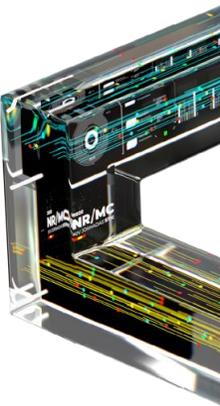
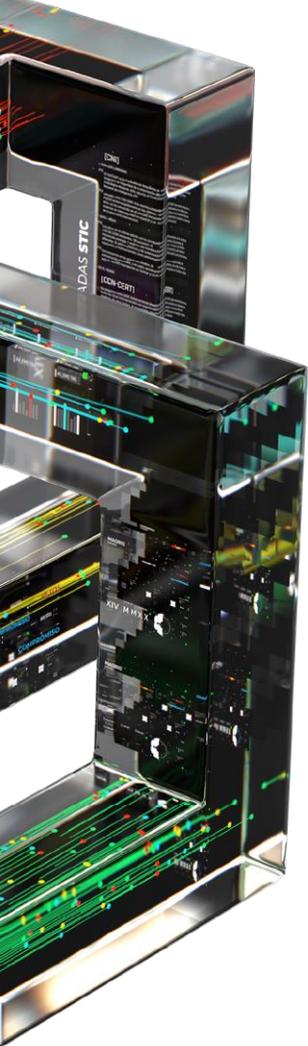


#XIVJORNADASCCNCERT

Los criminales no esperan a las normas, por eso, ¡debemos actuar ya!

Desarrollo Seguro y ciclo de vida





Alberto Fuentes Rodríguez

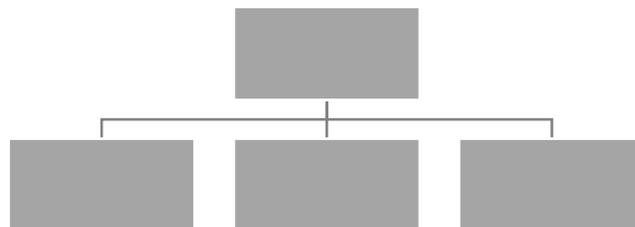
SGS Cybersecurity Services

alberto.fuentesrodriguez@sgs.com

¿Por qué necesitamos el desarrollo seguro?

Ofrecer un mejor producto

- Pensar en seguridad desde la captura de requisitos.
- Diseño e implementación orientados a la robustez.



¿Por qué necesitamos el desarrollo seguro?

Minimizar los costes

- Si tenemos que cumplir con un determinado estándar.
- Generar los **entregables** a posteriori es un mayor esfuerzo.
- **Reputación** de la compañía / indemnizaciones



¿Por qué necesitamos el desarrollo seguro?

Certificaciones

- Las certificaciones son para productos muy concretos.
 - Entornos militares.
 - Dispositivos criptográficos (DNI, pasaporte...).
 - Dispositivos seguros orientados a la administración.
 - Grandes empresas.

- **Desarrollos muy específicos.**



¿Por qué necesitamos el desarrollo seguro?

Certificaciones

- Las certificaciones se aplican a productos muy variados:
 - Entornos militares.
 - Dispositivos criptográficos (llaves, pasaporte...).
 - Dispositivos seguros orientados a la administración.
 - Grandes empresas.
- Desarrollos más específicos.



¿Por qué necesitamos el desarrollo seguro?

Certificaciones (Ejemplo IOT)

US centric:

Regulation:

- Federal Trade Commission Act (FTC Act)
- **CCPA**: The California Consumer Privacy Act (2018)
- **California Bills**: SB 327 & AB 1906: reasonable security features for connected products
- Children's Online Privacy Protection Act (COPPA)
- Internet of Things (IoT) Cybersecurity Improvement Act

Standards / Specifications:

- UL 2900-1: Standard for Software Cybersecurity for Network-Connectable Products, Part 1: General Requirements

Best Practices:

- NIST Cybersecurity Framework
- NIST Considerations for Managing IoT Cybersecurity and Privacy Risks
- NIST Device Cybersecurity Capability Core Baseline **NISTIR 8259A**

Certification:

- **CTIA**: IoT Cybersecurity Certification Program



EU centric:

Regulation:

- CE Marking
- **GDPR regulation** (effective since May 2018)
- **EU Cybersecurity Act** defining an EU-wide cybersecurity certification framework (effective since June 2019)

Standards/Specifications:

- ETSI:
 - TS 103645 & **EN 303 645** (in dev.) (Securing Consumer IoT)
 - TS 103 701 (in dev.) (Cybersecurity assessment for IoT products)
 - TS 103 485 (in dev.) (Privacy Assurance and verification)

■ DIN SPEC 27072

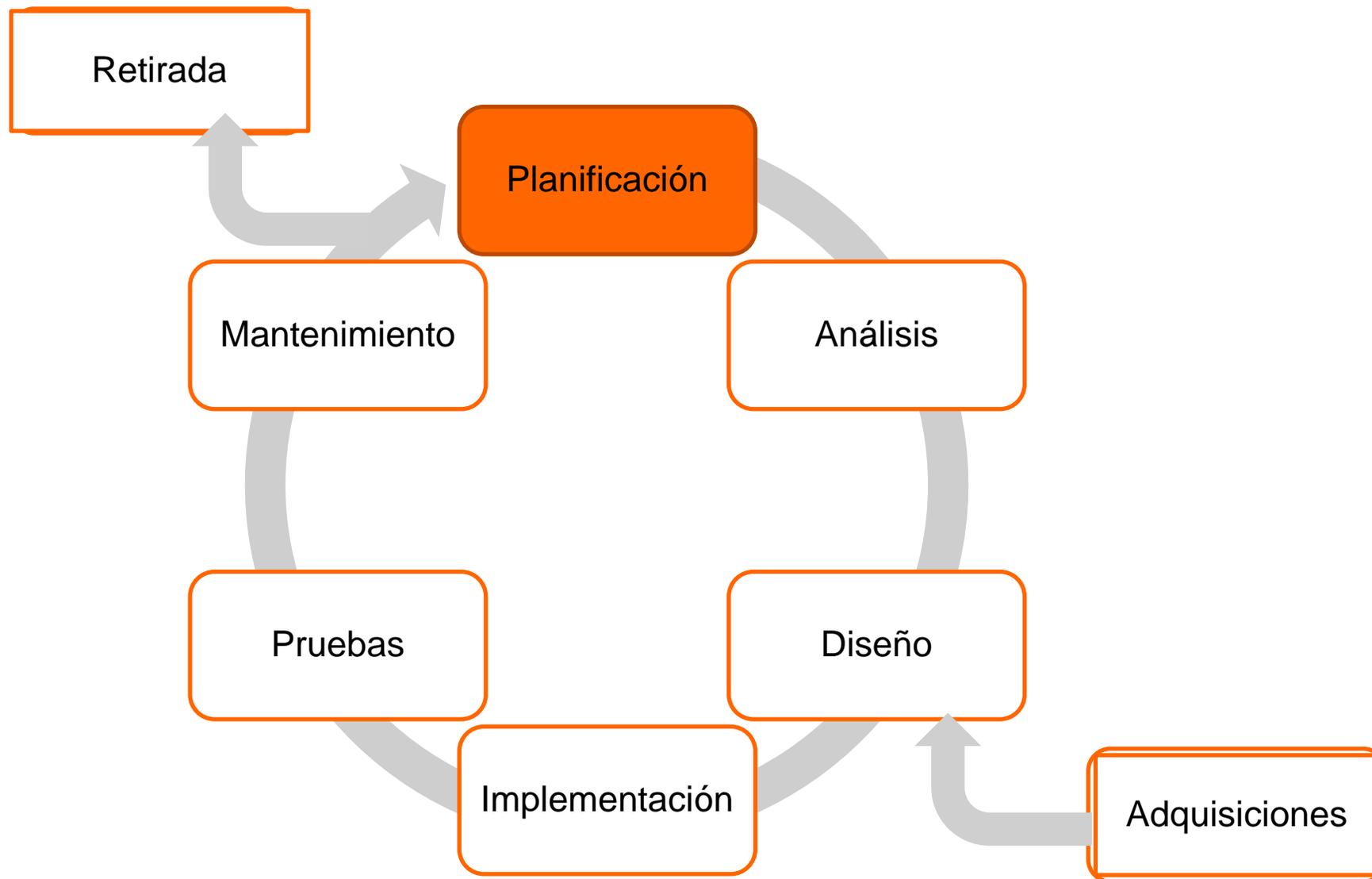
Best Practices / Guidance Documents:

- IT-Grundschutz, SYS4.4. IoT Devices (BSI)
- ENISA
 - Good Practices for Security of IoT in the context of Smart Manufacturing (11/2018)
 - Good practices for security of IoT - Secure Software Development Lifecycle (11/2019)
 - Towards secure convergence of Cloud and IoT (09/2018)

Widely accepted standards, specifications & guidances:

- GSMA: IoT Security Guidelines
- IoT Security Foundation: IoT Security Compliance Framework & Secure Design Best Practice Guides and more

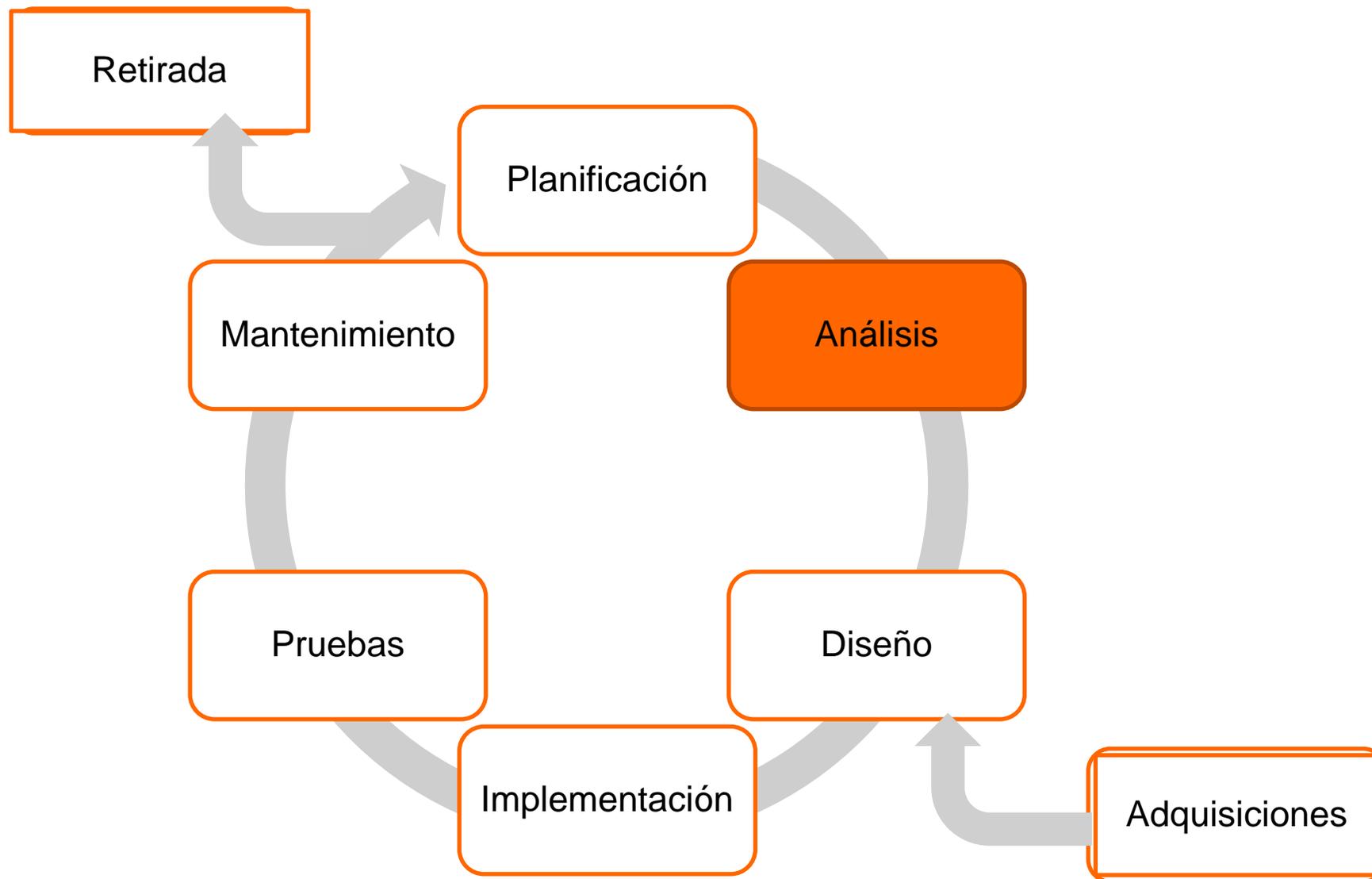
Desarrollo seguro (SDLC)



Planificación

- Se debe establecer un plan de desarrollo teniendo en cuenta:
 - **Profundidad** -> Entregables (Documentación, producto final).
- Asignar el rol de **Security architect** para el desarrollo.

Desarrollo seguro (SDLC)



Análisis

- Incluir requisitos regulatorios
 - Ej. GDPR, HIPAA
- Incluir los requisitos de las **certificaciones**

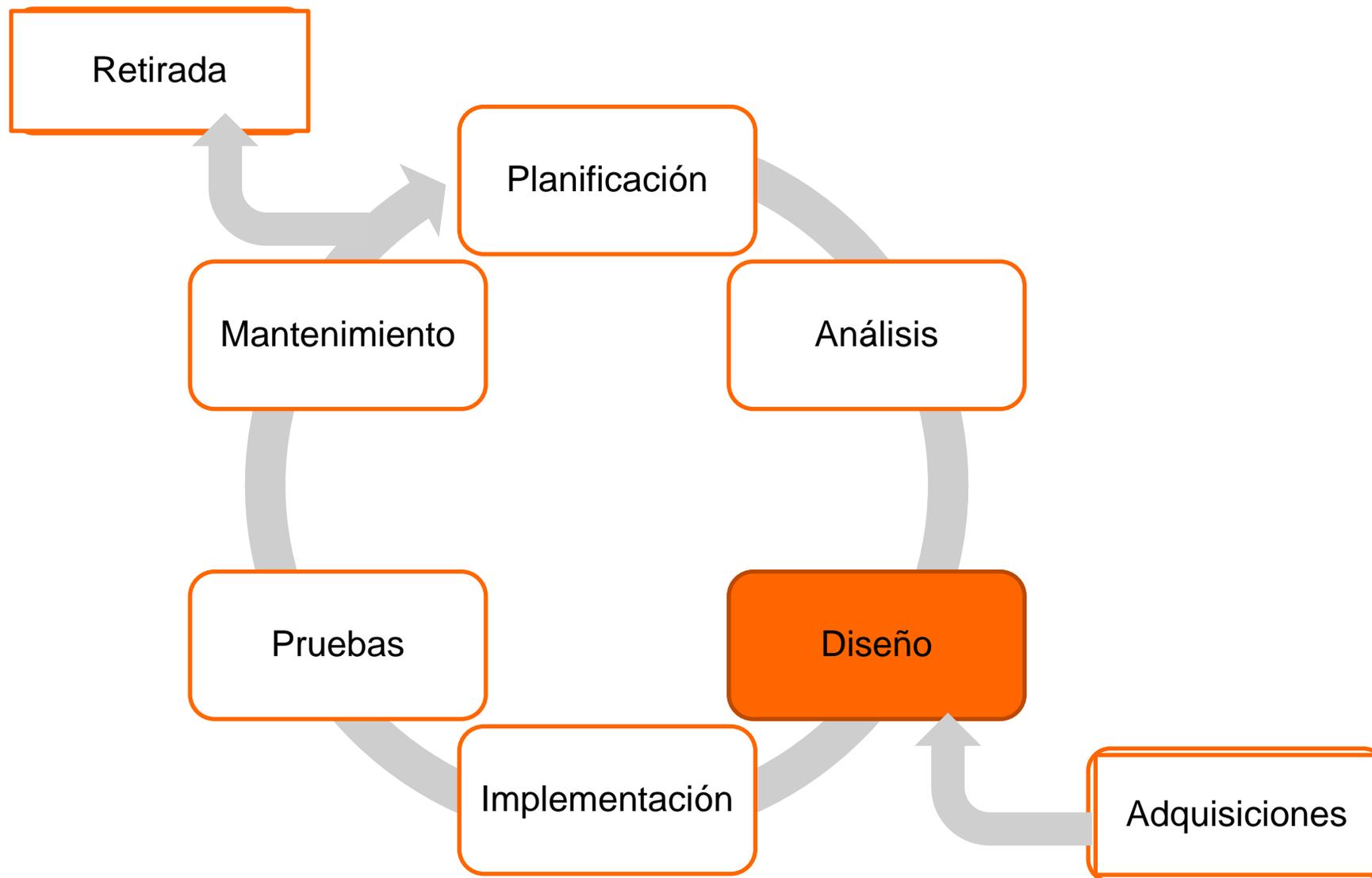


CC & LINCE



Industrial

Desarrollo seguro (SDLC)



Diseño

Interfaces

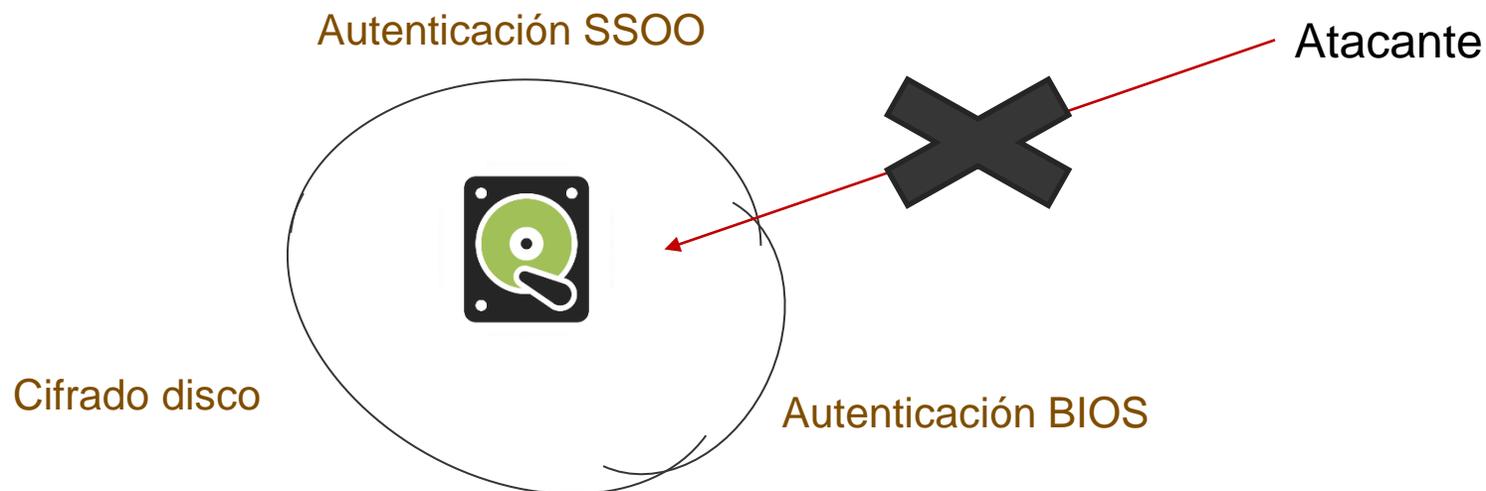
- Definir los **interfaces** de un producto en detalle para permitir identificar las amenazas y poner contramedidas.
- Analizar los **interfaces**
 - Categorizar los interfaces según su responsabilidad en la seguridad.
 - Validar las entradas.
 - Comprobar los errores.



Diseño

Arquitectura de seguridad

- La **arquitectura de seguridad** debe minimizar las amenazas tanto como sea posible → reducir superficie de ataque
- **Arquitectura** de seguridad
 - Especificar como protege el producto su propia funcionalidad de seguridad (tamper).
 - Comprobar que no quedan “huecos” sin cubrir (bypass).
 - Arranque seguro.
 - Separación de dominios



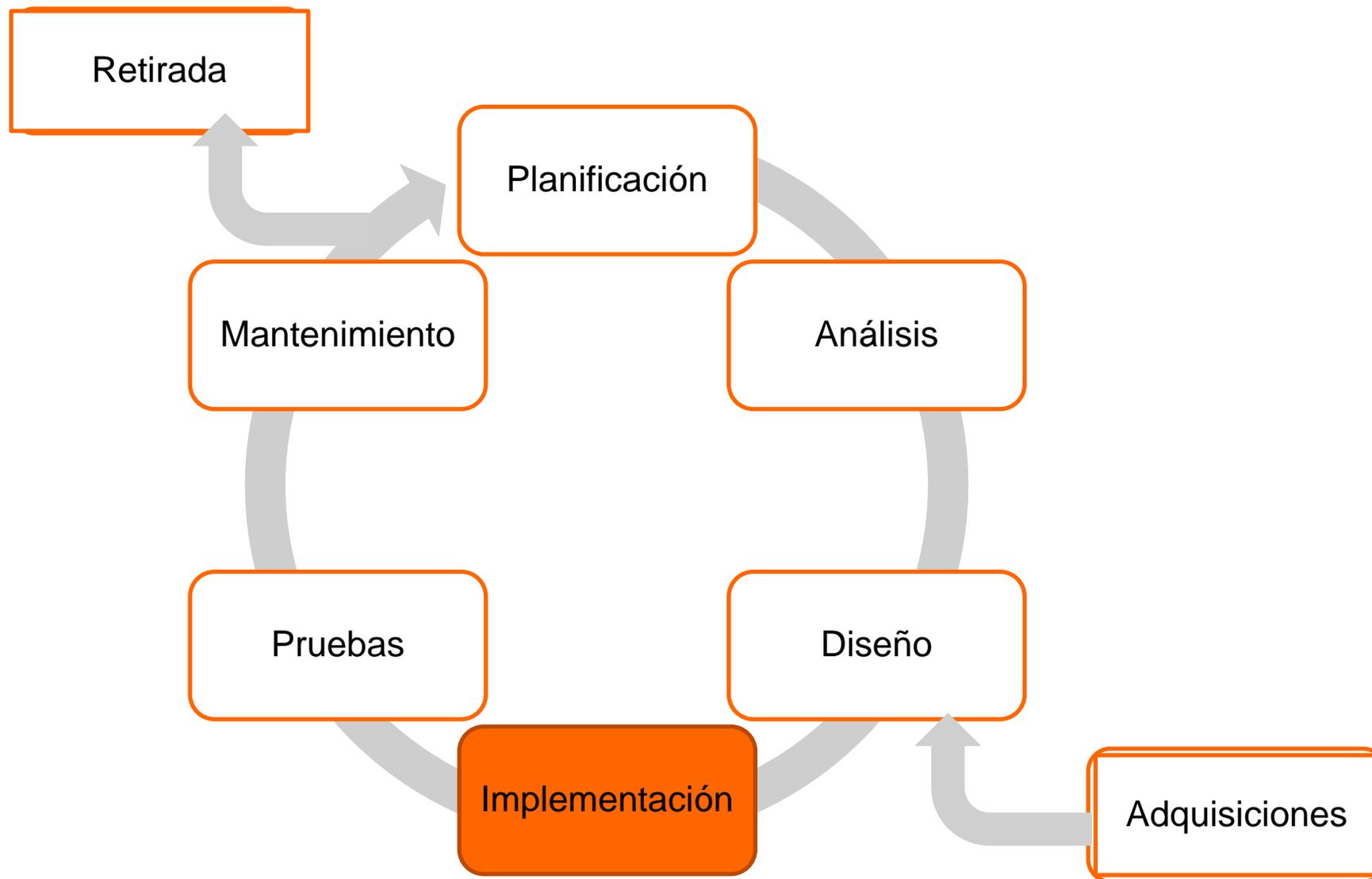
Diseño

Principios de seguridad

- Incluir **Técnicas de diseño** que fuercen a los desarrolladores a tener en cuenta la seguridad en cada línea de código (Security design principles)

- Least Privilege.
- Fail-Safe Defaults.
- Economy of Mechanism
- Complete Mediation
- Open Design
- Separation Privilege.
- Least Common Mechanism
- Psychological Acceptability
- Defense in depth

Desarrollo seguro (SDLC)

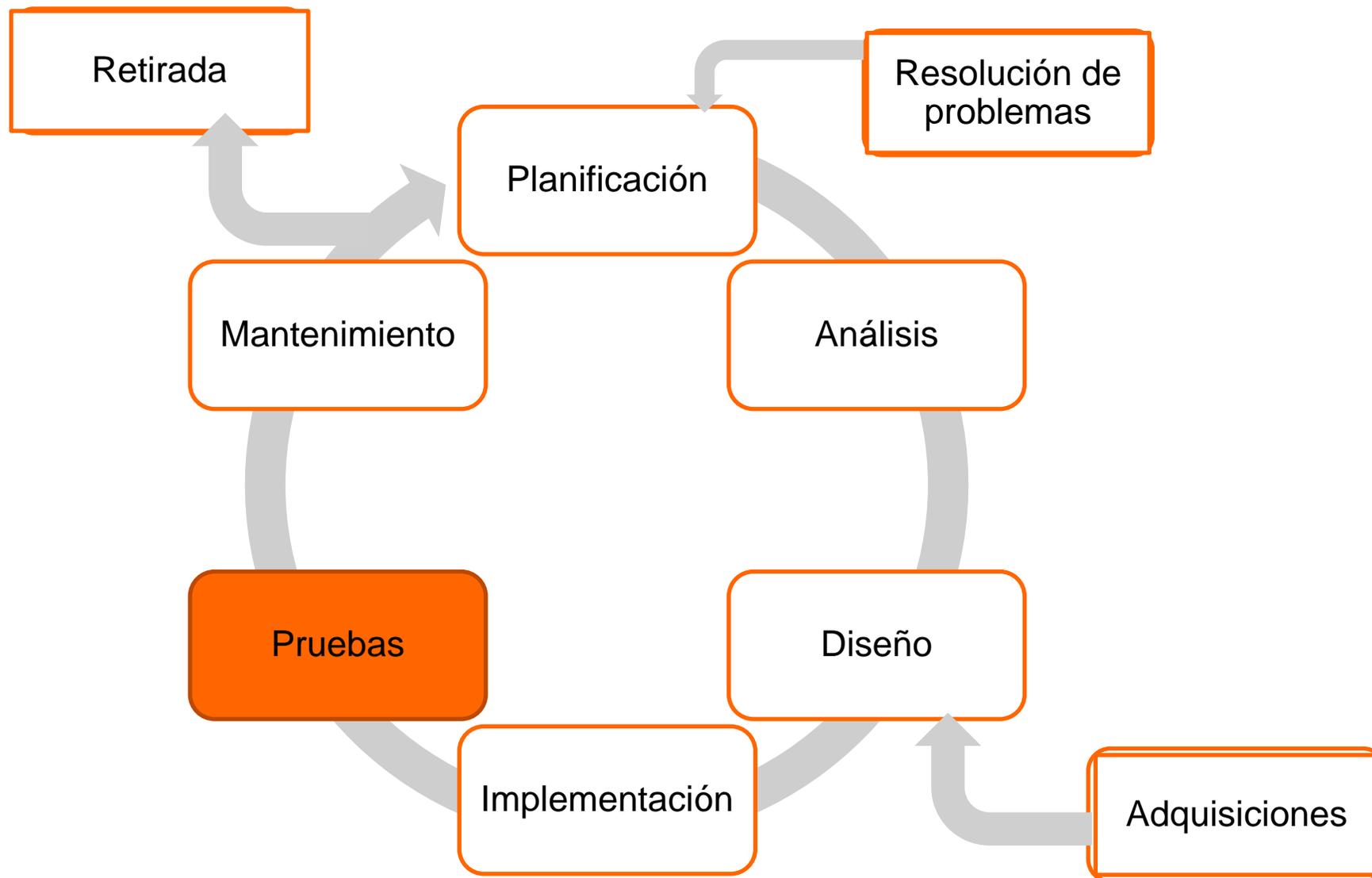


Desarrollo seguro (SDLC)

Implementación

- **Los errores de codificación ocurren** – el código es desarrollado por personas
 - Manejo de errores.
 - Evitar construcciones de código peligrosas.
 - Implementar validación de entradas y cifrado.
 - ...
- Seguir las “mejores prácticas” de implementación y tenerlas actualizadas.

Desarrollo seguro (SDLC)



Pruebas

- Incluir las pruebas de seguridad en el plan de pruebas
- Llevar a cabo **pruebas de penetración**



Pruebas

- Ejecutar pruebas automatizadas de seguridad
 - Análisis estático de código.
 - Análisis estático de binarios.
 - Análisis dinámico / Fuzzing (robustez).

```
40
41 $(function){cards();};
42 $(window).on('resize', function(){cards();});
43 function cards(){
44   var width = $(window).width();
45   if(width < 750){
46     cards_smallscreen();
47   }else{
48     cards_bigscreen();
49   }
50 }
51 function cards_smallscreen(){
52   var cards = $('#card').length;
```



Pruebas

Revisión de código

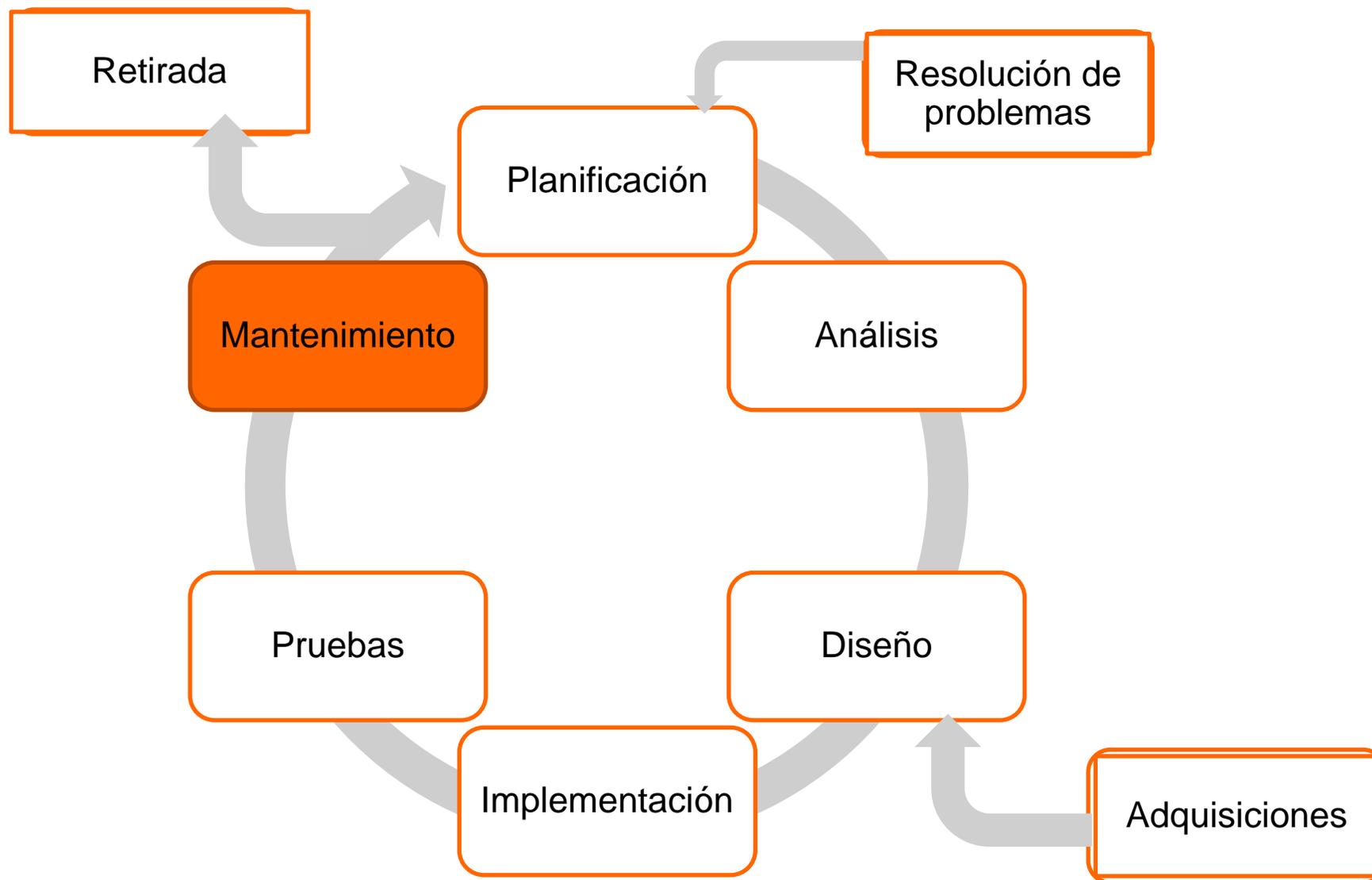
Revisión de código

- Llevar a cabo revisiones de código regulares
- Herramientas automatizadas de análisis de código



No tiene en cuenta la lógica del producto

Desarrollo seguro (SDLC)



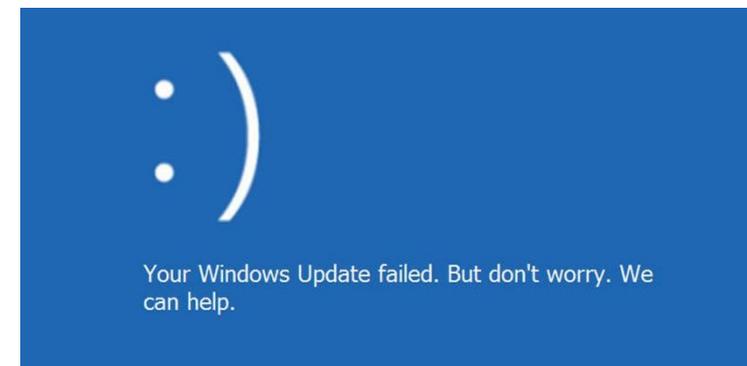
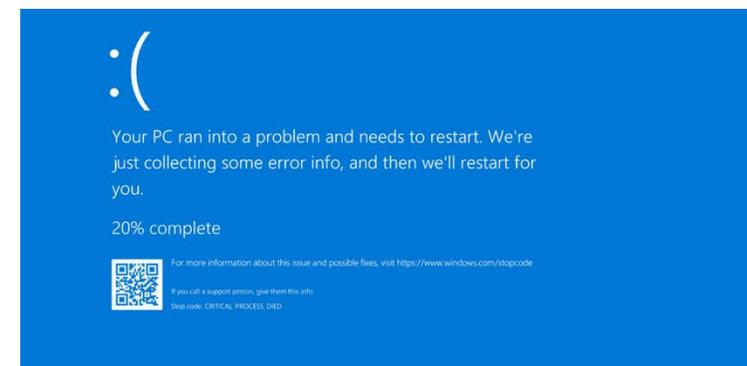
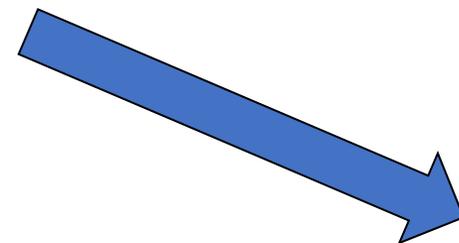
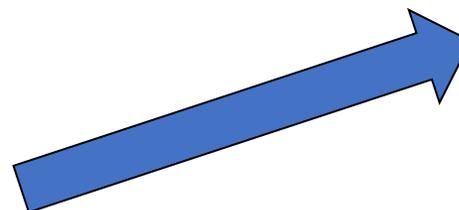
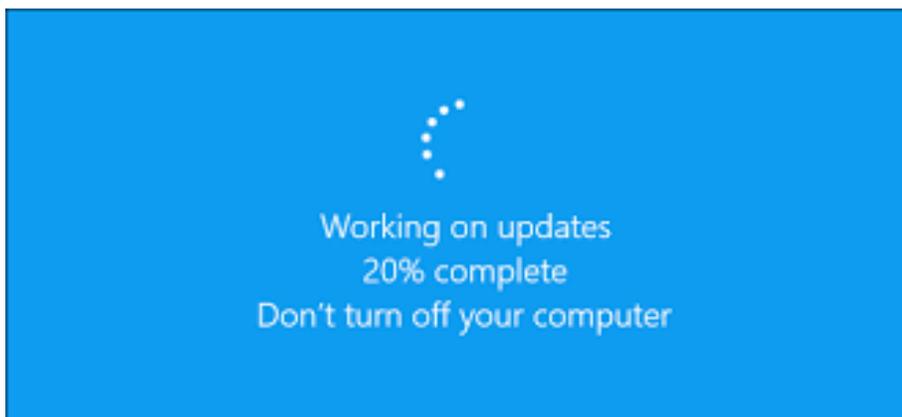
Mantenimiento

- **Gestión de vulnerabilidades**
 - Para tus propios productos.
 - Para productos de terceros utilizados por la solución.
- Vía de **comunicación de vulnerabilidades** encontradas por terceros.
- **Publicaciones** periódicas o especiales para eventos importantes relacionados con la seguridad.

Mantenimiento

Actualizaciones y parches

- El ciclo de vida también se aplica a parches.
- Comprobar **tiempo de respuesta** predeterminado.
- Los parches deberán estar firmados digitalmente.



Desarrollo seguro (SDLC)

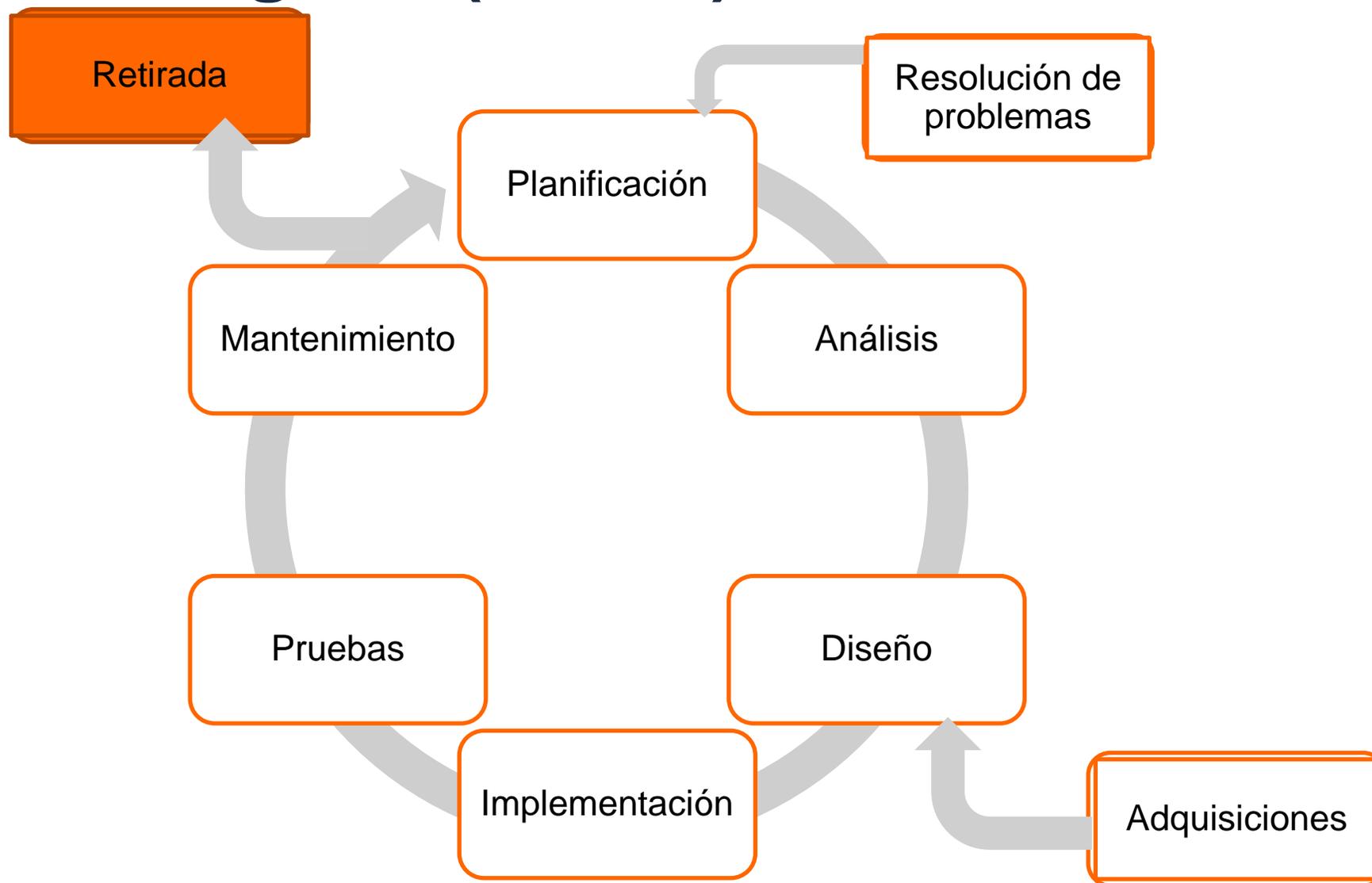


Adquisiciones

- Con el termino **adquisiciones**, en este campo nos referimos al uso de productos de terceros.
- Los procedimientos de adquisiciones tendrán que tener en cuenta:
 - La revisión de las especificaciones de seguridad
- Tener en cuenta el proceso de retirada de las adquisiciones (fecha finalización de mantenimiento)

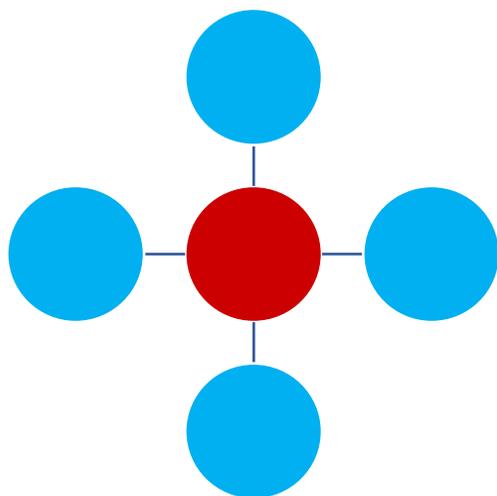


Desarrollo seguro (SDLC)



Retirada

- Definir procedimiento de retirada **sin comprometer la seguridad** del sistema.
- **Eliminar** información sensible del cliente.





MUCHAS GRACIAS

#XIVJORNADASCCNCERT