concatenación de Strings en SQL
- Datos mal intencionados a aplicaciones
- Manipulación data stores
- Escalar privilegios

Sugerencias

- JAMAS y NUNCA concatenar parametros
- Siempre utilizar mecanismos de `sanitizing`
- Parchar con OWASP ESAPI
- JDBC y JPA soportan de serie sanitizing si no se concatena
- Bean Validation en parametros

Peligro

```
String query = "SELECT p FROM AdmPhrase p " +
"where p.author LIKE " + autor +
"and p.phrase LIKE " + phrase;
```

Mejor

```
String query = "SELECT p FROM AdmPhrase p " +
    "where p.author LIKE :author " +
    "and p.phrase LIKE :phrase";

return em.createQuery(query, AdmPhrase.class)
    .setParameter("author", author)
    .setParameter("phrase", phrase)
    .getResultList();
```

JavaEE - A2-Broken Authentication and Session Management

Problemas y causas comunes

- Implementación de solución manual vs frameworks
- Falta de políticas
- Entrenamiento en la plataforma
- Comunicación y/o autenticación via http

Sugerencias

- Forzar https
- Utilizar adecuadamente los ciclos de vida de la plataforma (singleton != stateless != statefull) y cache
- No implementar en base a interceptores unicamente

App rest normal

```
1 public class DemoinfosecRestApplication extends Application {}
```

Activar mecanismo de autenticación (MicroProfile JWT)

```
1 @LoginConfig(authMethod = "MP-JWT")
2 @DeclareRoles({RolesEnum.Constants.MOBILE_VALUE, RolesEnum.
     Constants.WEB_VALUE})
3 public class DemoinfosecRestApplication extends Application {}
```

JavaEE - A3-Sensitive data exposure

Problemas y causas comunes

- Guardar datos sin cifrar
- Datos con cifrado ”debil”, AKA cifrado propio
- Transmitir credenciales via http
- Transmisión de excepciones completas a front-end

Sugerencias

- Identificar con un checklist los datos sensitivos
- Evitar cifrado de dos vías a menos que sea necesario
- Evitar transmisión de llaves
- Verificar código auto generado (excepciones)

JavaEE - A5-Broken Access Control

Problemas y causas comunes

- Implementación de solución manual vs frameworks
- Falta de políticas
- Escalar privilegios
- Comunicación y/o autenticación via http

Sugerencias

- Forzar https
- Implementación RBAC de application server
- Implementación RBAC de framework
- Entender el modelo de JAAS, SoteriaRI y MicroProfile security

JavaEE - A5-Broken Access Control

Endpoint normal

```
1  @GET
2  @Path("/{id:[0-9][0-9]*}")
3  public AdmPhrase findById(@PathParam("id") Long id) {
4      return admPhraseRepository.findById(id);
5  }
```

Endpoint con RBAC

```
1  @GET
2  @Path("/{id:[0-9][0-9]*}")
3  @RolesAllowed({RolesEnum.Constants.MOBILE_VALUE, RolesEnum.
       Constants.WEB_VALUE})
4  public AdmPhrase findById(@PathParam("id") Long id) {
5      return admPhraseRepository.findById(id);
6  }
```

JavaEE - A6-Security Misconfiguration

Problemas y causas comunes

- Configuración por defecto de applicaction server
- Configuración por defecto de SO
- Configuración `relajada` de capa de transporte

Sugerencias

- Configurar siempre el SO destino
- Proteger y actualizar runtime
- Firewall
- RBAC
- Evitar certificados autofirmados en entornos no controlados
- JVM tipo `server`

JavaEE - A8-Insecure deserialization

Problemas y causas comunes

- Self made frameworks
- No validation

Sugerencias

- Evaluar si no vale la pena utilizar un software listo
- Bean validation
- Sanitización

JavaEE - A9-Using components with known vulnerabilities

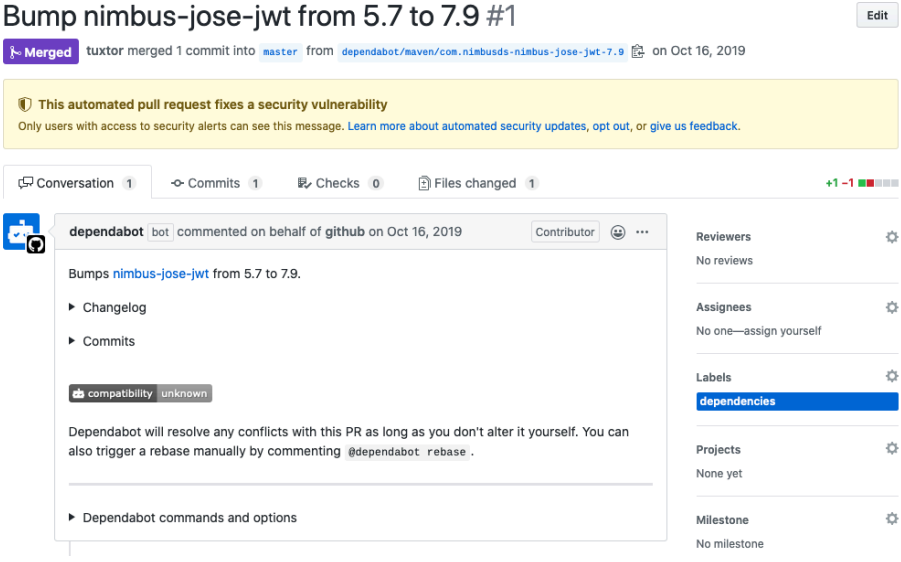
Problemas y causas comunes

- Difícil dar seguimiento a los lanzamientos
- Frameworks muy nuevos o muy viejos
- No seguir las notas del lanzamiento

Sugerencias

- Actualizar el app server con el calendario de lanzamiento
- Suscripción a mailing list/foros
- Servicios tipo Bintray, GitHub Security
- Servicios de análisis estático (Sonar)

JavaEE - A9-Using components with known vulnerabilities



JavaEE - Bonus-APIs sin proteger

Problemas y causas comunes

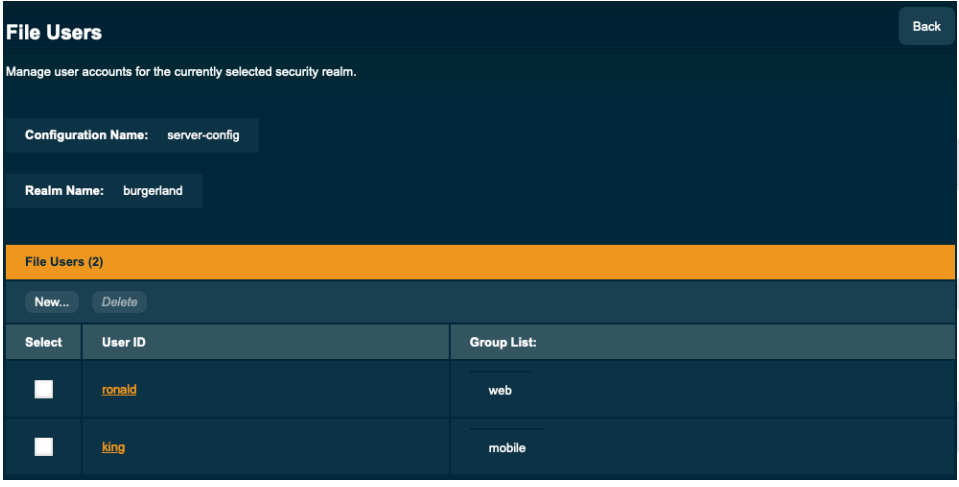
- Combinación de broken auth, broken access control o security misconfiguration
- No hay nada
- Especialmente grave en rich clients

Sugerencias

- Verificar código auto generado
- Programar en modo `deny all`

Demo

Demo Payara 5



Demo Payara 5

- Proveedor de tokens https://github.com/tuxtor/microjwt-provider/
- API protegida https://github.com/tuxtor/demoinfosec-service-b/

Víctor Orozco

- vorozco@nabenik.com
- @tuxtor
- http://vorozco.com
- http://tuxtor.shekalug.org

ACADEMIK

Escríbenos a cursos@academik.io

www.academik.io