



Desarrollo seguro de aplicaciones y servicios Web en ambiente Java:

mejores prácticas en la Dirección de
Sistemas de la DGPE

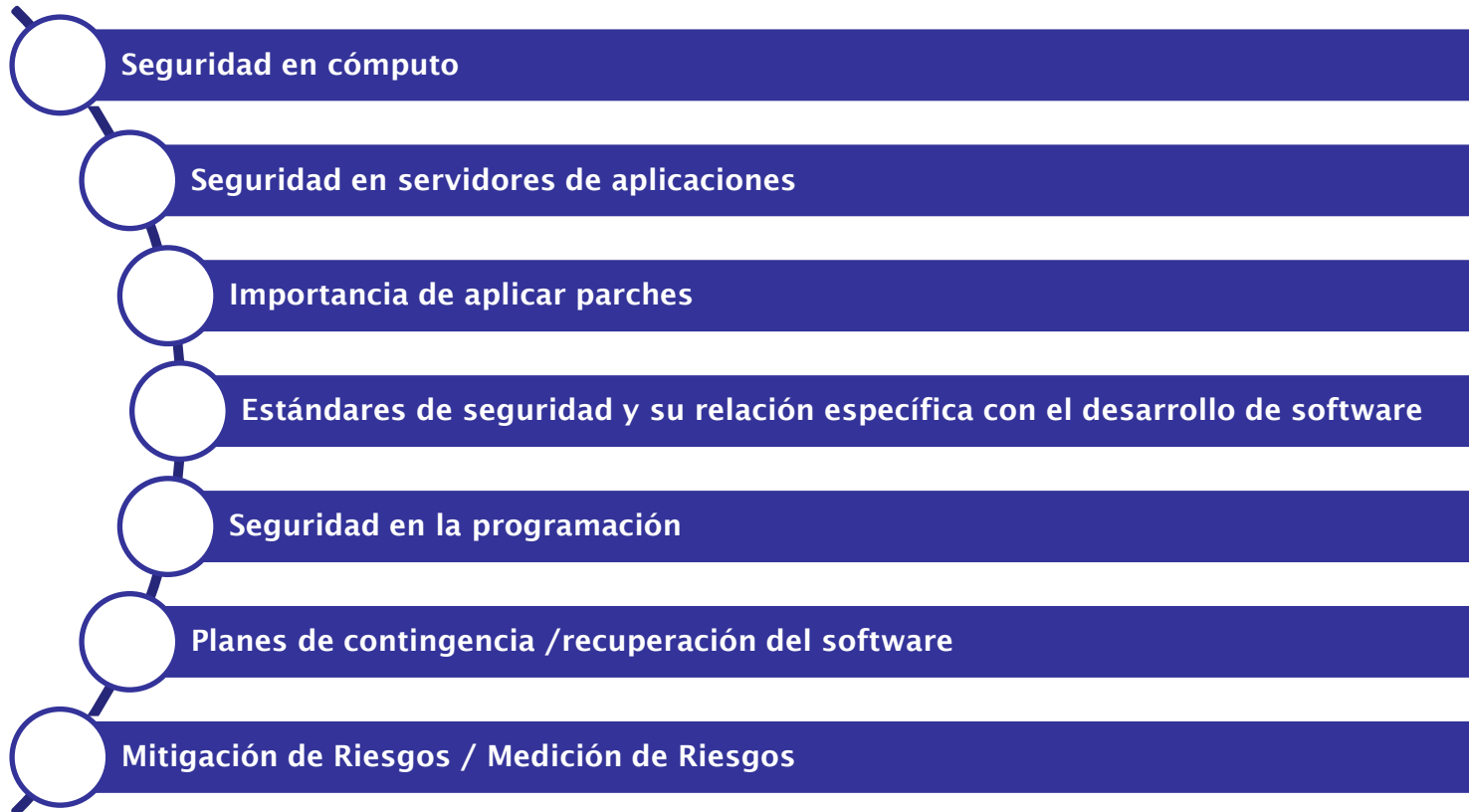


Presenta: Israel Ortega Cuevas
19/Mayo/2015





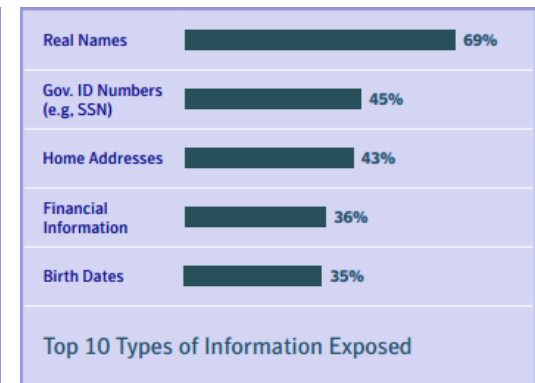
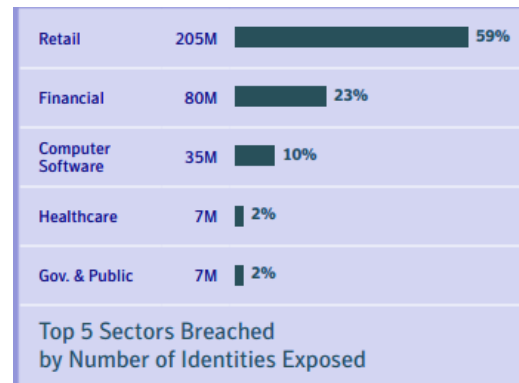
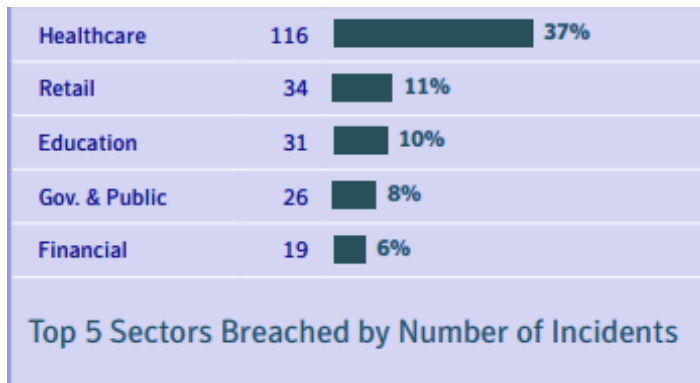
Contenido





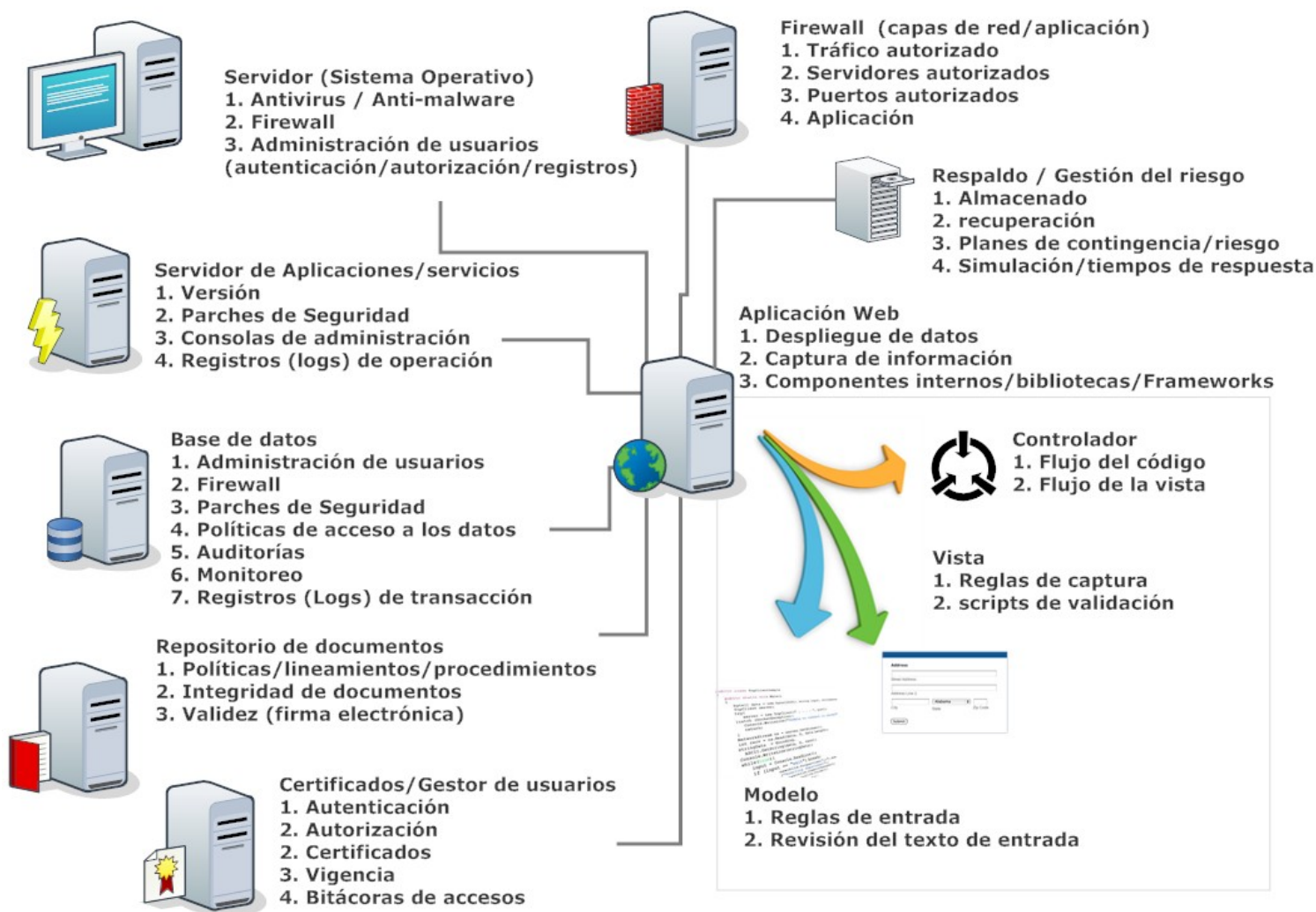
Seguridad en cómputo

- 312 brechas de seguridad
 - 348 Millones de identidades expuestas
 - 1 Millón de identidades por brecha



IS_TR 20 de Symantec

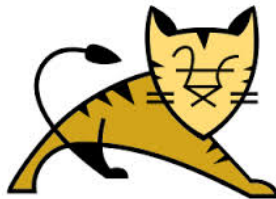
https://www4.symantec.com/mktginfo/whitepaper/ISTR/21347932_GA-internet-security-threat-report-volume-20-2015-social_v2.pdf



Vista de los elementos involucrados en la seguridad de una aplicación Web

Seguridad en Servidores de Aplicaciones

- Siempre tener la última versión



Verificar firmas (signatures) md5 o sha1 checksums

MD5 & SHA1
Checksum



Seguridad en Servidores de Aplicaciones

- Los servidores siempre deben estar detrás de un firewall

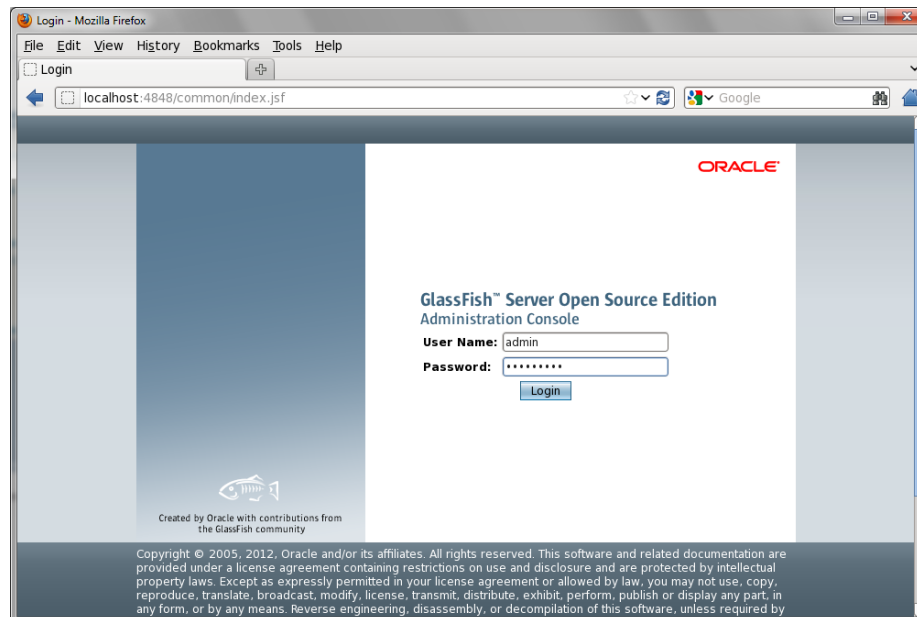


- Los servidores de base de datos preferiblemente no deben conectarse con el exterior del firewall o implementar firewall interno



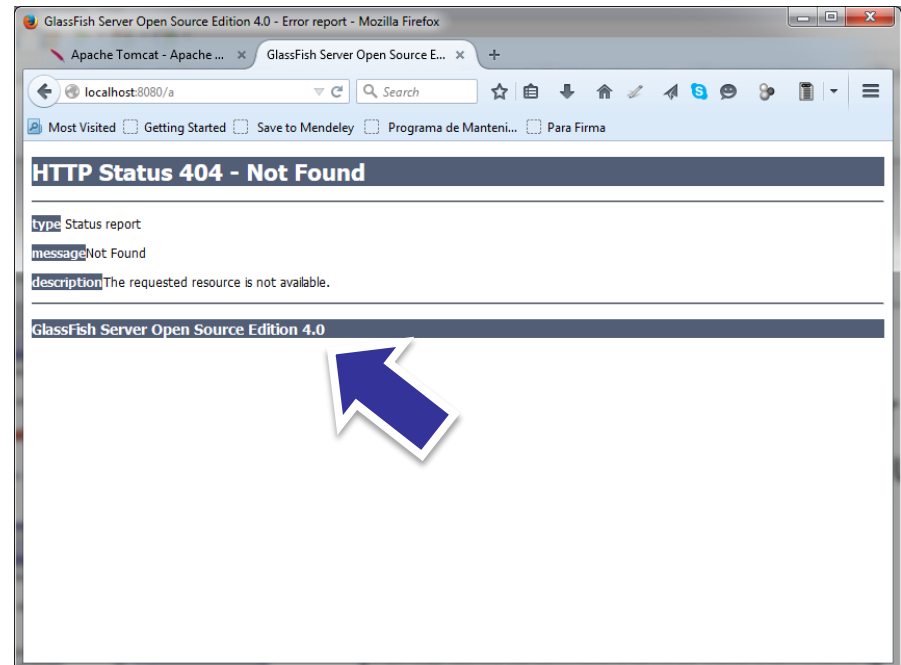
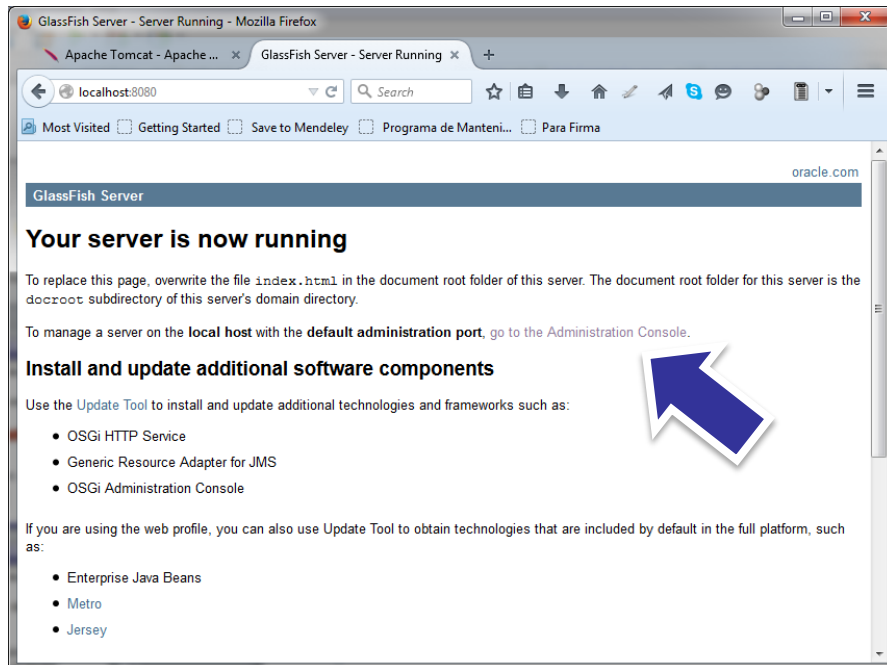
Seguridad en Servidores de Aplicaciones

- El puerto de administración únicamente debe estar disponible en el equipo cliente o equipos restringidos



Seguridad en Servidores de Aplicaciones

- No se debe proporcionar ninguna información del servidor utilizado





Consideraciones servidores JAVA

- Community edition (open source)
 - Licencias ilimitadas
 - Sin soporte técnico a corto plazo
 - Las actualización son liberadas por una comunidad de desarrolladores diversa
- Con soporte Comercial y Propietarios
 - Por servidor
 - Por organización

Consideraciones servidores JAVA

- Como funciona la incorporación de funcionalidades (solución de vulnerabilidades) en proyectos.

Ejemplo de GitHub Flow*

0. Detección de la falla



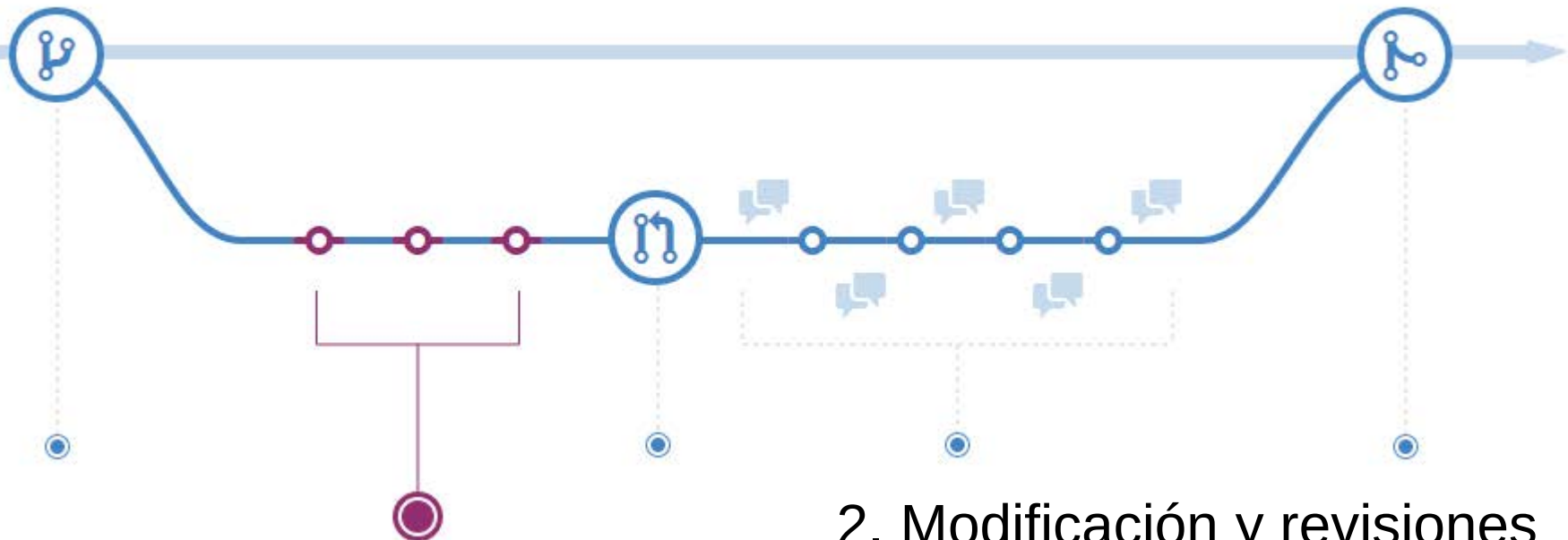
1. Creación de una rama del proyecto (Branch)

*<https://guides.github.com/introduction/flow/index.html>

Consideraciones servidores JAVA

- Como funciona la incorporación de funcionalidades (solución de vulnerabilidades) en proyectos.

Ejemplo de GitHub Flow*



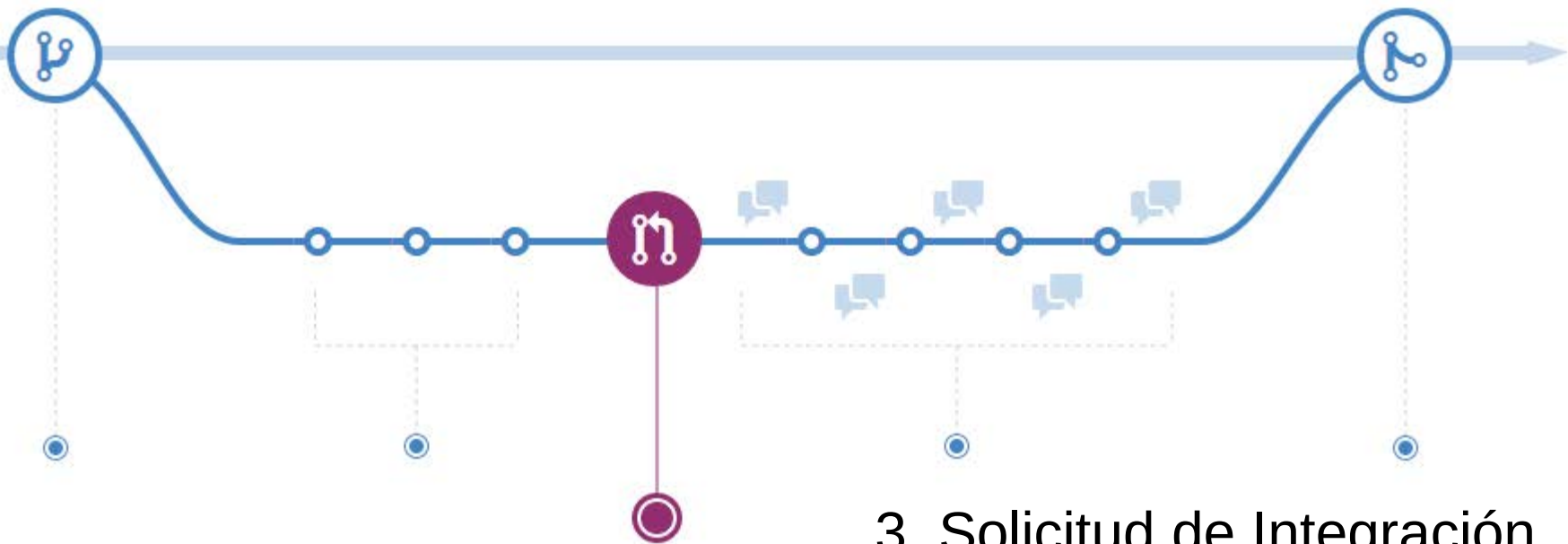
2. Modificación y revisiones
(Commits)

*<https://guides.github.com/introduction/flow/index.html>

Consideraciones servidores JAVA

- Como funciona la incorporación de funcionalidades (solución de vulnerabilidades) en proyectos.

Ejemplo de GitHub Flow*



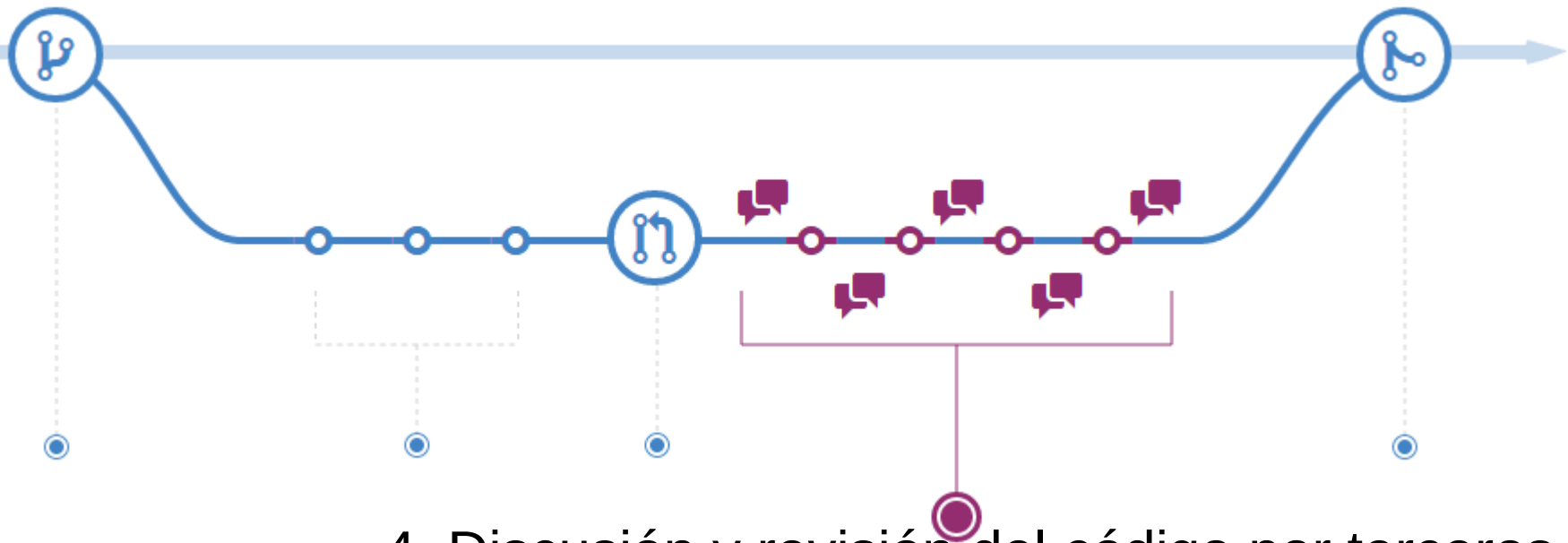
3. Solicitud de Integración
(pull request)

*<https://guides.github.com/introduction/flow/index.html>

Consideraciones servidores JAVA

- Como funciona la incorporación de funcionalidades (solución de vulnerabilidades) en proyectos.

Ejemplo de GitHub Flow*



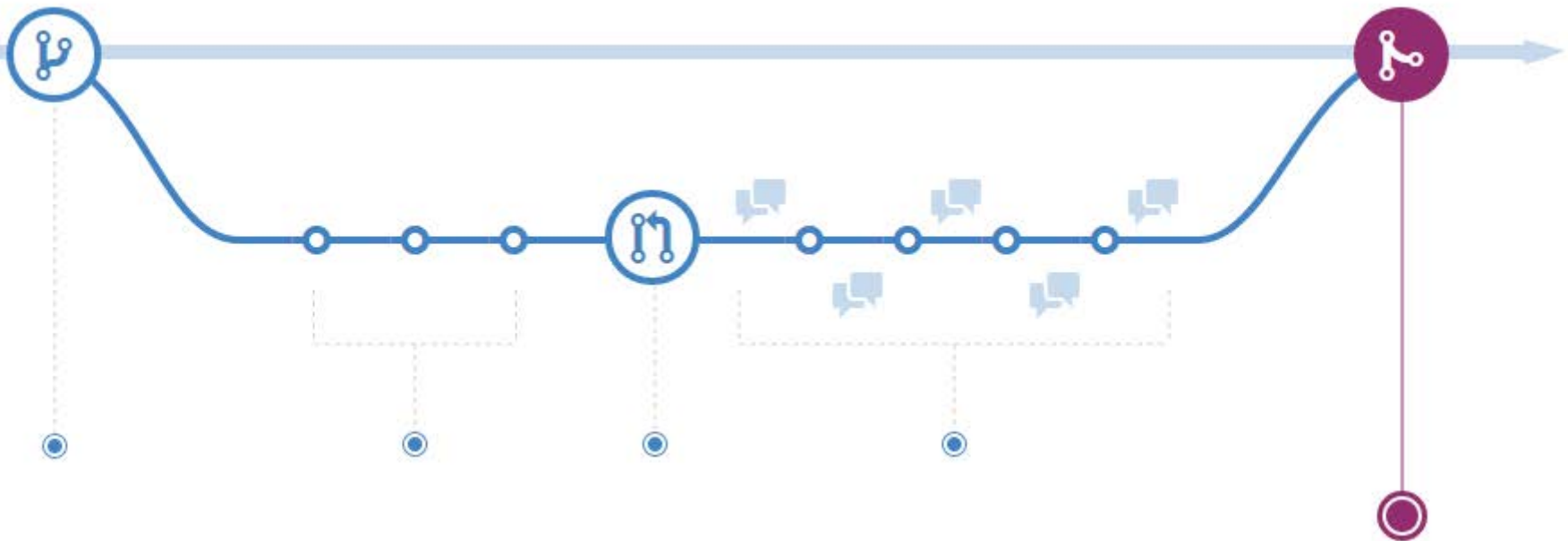
4. Discusión y revisión del código por terceros
(Discuss and review)

*<https://guides.github.com/introduction/flow/index.html>

Consideraciones servidores JAVA

- Como funciona la incorporación de funcionalidades (solución de vulnerabilidades) en proyectos.

Ejemplo de GitHub Flow*



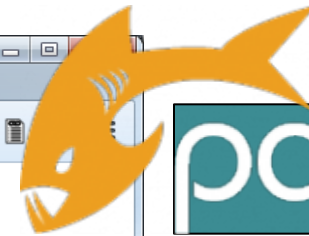
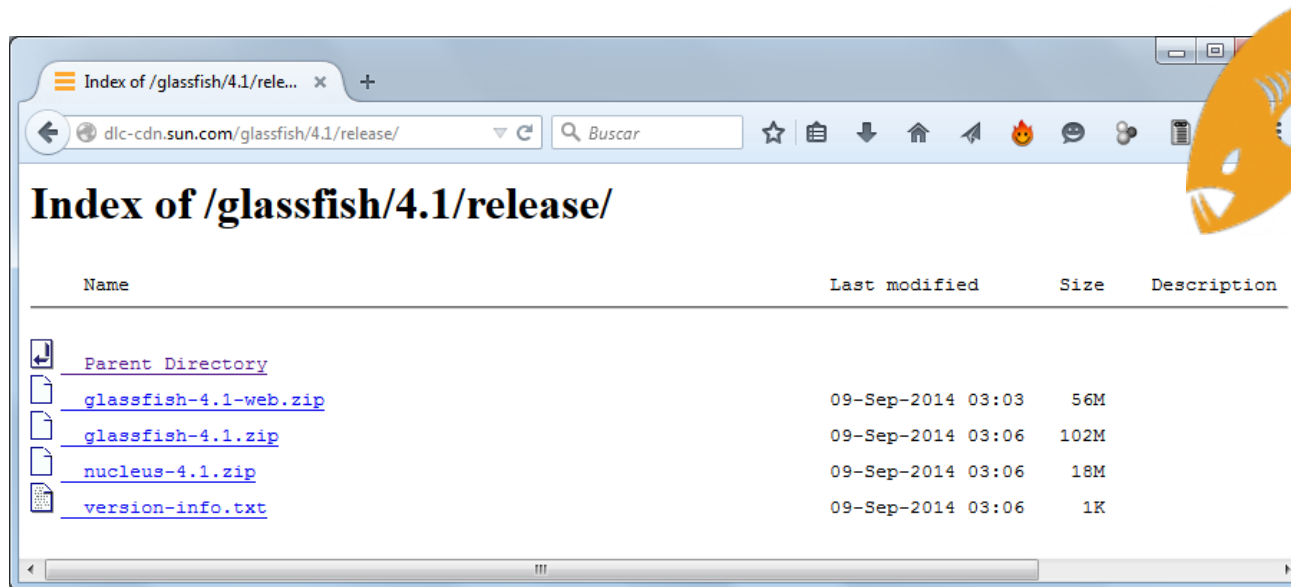
5. Aceptación de los cambios (Merge and Deploy)

*<https://guides.github.com/introduction/flow/index.html>



Consideraciones servidores JAVA

¿Qué situación tenemos con Glassfish?



Fuente: <http://blog.servidoresdeaplicaciones.com/tag/glassfish/>



Consideraciones servidores JAVA

The screenshot shows a web browser window with the address bar displaying `www.payara.co.uk/release_notes`. The browser has three tabs open: "Release Notes", "Colaboración e innovaci...", and "Servidores de Aplicacio...". The Payara website header features the logo and navigation links: Home, About, Services, Community, Blog, Downloads, and Contact. A secondary navigation bar highlights "Release Notes" among other options like Embedded, Multi-Language, Previous Releases, and Pre-Release Builds.

Release Notes

The release notes of previous releases can be found on our GitHub Wiki Documentation page [here](#).

Payara 4.1.152 Release Notes

Release Highlights

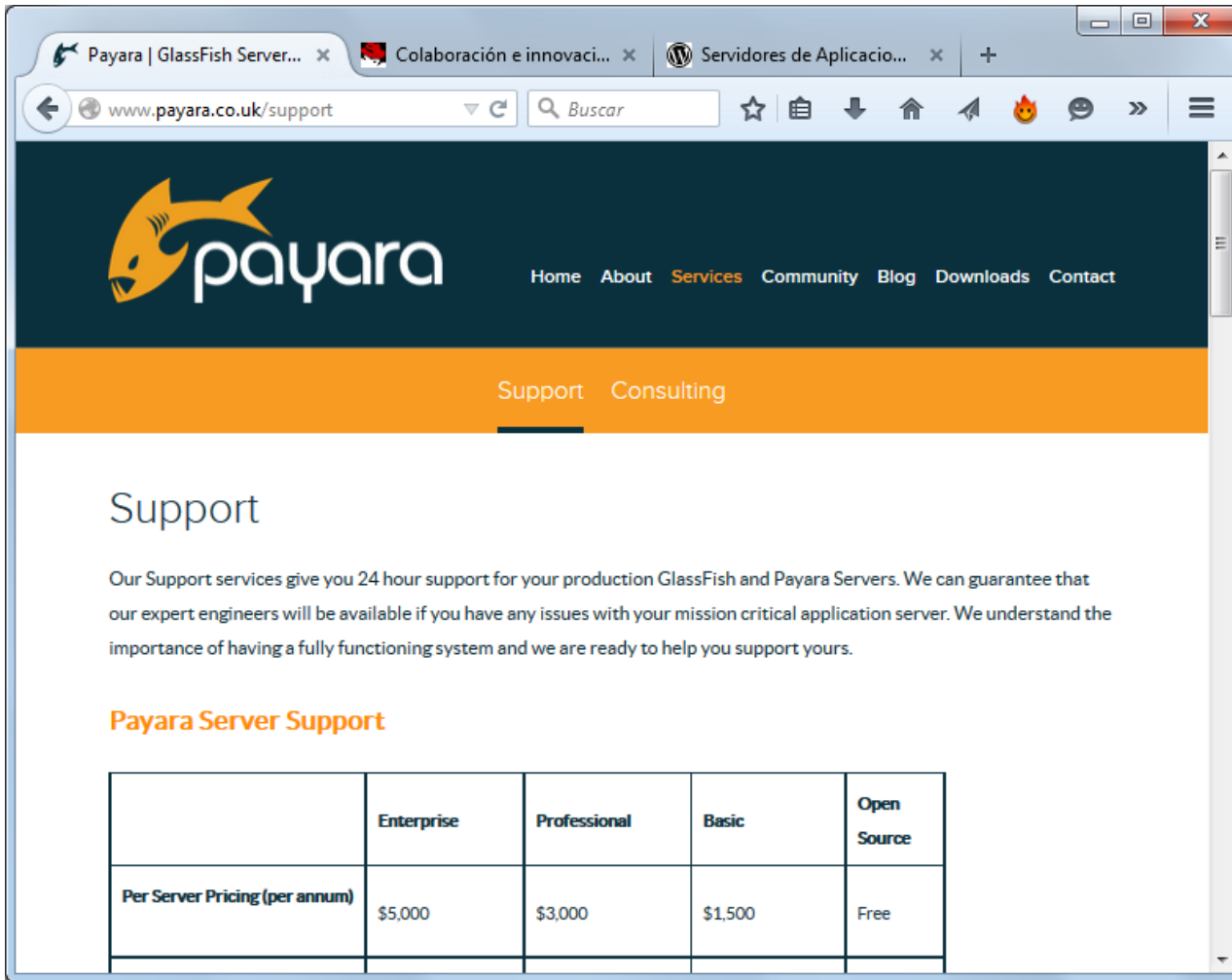
The payara-embedded-nucleus distribution has been replaced by the Payara Micro distribution. This is a light-weight distribution that has Hazelcast integrated to allow for automatic web session clustering and JCache distribution.

We have also added a separate Payara domain template, which has a few optimisations:

- Increased PermGen Size of 512m (previously 192m)



Consideraciones servidores JAVA

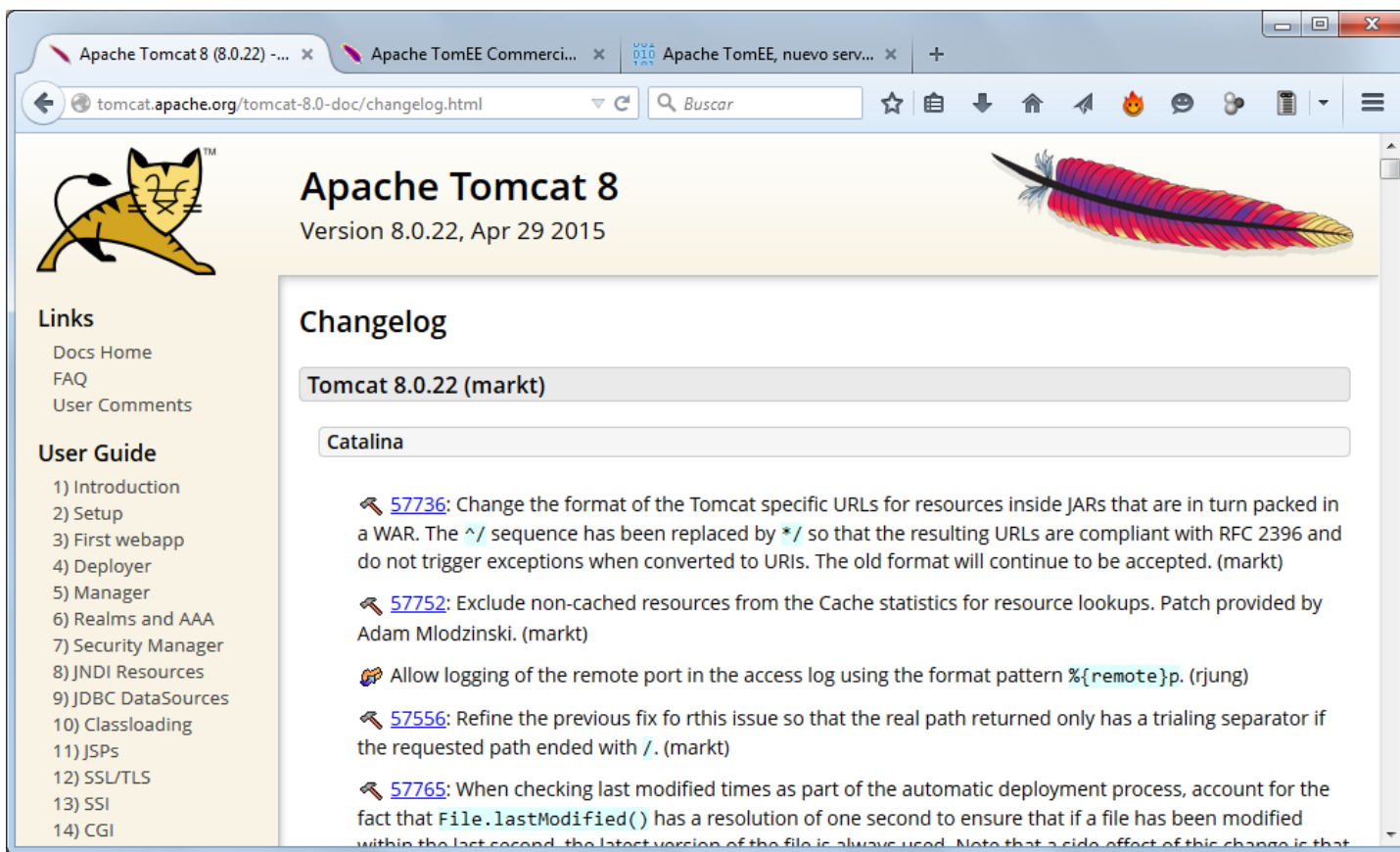


The screenshot shows a web browser window with the URL www.payara.co.uk/support. The page features the Payara logo (a stylized orange fish) and a navigation menu with links: Home, About, Services, Community, Blog, Downloads, and Contact. Below the navigation bar, there is an orange banner with the text "Support Consulting". The main content area is titled "Support" and contains a paragraph about 24-hour support services. Below this, there is a section titled "Payara Server Support" which includes a table showing pricing for different server types.

	Enterprise	Professional	Basic	Open Source
Per Server Pricing (per annum)	\$5,000	\$3,000	\$1,500	Free



Consideraciones servidores JAVA



The screenshot shows a web browser window displaying the Apache Tomcat 8.0.22 changelog page. The browser has three tabs: 'Apache Tomcat 8 (8.0.22) - ...', 'Apache TomEE Commerci...', and 'Apache TomEE, nuevo serv...'. The address bar shows 'tomcat.apache.org/tomcat-8.0-doc/changelog.html'. The page features the Tomcat logo (a yellow cat) and a colorful feather. The main content area is titled 'Apache Tomcat 8' and 'Version 8.0.22, Apr 29 2015'. A sidebar on the left contains 'Links' (Docs Home, FAQ, User Comments) and a 'User Guide' with a list of 14 items. The main content area is titled 'Changelog' and shows a section for 'Tomcat 8.0.22 (markt)' with a sub-section for 'Catalina'. The changelog entries include:

- [57736](#): Change the format of the Tomcat specific URLs for resources inside JARs that are in turn packed in a WAR. The `^/` sequence has been replaced by `*/` so that the resulting URLs are compliant with RFC 2396 and do not trigger exceptions when converted to URIs. The old format will continue to be accepted. (markt)
- [57752](#): Exclude non-cached resources from the Cache statistics for resource lookups. Patch provided by Adam Mlodzinski. (markt)
- [57756](#): Allow logging of the remote port in the access log using the format pattern `%{remote}p`. (rjung)
- [57556](#): Refine the previous fix for this issue so that the real path returned only has a trailing separator if the requested path ended with `/`. (markt)
- [57765](#): When checking last modified times as part of the automatic deployment process, account for the fact that `File.lastModified()` has a resolution of one second to ensure that if a file has been modified within the last second, the latest version of the file is always used. Note that a side effect of this change is that



Consideraciones servidores JAVA

Recomendaciones para elegir un servidor Java

Versión soporte comercial

o

Soportado por una gran comunidad de usuarios

Actualizaciones periódicas

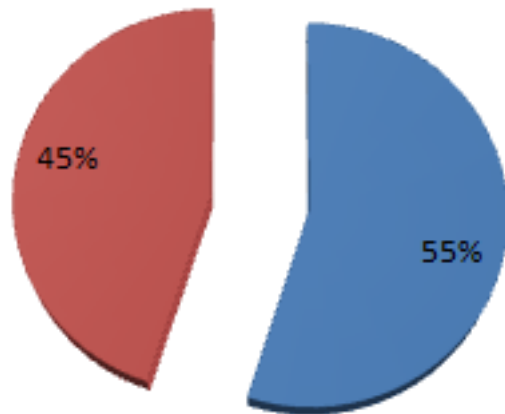
Actualización continua a componentes internos

Consideraciones de ligereza (Desarrollo/Producción)

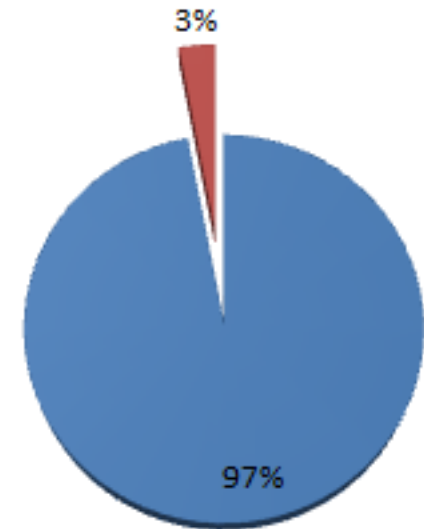
Soporte para EE, Web Services, EJB, JPA, JSF, JavaMail

Importancia de aplicar parches

Reacción ante una vulnerabilidad - Heartbleed



48 Horas después



Referencia: <https://jhalderm.com/pub/papers/heartbleed-imc14.pdf>



Importancia de aplicar parches

Reacción ante una vulnerabilidad - Heartbleed

Consecuencias:

Robo de datos de 4.5 millones de miembros de la red hospitalaria Community Health Systems

Referencia <http://www.technologyreview.es/internet/46696/>



Importancia de aplicar parches

Buenas prácticas de la Dirección de Sistemas Servidores

Aplicar parches a servidores de aplicaciones

En el caso de servidores Java, tener la última versión de JDK.



Nota: Probar siempre en un equipo alternativo al de producción

Importancia de aplicar parches

Buenas prácticas de la Dirección de Sistemas

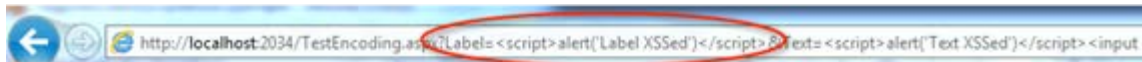
Monitorear constantemente nuevas versiones (PDF, reportadores, etc.)

Reemplazar JAR en las aplicaciones Web





Importancia de aplicar parches



This is the value of the Label:

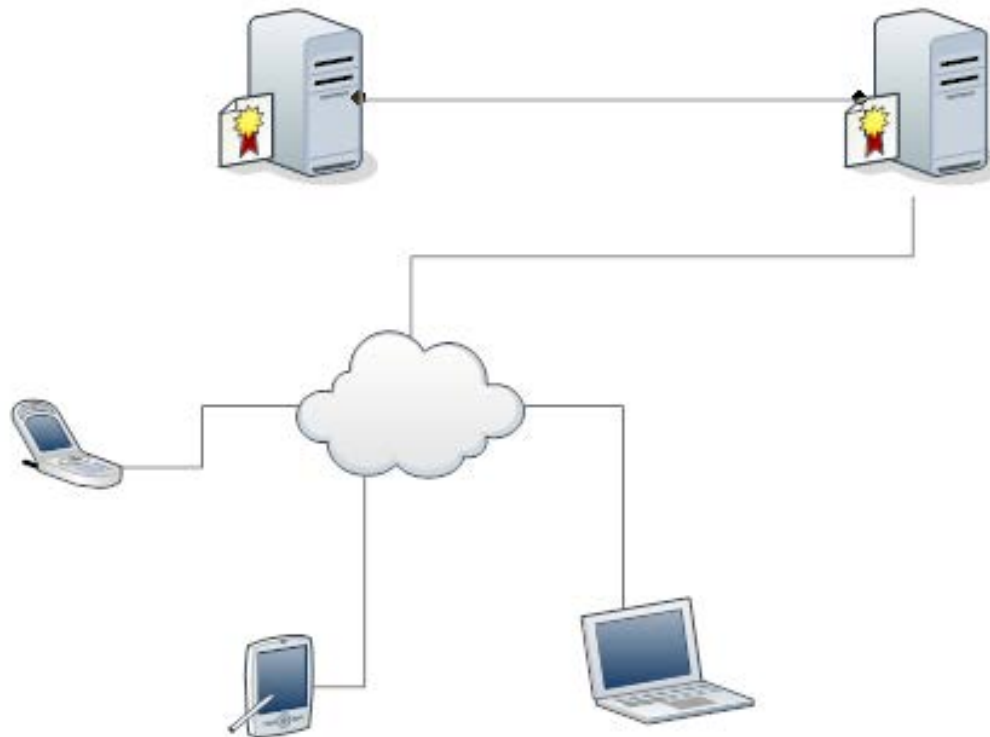


The Apache
Software Foundation
<http://www.apache.org/>

Struts™

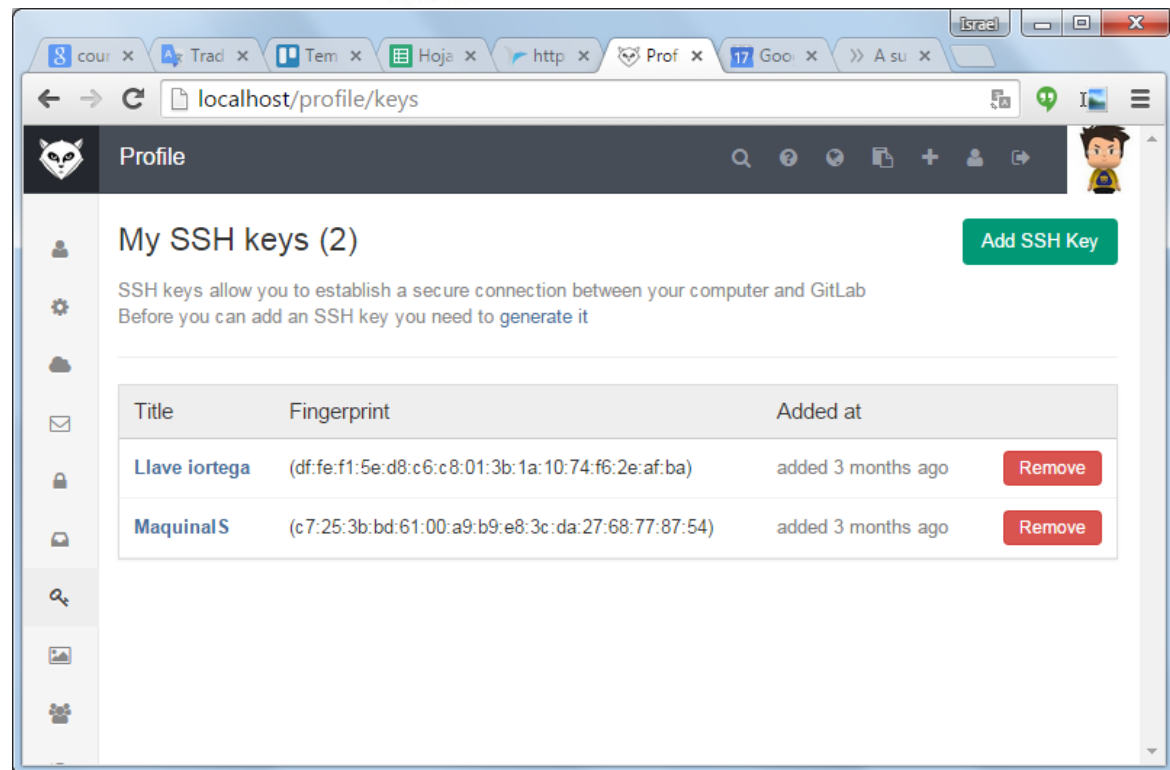
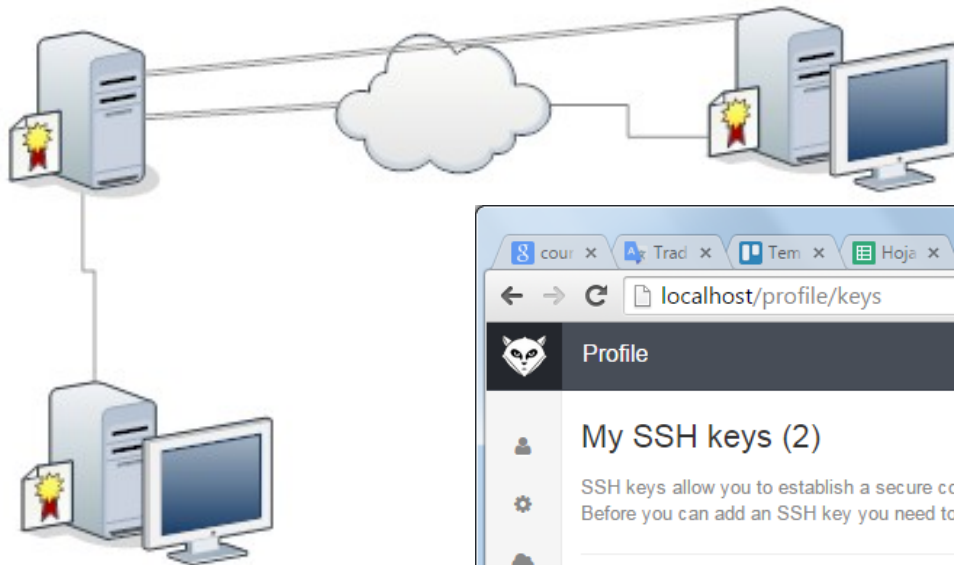
Cross-site scripting

Certificados



1. Certificados en servidor

Certificados



Profile

My SSH keys (2)

SSH keys allow you to establish a secure connection between your computer and GitLab
Before you can add an SSH key you need to [generate it](#)

Title	Fingerprint	Added at	
Llave iortega	(df:fe:f1:5e:d8:c6:c8:01:3b:1a:10:74:f6:2e:af:ba)	added 3 months ago	Remove
Maquina1 S	(c7:25:3b:bd:61:00:a9:b9:e8:3c:da:27:68:77:87:54)	added 3 months ago	Remove

[Add SSH Key](#)

2. Certificados en cliente/servidor



Estándares de seguridad y su relación específica con el desarrollo de software

ISO 27000

La serie contiene las mejores prácticas recomendadas en Seguridad de la información para desarrollar, implementar y mantener Especificaciones para los Sistemas de Gestión de la Seguridad de la Información (SGSI).





Estándares de seguridad y su relación específica con el desarrollo de software

ISO 27000

- ISO/IEC 27000 - es un vocabulario estándar para el SGSI
- ISO/IEC 27001 - es la certificación que deben obtener las organizaciones.
- ISO/IEC 27002 - Information technology - Security techniques - Code of practice for information security management.
- ISO/IEC 27003 - son directrices para la implementación de un SGSI
- ISO/IEC 27004 - son métricas para la gestión de seguridad de la información.
- ISO/IEC 27005 - trata la gestión de riesgos en seguridad de la información





Estándares de seguridad y su relación específica con el desarrollo de software

ISO 27002:2005

- 10.5.1 Information back-up
- 10.6.2 Security of network services
- 10.7.3 Information handling procedures
- 10.7.4 Security of system documentation
- 10.8.2 Exchange agreements
- 12.1.1 Security requirements analysis and specification
- 12.2.1 Input data validation
- 12.2.2 Control of internal processing
- 12.2.3 Message integrity
- 12.2.4 Output data validation
- 12.4.1 Control of operational software





Estándares de seguridad y su relación específica con el desarrollo de software

ISO 27002:2005

12.4.2 Protection of system test data

12.5.2 Technical review of applications after operating system changes

12.4.3 Access control to program source code

12.5.1 Change control procedures

12.5.3 Restrictions on changes to software packages

12.5.5 Outsourced software development

12.6.1 Control of technical vulnerabilities

13.1.1 Reporting information security events

13.1.2 Reporting security weaknesses

13.2.1 Responsibilities and procedures 13.2.2





Estándares de seguridad y su relación específica con el desarrollo de software

ISO 27002:2005

13.2.2 Learning from information security incidents

13.2.3 Collection of evidence

15.1.1 Identification of applicable legislation

15.1.2 Intellectual property rights (IPR)

15.1.3 Protection of organizational records

15.1.4 Data protection and privacy of personal information

15.1.6 Regulation of cryptographic controls

15.2.1 Compliance with security policies and standards

15.2.2 Technical compliance checking





Estándares de seguridad

Areas of concern (based on ISO 27002: 2005)	Published standards or series of standards	Standards under development	Will be revised by
<i>ISMSs</i>	Series ISO/IEC 27000 ISO/IEC 13335-1:2004	ISO/IEC CD 27003 ISO/IEC WD 27007	
<i>Security Policy</i>	ISO 27002: 2005		
<i>Organizing Information Security</i>	ISO 27002: 2005 ISO/IEC TR 14516:2002 ISO/IEC 15945:2002		
<i>Asset Management</i>	ISO 27002: 2005		
<i>Human Resources Security</i>	ISO 27002: 2005		
<i>Physical and Environmental Security</i>	ISO 27002: 2005		
<i>Communications and Operations Management</i>	Series ISO/IEC 18028	ISO/IEC WD 27007	Series ISO/IEC 27033
	Series ISO/IEC 18014		
	ISO/IEC 18043:2006		
	ISO/IEC TR 14516:2002		
	ISO/IEC 15945:2002		
<i>Access Control</i>	ISO/IEC 15816:2002		
	Series ISO/IEC 9798		
<i>Information Systems Acquisition,</i>	Series ISO/IEC 11770		ISO/IEC CD 11770-1
	Series ISO/IEC 14888		

Referencia: TSOHOU, A., KOKOLAKIS, S., LAMBRINOUDAKIS, C. y GRITZALIS, S. 2010. Information Systems Security Management: A Review and a Classification of the ISO Standards. *Next Generation Society. Technological and Legal Issues* [en línea], vol. 26, pp. 220-235. .



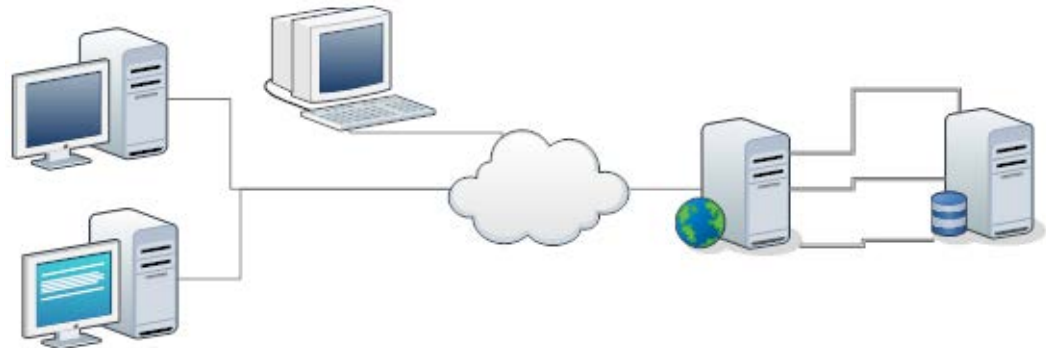
Estándares de seguridad

<i>Access Control</i>	ISO/IEC 15816:2002		
	Series ISO/IEC 9798		
<i>Information Systems Acquisition, Development and Maintenance</i>	Series ISO/IEC 11770		ISO/IEC CD 11770-1
	Series ISO/IEC 14888		
	Series ISO/IEC 9796		
	Series ISO/IEC 18033		
	Series ISO/IEC 9797		ISO/IEC FCD 9797-1 ISO/IEC FCD 9797-2
	Series ISO/IEC 10118		ISO/IEC CD 10118-2
	ISO/IEC 21827:2008		
	Series ISO/IEC 13888		
<i>Information Security Incident Management</i>	ISO/IEC TR 18044:2004		
<i>Business Continuity Management</i>	ISO/IEC 24762:2008		
	ISO/IEC 18045:2008		
	Series ISO/IEC 15408		ISO/IEC FCD 15408-1.3
	ISO/IEC TR 19791:2006		
	Series ISO/IEC TR 15443		ISO/IEC NP TR 15443-1 ISO/IEC NP TR 15443-2 ISO/IEC NP TR 15443-3
<i>Compliance</i>		ISO/IEC WD 27007	

Referencia: TSOHOU, A., KOKOLAKIS, S., LAMBRINOUDAKIS, C. y GRITZALIS, S. 2010. Information Systems Security Management: A Review and a Classification of the ISO Standards. *Next Generation Society. Technological and Legal Issues* [en línea], vol. 26, pp. 220-235. .

Pool de conexiones

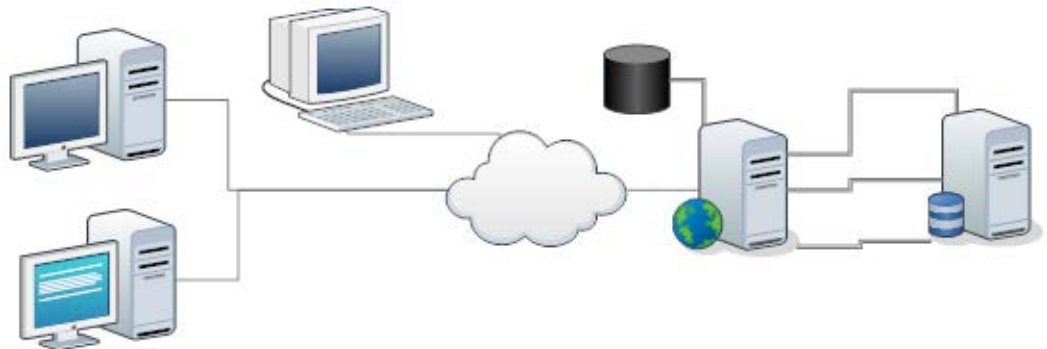
- Ventajas:
 1. No se almacena la contraseña con la aplicación
 2. Los clientes son transparentes ante el cambio de contraseñas
 3. La contraseña no viaja entre la aplicación y el servidor web
 4. Menor consumo de recursos, pues las conexiones se reciclan en lugar de abrirlas y cerrarlas.



Seguridad en la programación

- Persistencia y seguridad
 - Los pools de conexión no son la solución para todos los problemas de velocidad
 - La persistencia nos evita acceder continuamente a las tablas de las bases de datos

- Variables de sesión
- JPA
- Hibernate





Seguridad en la programación

1. POST

- Los valores del formulario viajan ocultos hacia el script de destino.

2. Token

- Permite reconocer que la información proviene del formulario para el cuál fue diseñado.

3. Validaciones Javascript

- Validar que los campos requeridos de un formulario se hayan completado y que tengan un formato y extensión apropiados mejora grandemente la experiencia del usuario y dificulta los posibles ataques.

4. Validaciones Internas

- Esto es importante ya que los ataques maliciosos podrían intentar enviar código por un formulario, que al ser leído o desplegado podría ejecutarse y causar serios problemas.



Seguridad en la programación

- Tokens al implementar formularios
- Evitar la inyección de código
 - PreparedStatement
 - Stored Procedures en el servidor de base de datos
- Encriptación de parámetros de hipervínculos y limitación de parámetros
- Uso de Frameworks de desarrollo

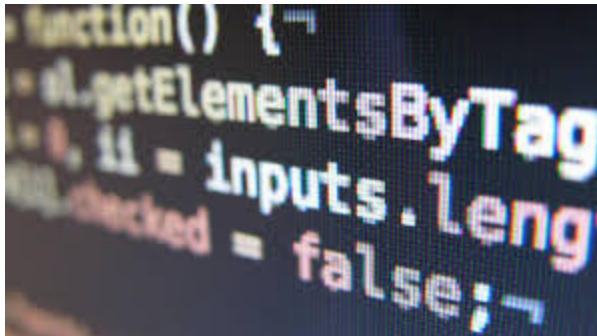
Struts



GRAILS

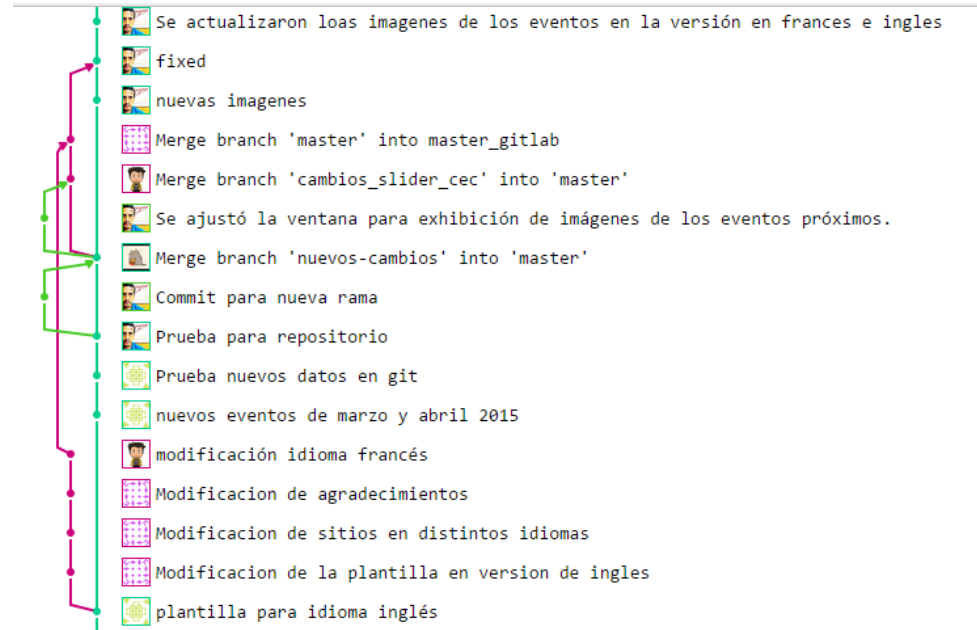
Seguridad en la programación

- Seguridad en los clientes



Planes de contingencia de recuperación del software

- RespalDOS de fuentes
- Documentación de implementaciones
- Control de versiones



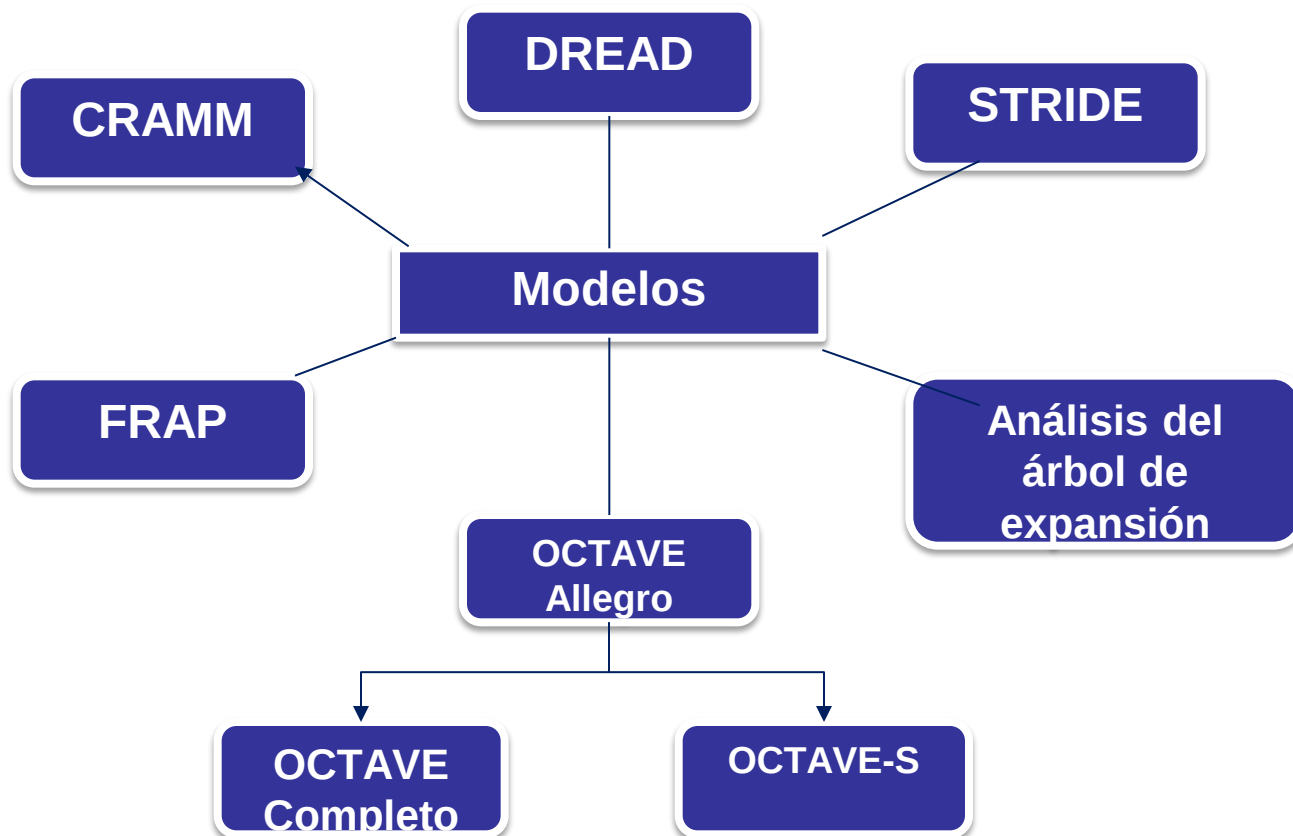


Mitigación de Riesgos (Contramedidas)

- Políticas de seguridad más ajustadas
- Implementación de reglas más estrechas (limitación a cierto equipo, hora, usuario, puertos, cambio de contraseñas)
- Ubicación del atacante
- Actualización /Cambio del software
- Revisión de accesos



Modelos de cálculos de Riesgos (seguridad de TI)

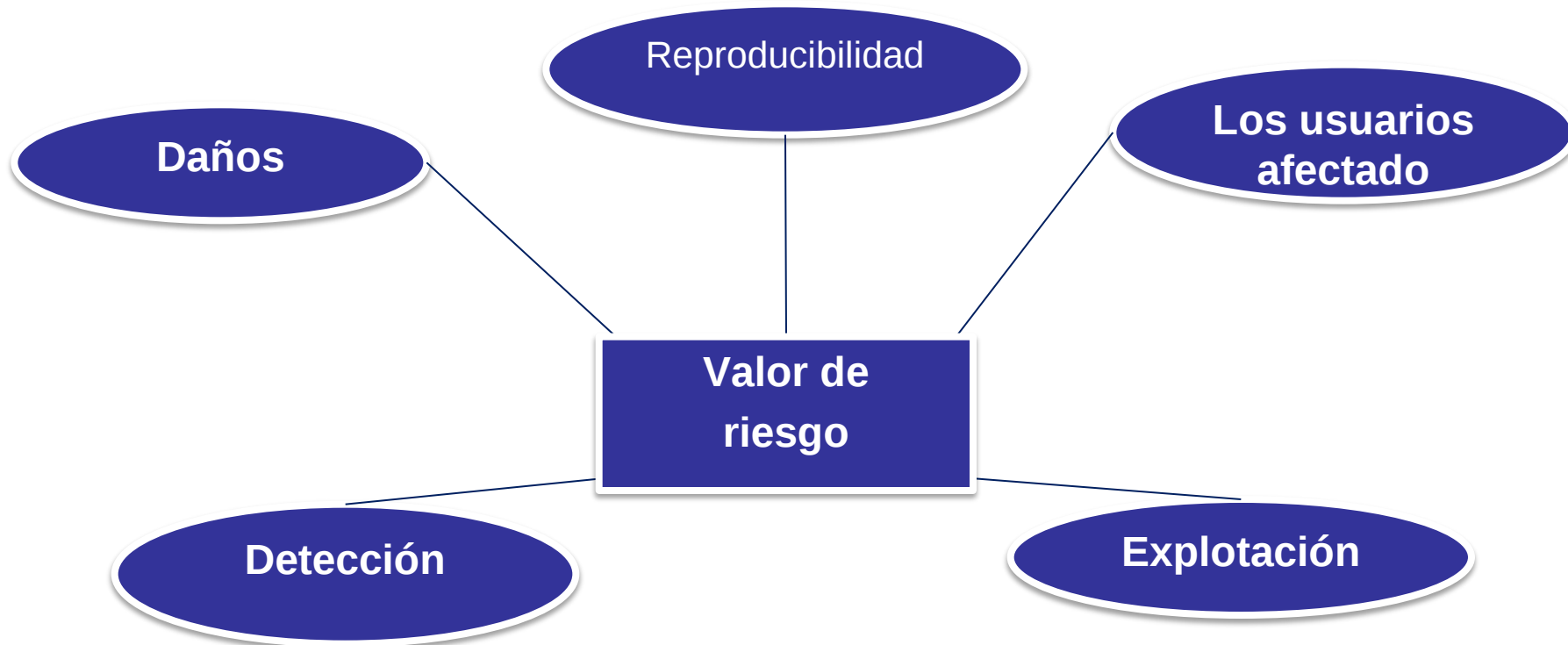


Referencia: Lepofsky, Ron (2014). The Manager's Guide to Web Application Security
A Concise Guide to the Weaker Side of the Web. Apress



DREAD

Esquema de clasificación para cuantificar, comparar y priorizar la cantidad de riesgo que presenta cada amenaza evaluada.





VALOR MEDIO

Escala

0: Nada.

5: Riesgo medio.

10 Riesgo alto.

EJEMPLO:

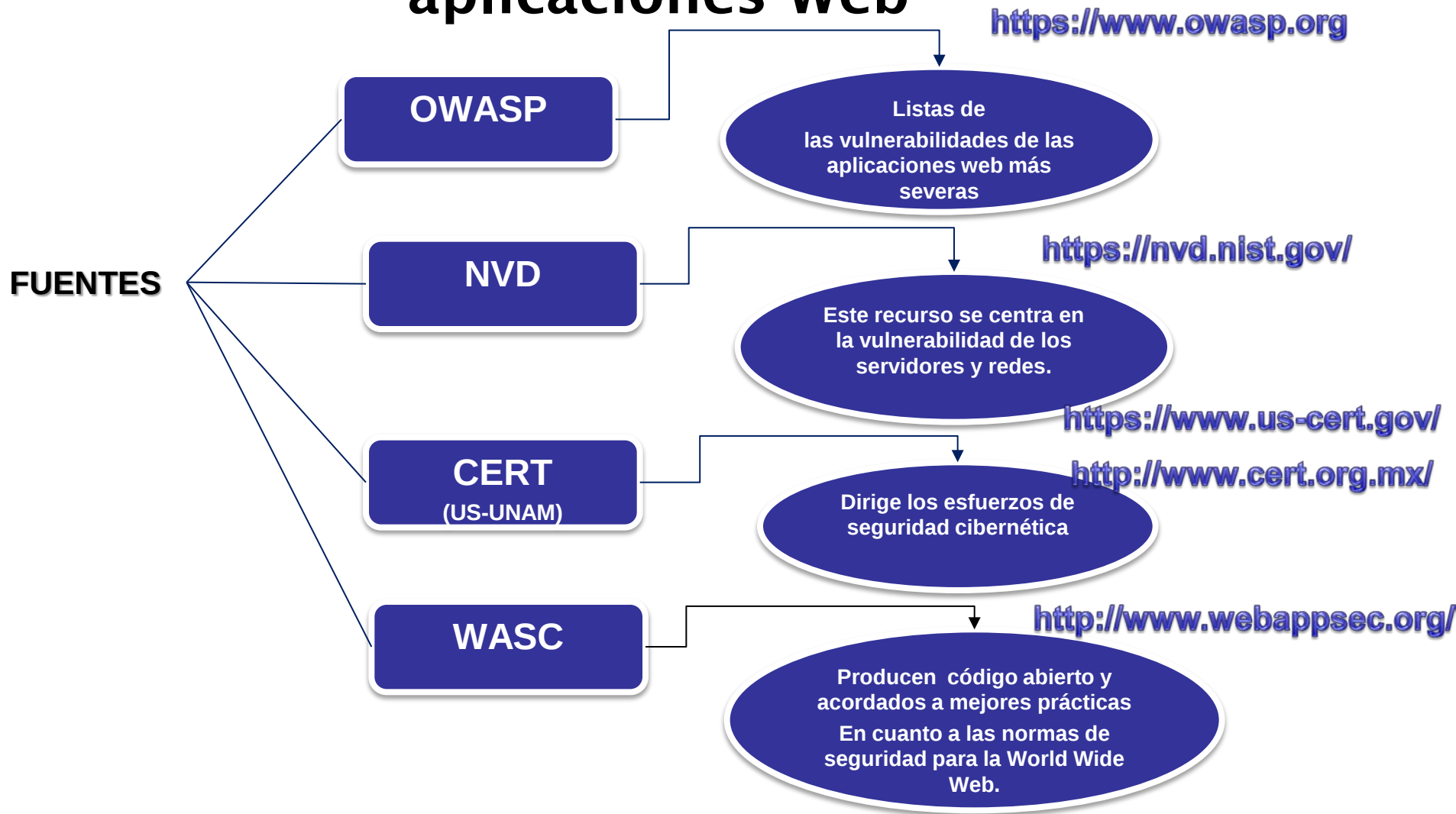
La vulnerabilidad de cross-site scripting, cuya puntuación DREAD pueden ser.

- Potencial de daños: 10
- Reproducibilidad: 5
- Explotación: 10
- Los usuarios afectados: 10
- Detección: 5

Puntuación total: 40



Fuentes de seguridad de las aplicaciones Web





OWASP Top 10 - web application security

A1-Inyección

A2-Pérdida de autenticación y gestión de sesiones

A3-Secuencia de comandos en sitios cruzados XSS

A4-Referencia directa insegura a objetos

A5-Configuración de seguridad incorrecta

A6-Exposición de datos sensibles

A7-Ausencia de control de acceso a las funciones

A8-Falsificación de peticiones en sitios cruzados

A9-Uso de componentes con vulnerabilidades conocidas

A10-Redirecciones y reenvíos no validados



National Vulnerability Database - Vulnerabilidades MV Java

3:33:14 PM
CVE Publication rate: 11.97

Email List

NVD provides four mailing lists to the public. For information and subscription instructions please visit [NVD Mailing Lists](#)

Workload Index

Vulnerability [Workload Index](#): 5.47

About Us

NVD is a product of the NIST [Computer Security Division](#) and is sponsored by the Department of Homeland Security's [National Cyber Security Division](#). It supports the U.S. government multi-agency ([OSD](#), [DHS](#), [NSA](#), [DISA](#), and [NIST](#)) Information Security Automation Program. It is the U.S. government content repository for the Security Content Automation Protocol (SCAP).

REPORT A VULNERABILITY

REPORT AN INCIDENT

consumption) via the ContentManager.
Published: 4/24/2015 10:59:06 AM

CVSS Severity: 9.0 HIGH

CVE-2015-0492

Summary: Unspecified vulnerability in Oracle Java SE 7u76 and 8u40, and JavaFX 2.2.76, allows remote attackers to affect confidentiality, integrity, and availability via unknown vectors, a different vulnerability than CVE-2015-0484.
Published: 4/16/2015 12:59:43 PM

CVSS Severity: 9.3 HIGH

CVE-2015-0491

Summary: Unspecified vulnerability in Oracle Java SE 5.0u81, 6u91, 7u76, and 8u40, and Java FX 2.2.76, allows remote attackers to affect confidentiality, integrity, and availability via unknown vectors related to 2D, a different vulnerability than CVE-2015-0459.
Published: 4/16/2015 12:59:42 PM

CVSS Severity: 10.0 HIGH

CVE-2015-0488

Summary: Unspecified vulnerability in Oracle Java SE 5.0u81, 6u91, 7u76, and 8u40, and JRockit R28.3.5, allows remote attackers to affect availability via vectors related to JSSE.
Published: 4/16/2015 12:59:39 PM

CVSS Severity: 5.0 MEDIUM

CVE-2015-0486

Summary: Unspecified vulnerability in Oracle Java SE 8u40 allows remote attackers to affect confidentiality via unknown vectors related to Deployment.
Published: 4/16/2015 12:59:37 PM

CVSS Severity: 5.0 MEDIUM

CVE-2015-0484

Summary: Unspecified vulnerability in Oracle Java SE 7u76 and 8u40, and Java FX 2.2.76, allows remote attackers to affect confidentiality, integrity, and availability via unknown vectors, a different vulnerability than CVE-2015-0492.
Published: 4/16/2015 12:59:36 PM



Complementos de seguridad para aplicaciones Web

Herramientas en línea para testeo de aplicaciones
(**Web Application Security Scanner**)

- **Comerciales**
- **Software-as-a-Service Providers**
- **Free / Open Source Tools**
 - ✓ Cross-Site Scripting
 - ✓ SQL Injection
 - ✓ File Inclusion
 - ✓ Backup files check
 - ✓ Simple AJAX check (parse every JavaScript and get the URL and try to get the parameters)
 - ✓ JavaScript source code analyzer: Evaluation of the quality/correctness

<http://projects.webappsec.org/w/page/13246988/Web%20Application%20Security%20Scanner%20List>



Complementos de seguridad para aplicaciones Web

Web Application Firewall

Capacity			
Backend Servers	5 - 10	10 - 25	25 - 150
Throughput	50 Mbps	200 Mbps	1 Gbps
HTTP Transactions/Sec.	15,000	30,000	90,000
SSL Transactions/Sec.	4,000	12,000	30,000
Hardware			



Resumen de buenas prácticas

A. Identificar los activos de software en funcionamiento

1. Activos
2. Políticas claras y del conocimiento del equipo de desarrollo
3. Planes de contingencia y recuperación
4. Uso de procedimientos de seguridad (normas ISO)
5. Servidores (aplicaciones, base de datos, autenticación, etc.) actualizado y con los últimos parches
6. Puertos de administración sin acceso externo y solo desde direcciones autorizadas
7. Resguardo del software de desarrollo vía control de versiones
8. Planes de Mitigación



Resumen de buenas prácticas

B. Crear una vista del funcionamiento del software

1. Autenticar y autorizar
2. Uso de certificados SSL
3. Usar seguridad por capas (Framework - modelo MVC)
 - a) Validación de todos los datos en cliente (vista), servidor (modelo/controlador), base de datos
 - b) uso de criptografía correctamente y donde sea pertinente
4. Actualización periódica de Componentes internos (jar)
5. Implementación de pruebas de seguridad frecuentes



Resumen de buenas prácticas

C. Documentación y difusión

1. Documentación de cambios al software
2. Documentar las soluciones a amenazas, mitigaciones.
3. Análisis de riesgos periódicos
4. Vigilancia tecnológica constante (sitios seguridad – cert – OWASP, NVD)
5. Difundir las buenas prácticas de seguridad



Gracias por su atención



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO