jtsec

BEYOND IT SECURITY

Common Criteria: una herramienta para el desarrollo seguro

12/05/2018

Índice

- Introducción a Common Criteria
- Common Criteria y Desarrollo Seguro
- Seguridad más allá del desarrollo
- Perfiles de protección
- Catálogo de productos STIC (CPSTIC)

Introducción a Common Criteria

Common Criteria: certificación de seguridad

- ¿Es posible demostrar la seguridad de un producto?
- Demostrar que un producto es seguro (libre de vulnerabilidades) es como demostrar que una ciudad construida junto a un río nunca se va a inundar.
- Sin embargo, sí que es posible asegurar con un **determinado nivel de garantí**a que un producto **satisface unos requisitos** funcionales de seguridad.

Common Criteria: certificación de seguridad

- Common Criteria es un estándar reconocido internacionalmente para **evaluar las funcionalidades y garantías de seguridad de los productos de IT** (ISO 15408)
- Los certificados de Common Criteria cuentan con un **reconocimiento mundial e interprofesional**
- En España el organismo de certificación es el Centro Criptológico Nacional (CCN) dependiente del CNI.
- Un producto certificado Common Criteria tiene un **elemento diferenciador en términos de seguridad**, ya que ha sido evaluado por un tercero bajo una metodología de evaluación sólida y bien definida.

COMMON CRITERIA

Requisitos Functionales de Seguridad (SFRs)

- FAU. Security audit.
- FCO. Communication.
- FCS. Cryptographic support
- FDP. User data protection
- FIA. Identification and authentication
- FMT. Security management
- FPR. Privacy
- FPT. Protection of the TSF
- FRU. Resource utilisation
- FTA. TOE access
- FTP. Trusted Path/Channels

Clases de Garantía (SARs)

- **Documentación**:
  - **ASE**: evaluación de la Declaración de Seguridad
  - **ADV**: evaluación de la documentación de diseño

- **Testing**:
  - **AVA**: Análisis de vulnerabilidades

- **Documentación + Testing:**
  - **AGD:** guías operacionales y de instalación
  - **ALC**: evaluación de la documentación de ALC (Ciclo de vida, gestión de la configuración, herramientas y técnicas, entrega, resolución de defectos, y medidas de seguridad del sitio) y auditoría del sitio de desarrollo.
  - **ATE**: evaluación de la documentación de pruebas y testing independiente

Declaración de Seguridad

- El documento clave en una evaluación Common Criteria es la Declaración de Seguridad.

- En la Declaración de Seguridad se define:
  - El **TOE** (límites)
  - Las propiedades de seguridad -> **SFRs**
  - El nivel de garantía (EAL): **SARs**

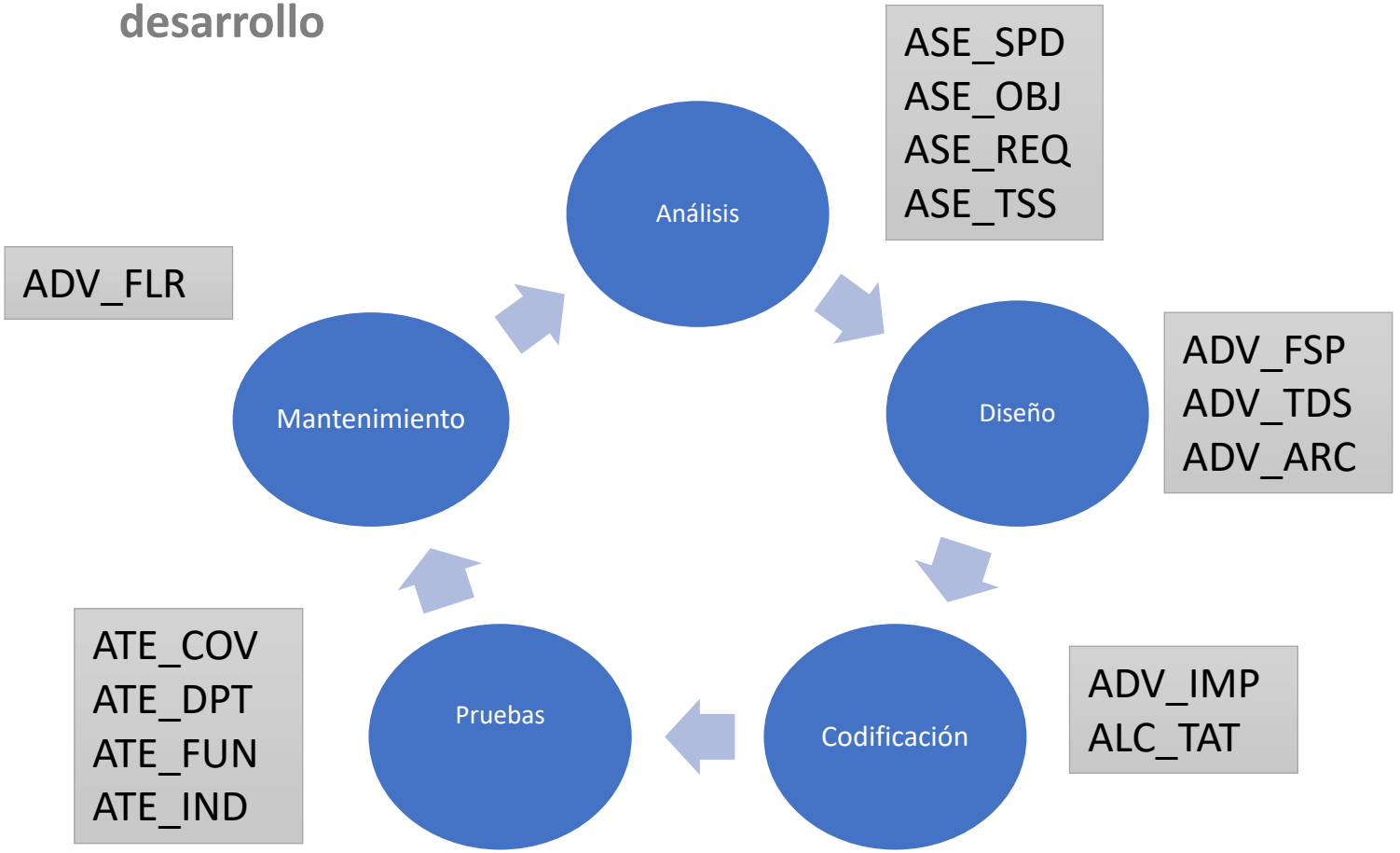
- La Declaración de Seguridad es un documento **público**.
  - https://oc.ccn.cni.es/index.php/es/productos-certificados/productos-certificados
  - https://www.commoncriteriaportal.org/products/


#DóndeEstabas8
¡HABLEMOS DEL DESARROLLO!
imgflip.com

Common Criteria y Desarrollo Seguro

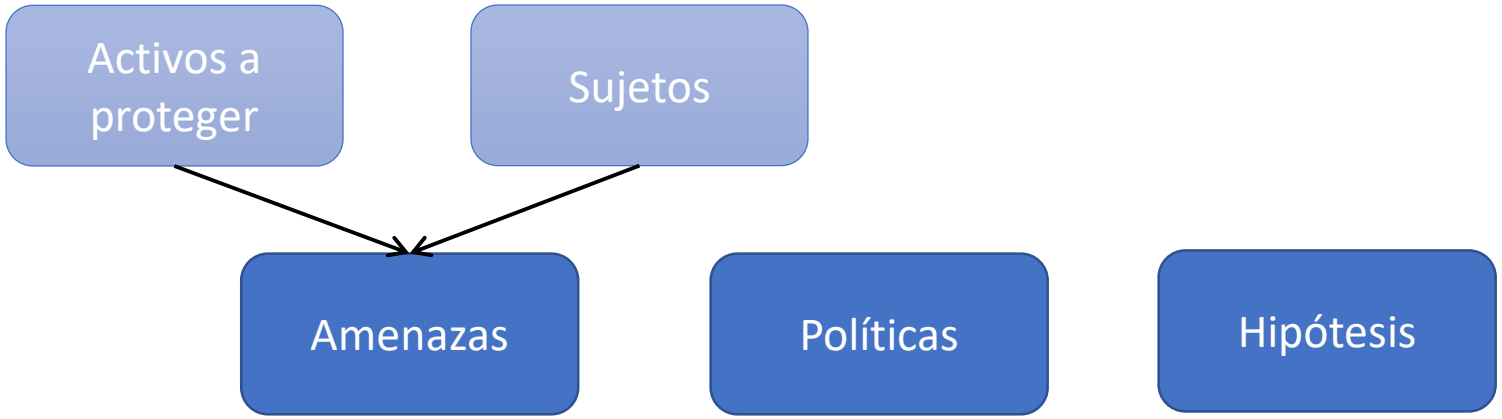
Common Criteria y Desarrollo Seguro

- Desarrollar un producto con siguiendo los requisitos de garantía de Common Criteria **contempla la seguridad en todas las fases del desarrollo**



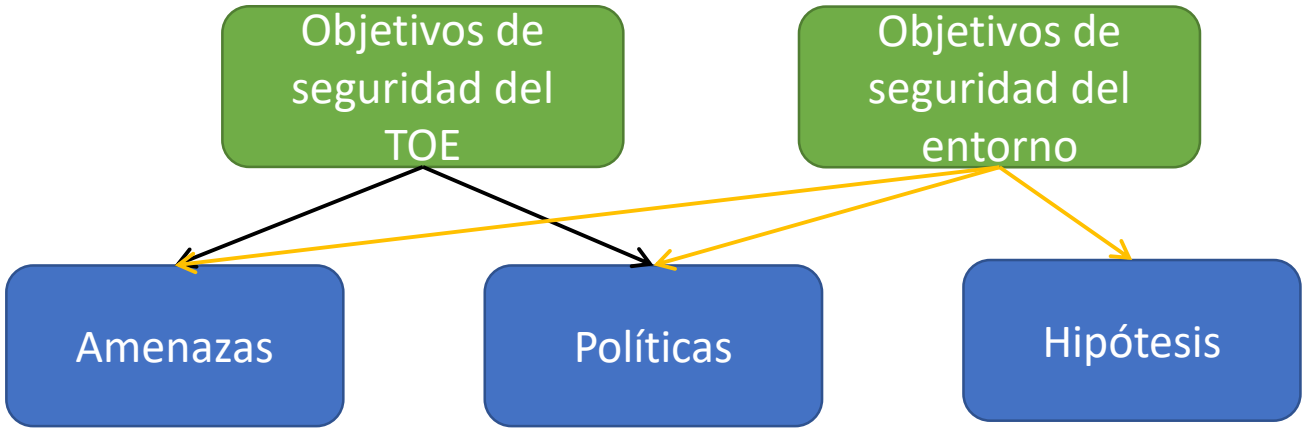
Fase de Análisis – Problema de seguridad

- ASE_SPD: **definición del problema de seguridad.**
- CC requiere plantear el análisis como la resolución de un **problema desde el punto de vista de la seguridad**.
- Requisitos de la clase ASE -> Análisis plasmado en la **Declaración de Seguridad**



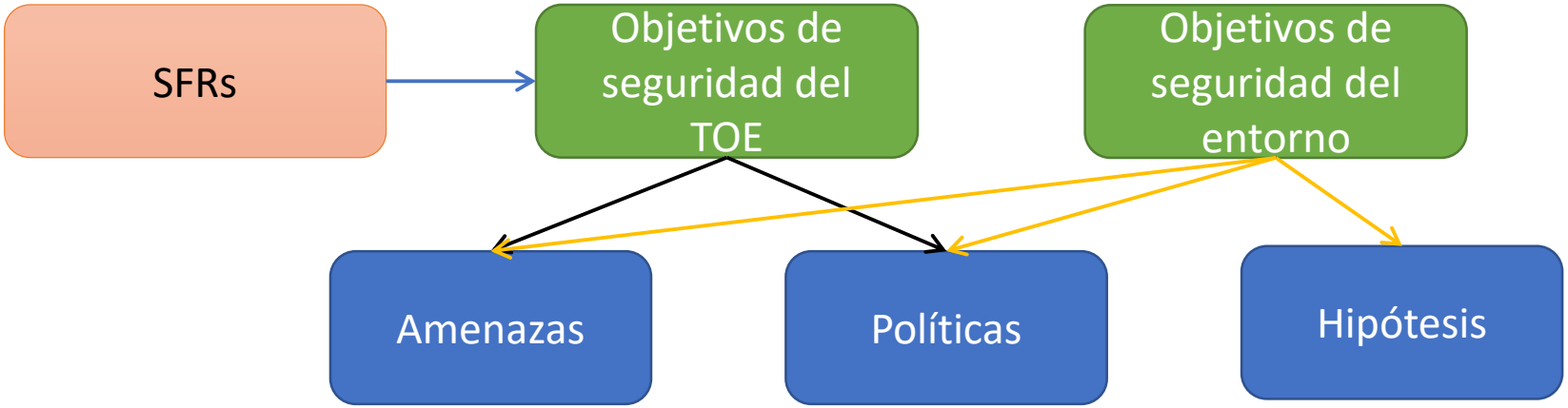
Fase de Análisis – Objetivos de seguridad

- ¿Cuáles son los **objetivos de seguridad** que debe cumplir el producto para **resolver el problema de seguridad planteado**?
  - **Objetivos de seguridad del TOE**: qué parte del problema la resuelve el TOE mediante su funcionalidad de seguridad
  - **Objetivos de seguridad del entorno** operacional: qué parte del problema resuelve el entorno.



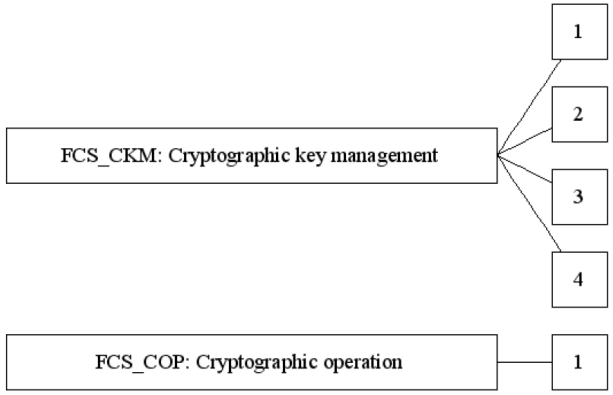
Fase de Análisis - SFRs

- ¿Cómo cumplirá el producto los objetivos de seguridad?
  - ASE_REQ: definir **requisitos funcionales de seguridad (SFRs)** que permiten al producto cumplir los objetivos de seguridad.
  - Razonamiento: Problema de seguridad -> Objetivos de seguridad -> SFRS



Fase de Análisis - SFRs

- Los requisitos funcionales de seguridad **SFRs** se escogen de entre un catálogo incluido en la Parte 2 de Common Criteria
- Están organizados en **clases, familias y componentes**.
- Se contempla añadir componentes extendidos (fuera del catálogo)



Clase FCS: soporte criptográfico
- Familia FCS_CKM: Manejo de claves criptográficas
- Familia FCS_COP: Operaciones criptográficas
  - Componente FCS_COP.1

FCS_COP.1.1 **The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].**

Fase de Análisis

re-cap:

Cumplir los requisitos de Common Criteria implica hacer un análisis un razonamiento basado en una solución demostrable a un problema de seguridad

Fase de Diseño

- Seguridad en el diseño
  - Especificación Funcional (ADV_FSP)
  - Diseño del TOE (ADV_TDS)
  - Arquitectura de Seguridad (ADV_ARC

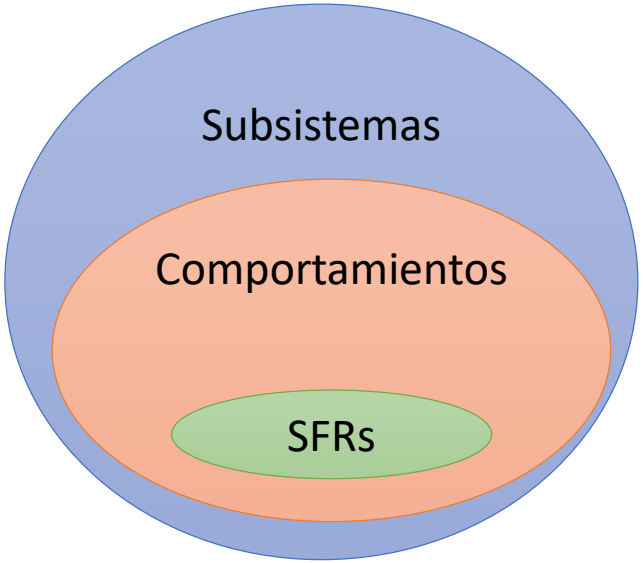
Fase de Diseño – Especificación Funcional

- Definir y describir **interfaces** del TOE con el exterior.
- **TSFI** (TOE Security Functional Interfaces). Ejemplos: HTTPS para una aplicación web, Interfaz eléctrónica del chip o interfaz contactless para una tarjeta inteligente.
- ¿Qué funciones de seguridad están asociadas a cada TSFI?

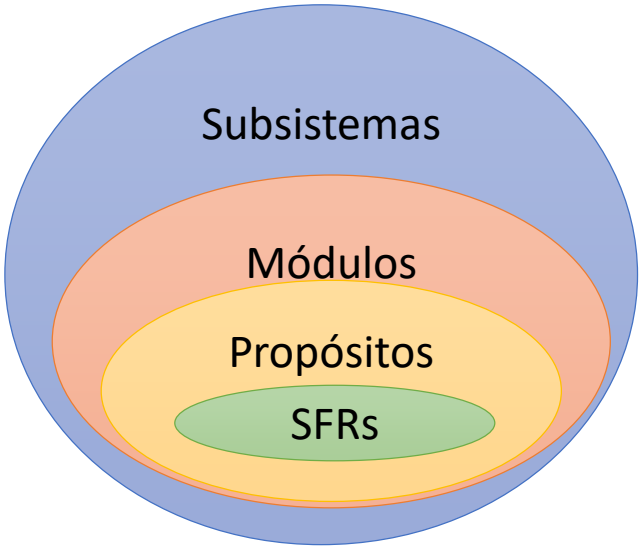
- Descripción completa: **propósito, método de uso, parámetros, mensajes de error y excepciones**.

Fase de Diseño – Diseño del TOE

- Por el requisito ADV_TDS de CC se dará un **diseño de los componentes** del TOE en términos de **subsistemas** y de **módulos** que lo componen, según el nivel de evaluación.



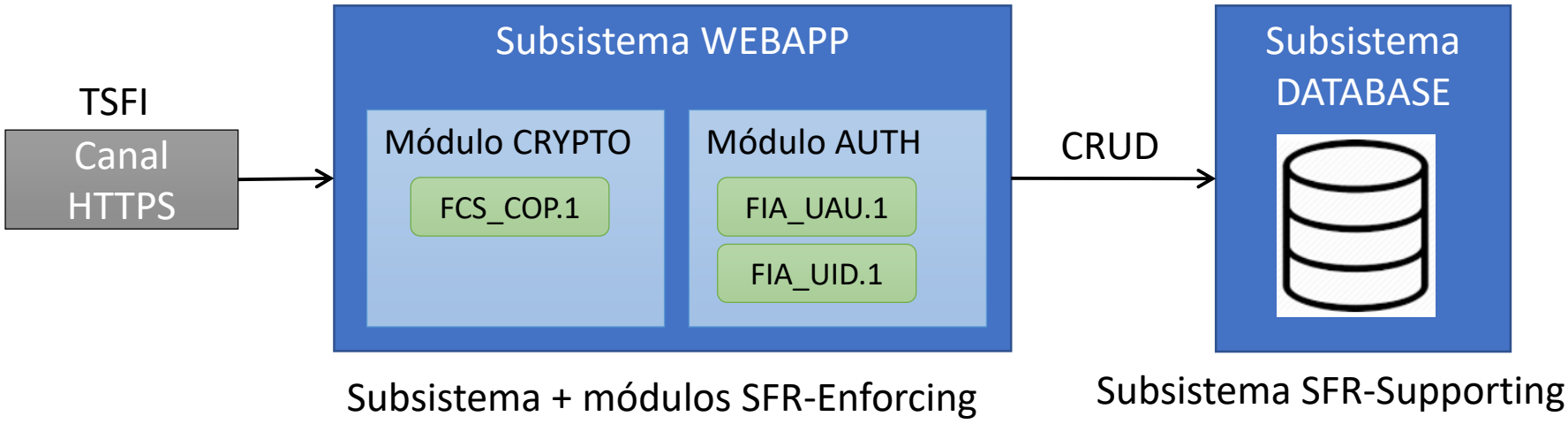
ADV_TDS.1, ADV_TDS.2



ADV_TDS.3 y superior

Fase de Diseño – Diseño del TOE

- Cómo participan los componentes del TOE en la seguridad:
  - Caracterización**: SFR-Enforcing, SFR-Supporting, SFR-Non-interfering**
  - Interacciones entre módulos.
  - Relación con las interfaces (TSFIs)



Fase de Diseño – Arquitectura de Seguridad

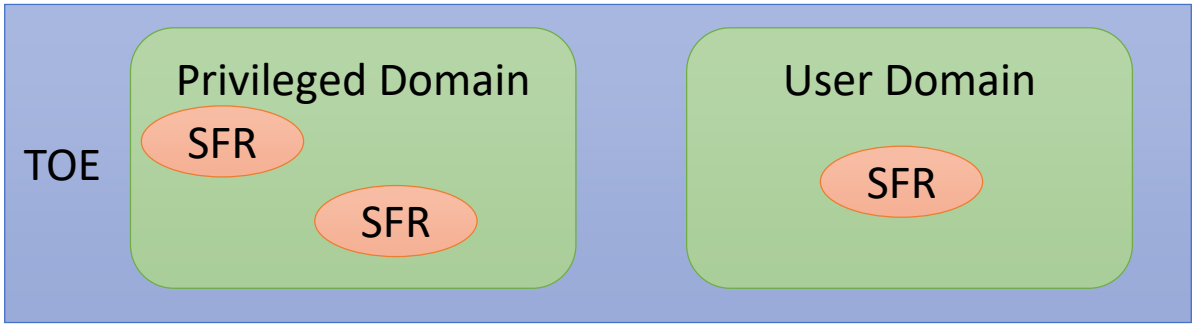
- Como parte de la fase de diseño se debe describir la arquitectura de seguridad del TOE (ADV_ARC).
- El propósito es garantizar que la funcionalidad de seguridad no se pueda ser evadida o alterada.

Dominios de seguridad	Inicialización segura
Auto-protección TSF	TSF Non-bypassability

Fase de Diseño – Arquitectura de Seguridad

Separación de dominios de seguridad

- Dominio de seguridad: colección de **recursos** a los que una entidad (p. ej. programa en ejecución) tiene **privilegios de acceso**.
- Ejemplos: Si hay varios procesos, el espacio de memoria de cada uno es un dominio de seguridad; Modo de ejecución con privilegios o sin privilegios
- Cada dominio de seguridad puede tener asociada una serie de funcionalidades de seguridad (SFRs)



Fase de Diseño – Arquitectura de Seguridad

Inicialización segura

- Se debe describir el proceso que sigue el producto desde que se inicia la ejecución hasta que alcanza un estado seguro.
- **Estado seguro:** toda la funcionalidad de seguridad (SFRs) está desplegada y operativa.
- Ejemplo:

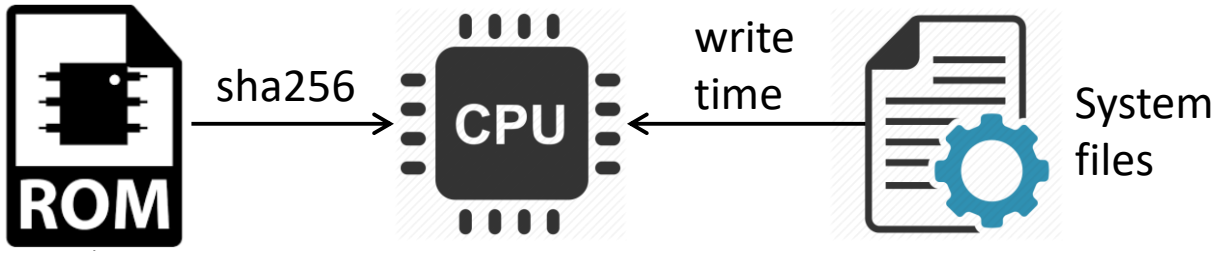
```
1.  Load crypto library
2.  Connect to encrypted database
3.  Initialize HTTPS server
```

- Permite detectar problemas de seguridad en la secuencia de inicialización desde la fase de diseño.

Fase de Diseño – Arquitectura de Seguridad

Protección contra alteración / self-protection vs tampering

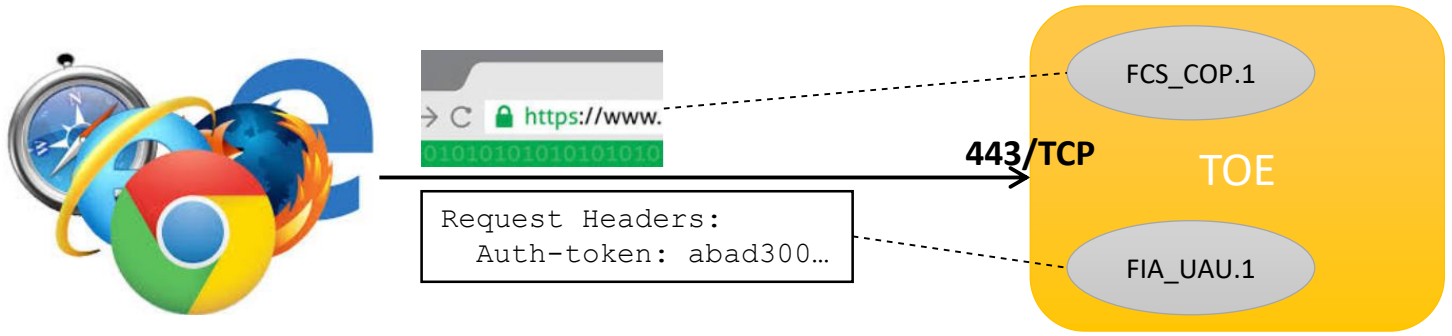
- Se justifica cómo se protege la **funcionalidad de seguridad** contra alteraciones externas que puedan **modificar su funcionamiento**.
- La principal vía de alteración es a través de las interfaces de uso de relacionadas con la seguridad (TSFIs).
- Se implementa a través de **mecanismos de auto-protección**, físicos (epoxy, mallas aislantes) o lógicos (checksums para integridad).



Fase de Diseño – Arquitectura de Seguridad

Non-bypassability: seguridad no sorteable

- Se debe justificar cómo se evita que se esquive (bypass) la funcionalidad de seguridad del producto, de manera que siempre sea invocada.
- TSFIs -> vías potenciales para el bypass de la TSF.
- Demostrar que los usos de las TSFIs implica que la **funcionalidad de seguridad siempre se invoca**.



Fase de Diseño

Re-cap: CC = seguridad desde el diseño

- **Descripción funcional (ADV_FSP):** especificación completa de las interfaces de uso + SFRs relacionados.
- **Diseño del TOE (ADV_TDS):** subsistemas / módulos / interacciones + SFRs relacionados
- **Arquitectura de seguridad (ADV_ARC):** dominios de seguridad, inicialización segura, auto-protección, non-bypassability

Fase de Implementación

- ¿Cómo ayuda Common Criteria a contemplar la seguridad durante la implementación o codificación?
- Por el requisito ALC_TAT se impone el uso de **herramientas y estándares** de desarrollo reconocidos así como la definición concreta de la **opciones dependientes de la implementación**.
- El requisito ADV_IMP obliga a presentar el **código fuente del producto para la evaluación**.
- Por el requisito AVA_VAN se lleva a cabo **búsqueda de vulnerabilidades** en el código fuente analizado

Fase de Implementación - Tools and Techniques

- ALC_TAT impone la utilización de **herramientas y estándares** de desarrollo bien definidos que produzcan **resultados consistentes y predecibles**.
- **Identificación bien definida** de las herramientas usadas en desarrollo: nombre, versión, etc. **Herramientas**:
- Herramientas: frameworks, compiladores, linkers, test-suites, hardware para testing, etc.
- Se requiere la definición de las **"implementation-dependent options"**: parámetros y flags de compilación y linkado.

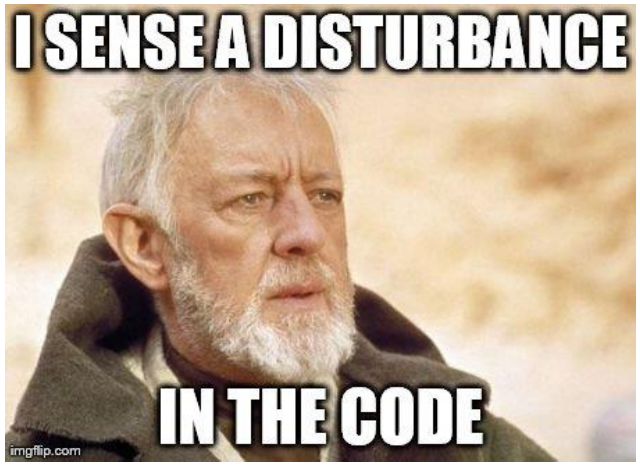
Fase de Implementación – Tools and Techniques

- Aplicando los requisitos de ALC_TAT pueden utilizarse metodologías de desarrollo conocidas y contrastadas, pues el requisito también habla de técnicas.

- Estas metodologías pueden contribuir a mejoras en la seguridad:
  - SDL (Microsoft)
  - BSIMM2
  - OpenSAMM

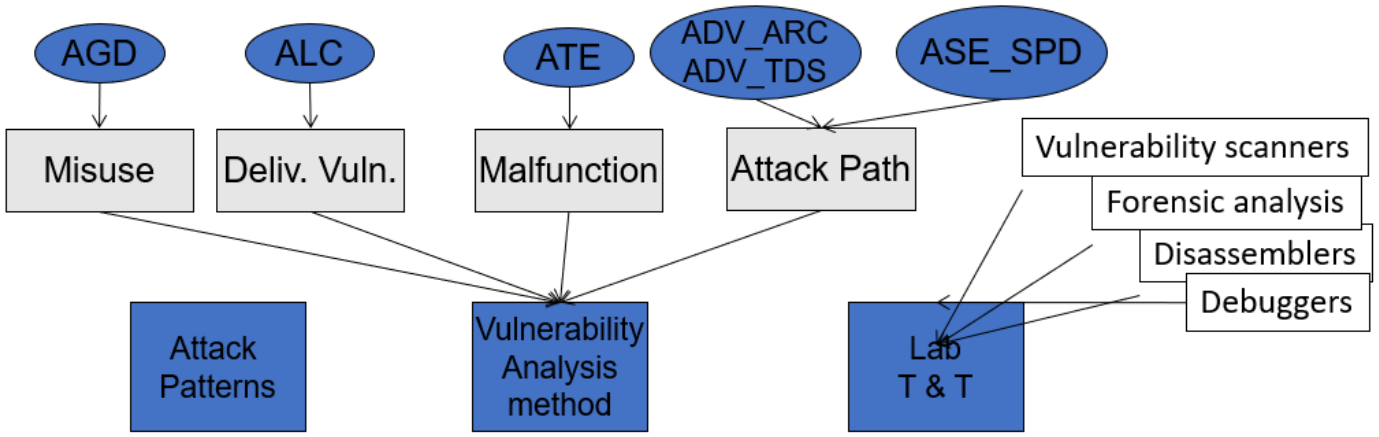
Fase de Implementación – Revisión de código

- ADV_IMP requiere que el código fuente se entregue para la evaluación. En la evaluación se comprobará:
  - Completitud del código y legibilidad
  - Correspondencia con el diseño
  - Generación exacta del TOE con las herramientas usadas (ALC_TAT)



Fase de Implementación - Vulnerabilidades

- AVA_VAN: vulnerabilidades en el código… y más allá.
- Los evaluadores realizarán una búsqueda de vulnerabilidades extensa y minuciosa en el producto evaluado. Gran número de actividades.
- Una de las actividades de AVA_VAN consiste en la búsqueda de vulnerabilidades a lo largo del código fuente mediante inspección



Fase de Testing

- Common Criteria incluye la **clase ATE** con requisitos de garantía para el testing del producto.
- **Objetivo**: determinar que el TOE se comporta como se ha especificado a través de los **SFRs** y de la **especificación funcional** dada en el diseño.
- Se consigue realizando un **plan de tests adecuado y completo**.

Fase de Testing – ATE_FUN + ATE_IND

- Plan de tests adecuado y completo

ATE_FUN: Functional tests

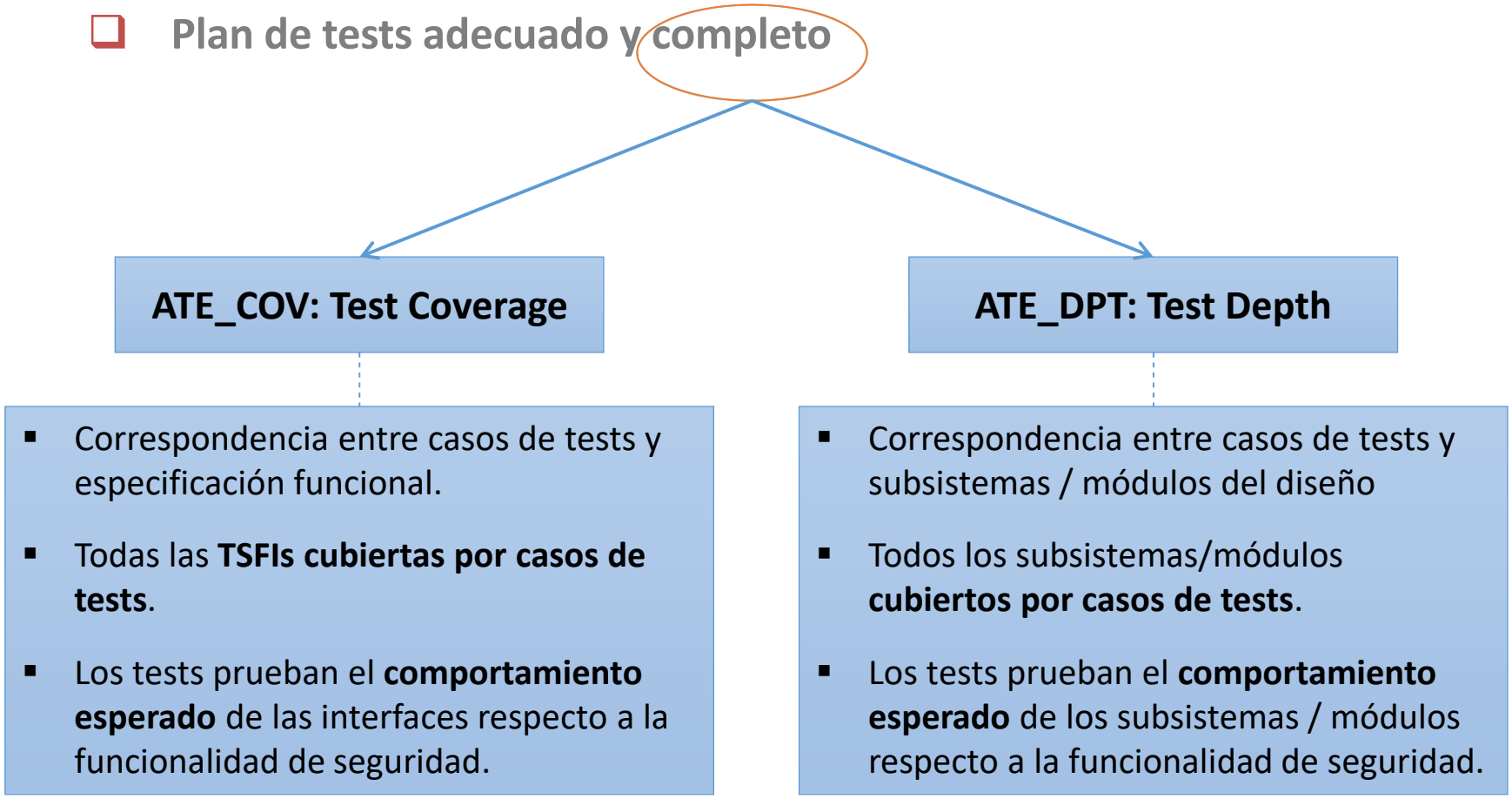
- Plan de pruebas con **conjunto de tests, resultados esperados y resultados obtenidos.**
- Descripción completa de los **escenarios de ejecución, orden de ejecución y dependencias** entre tests.
- Tests **repetibles** que arrojen los mismos resultados que en el plan de pruebas.

ATE_IND: Independent Tests

- Los evaluadores repetirán parte de los tests del plan de pruebas.
- Proporcionar producto + entorno adecuado para testing.
- Validación de las pruebas por un **tercero independiente**, que comprueba comportamiento y consistencia.

Fase de Testing – ATE_COV + ATE_DPT

- Plan de tests adecuado y completo



Fase de Mantenimiento

¿Cómo ayuda Common Criteria a la seguridad durante la fase de mantenimiento?

- CC contempla la fase de mantenimiento en la que se los defectos encontrados durante el uso y se implementan mejoras.

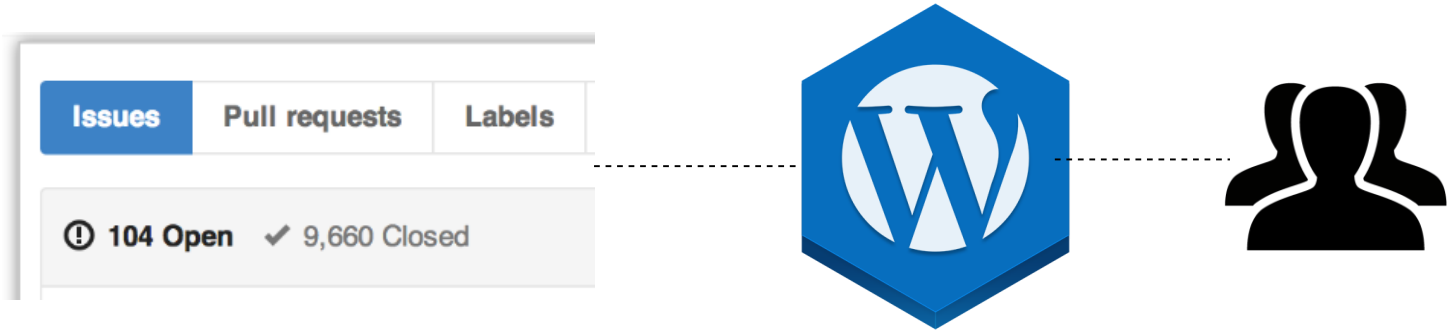
- Los componentes de la familia ALC_FLR imponen una serie de requisitos para establecer procedimientos de **seguimiento de defectos de seguridad, identificación de acciones correctivas y distribución de correcciones** a los usuarios.

Fase de Mantenimiento – Flaw Remediation

El requisito de garantía ALC_FLR, que trata corrección de fallos, establece una serie de requisitos durante para la fase de mantenimiento:

- Deben existir procedimientos para el **tracking de fallos de seguridad** en cada release del TOE. Se producirán **acciones correctivas** para cada fallo.

- Se tendrán mecanismos para **notificar a los usuar**ios de los fallos de seguridad y correcciones, proporcionando las guías adecuadas.



Fase de Mantenimiento – Flaw Remediation

- Requisitos de seguridad y calidad en la gestión de fallos:
  - Asegurar que **todos los defectos de seguridad sean tratados**.
  - Asegurar que las correcciones de un problema **no introducen otros problemas** (ej. Test regresión)
  - Canales de comunicación con los usuarios.
  - **Distribución activa y controlada** de correcciones a los usuarios.

Fase de Mantenimiento – Documentación

- Otro factor de Common Criteria que potencia la seguridad durante el mantenimiento del producto es la **extensa y completa documentación** que se genera al seguir la norma.

- Las acciones de mantenimiento serán más efectivas y seguras al disponer de **gran cantidad de documentación** con información necesaria y útil para el mantenimiento:
  - Declaración de seguridad
  - Especificación funcional
  - Diseño del TOE
  - Arquitectura de seguridad
  - Planes de pruebas
  - Guías operativas y del entorno
  - Guías de entrega, gestión de la configuración, etc.

Fase de Mantenimiento

re-cap: CC incorpora seguridad en todas las fases del desarrollo:

- **Análisis**: problema de seguridad, objetivos de seguridad, requisitos funcionales de seguridad.
- **Diseño**: especificación funcional, diseño del TOE, arquitectura de seguridad.
- **Implementación**: herramientas y técnicas, revisión de código, búsqueda de vulnerabilidades
- **Pruebas**: cobertura de los tests, profundidad de los tests, pruebas funcionales, testing independiente.
- **Mantenimiento**: gestión de reporte de fallos, documentación extensa y completa

Recursos para el desarrollador

- *Guidelines for Developer Documentation according to Common Criteria Version 3.1*

  http://www.commoncriteriaportal.org/files/ccfiles/CommonCriteriaDevelopersGuide_1_0.pdf)

- Guía para afrontar el desarrollo del producto teniendo en cuenta las actividades y requisitos de Common Criteria.

- Está pensada principalmente para desarrollar productos que se pretenden certificar en Common Criteria.

- Es una buena guía para incorporar la seguridad a lo largo de todo el proceso de desarrollo, tal y como se ha ido explicando en esta presentación.

Seguridad más allá del desarrollo

Seguridad más allá del desarrollo

- Common Criteria contempla la seguridad en el producto de manera integral durante todo el **ciclo de vida**.
- Las actividades de la clase ALC contemplan la seguridad en las distintas fases del **ciclo de vida de un producto, durante y después del desarrollo**.
- Las actividades de la clase AGD contemplan requisitos para la **seguridad en la documentación y guías de usuario**.

Seguridad en el ciclo de vida

Plan de gestión de la configuración (ALC_CMC)

- Trata **la gestión de los ítems usados durante el desarrollo** y tras la primera release: ficheros de código fuente, documentación, releases, librerías, etc.
- Impone el uso de **un sistema de identificación y etiquetado de versiones** del producto y sus componentes (reglas + herramientas).
- Utilización de **técnicas y herramientas** para crear nuevas versiones, reemplazar versiones existentes, generar releases, etc. (SVN, GIT, + normas de uso)
- Gestión del **control de cambios** en todos los ítems.

Seguridad en el ciclo de vida

Lista de ítems de configuración (ALC_CMS)

- Requiere mantener un **registro detallado de los ítems** existentes de cara a la evaluación Common Criteria, incluyendo nombre, tipo, versión y descripción.
- Ítems para la evaluación: TOE, sus partes, y documentación CC. En niveles altos de evaluación, código fuente, y documentación de relacionada con notificación de defectos (ALC_FLR)

Procedimiento de entrega segura (ALC_DEL)

- Deben existir **procedimientos para entregar el producto** a cliente, fabricantes, integradores, etc. **de manera segura**.
- Se deben dar indicaciones para seguridad en el transporte, identificación fehaciente del producto entregado, etc.

Seguridad en el ciclo de vida

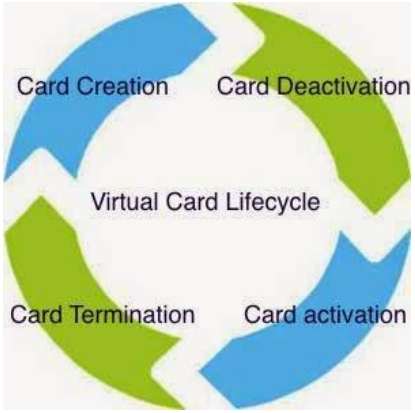
Seguridad en el entorno de desarrollo (ALC_DVS)

- Exige que se cumplan requisitos de seguridad en los sitios y entornos donde se desarrolla el producto.
- Procedimientos para aspectos relevantes de seguridad:
  - Seguridad perimetral, acceso físico, zonas seguras.
  - Seguridad en el acceso lógico a la información: políticas de control de acceso, gestión de permisos, cuentas de usuario, etc.
  - Seguridad en el personal: procesos de contratación, altas y baja de usuarios y permisos, devolución de activos, etc.
  - Seguridad en las operaciones: acceso a las redes, backups, destrucción de medios de almacenamiento, etc.
- En niveles altos de evaluación, se requiere realizar **auditorías presenciales en los sitios de desarrollo.**

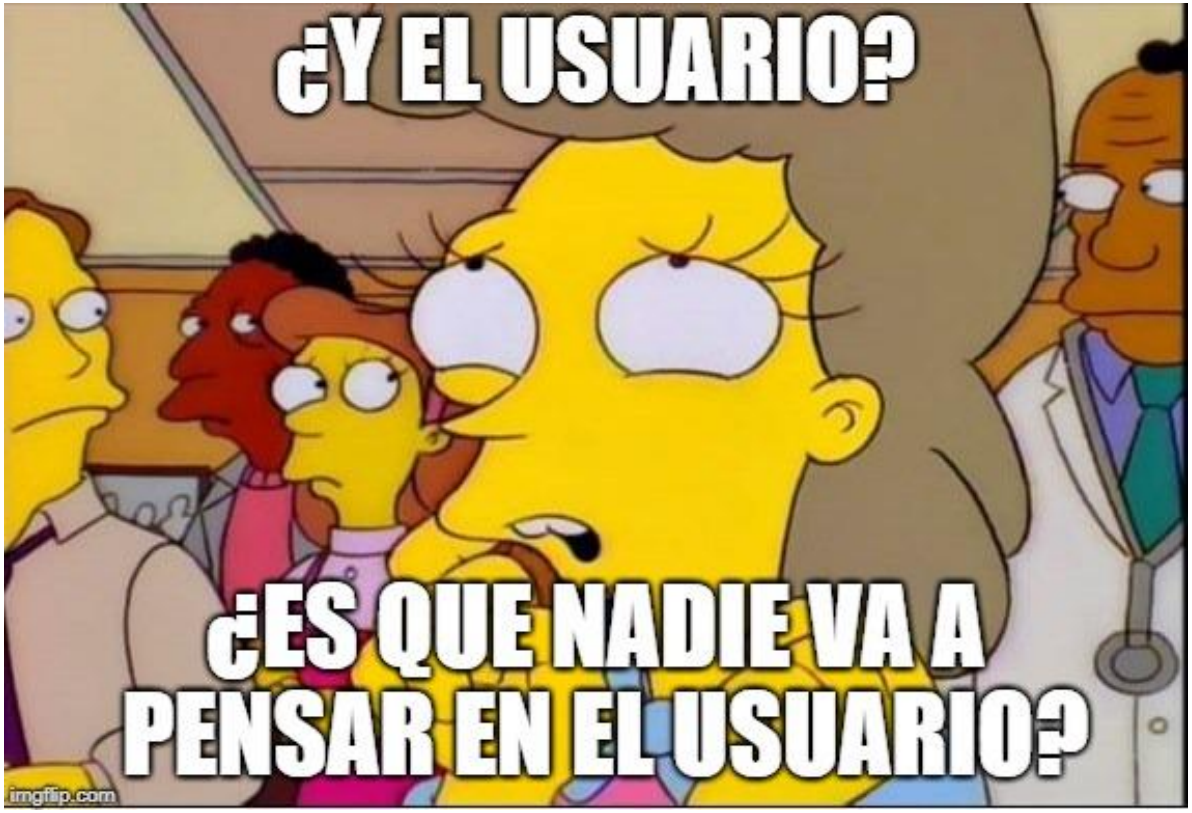
Seguridad en el ciclo de vida

Gestión del ciclo de vida (ALC_LCD)

- Se debe especificar y documentar las fases del ciclo de vida del producto, no sólo en lo referente a desarrollo.
- **Instrucciones de seguridad para cada fase** y para las transiciones entre fases.
- Aplica más en determinados tipos de producto, por ejemplo si intervienen varios actores en la producción.



Seguridad y guías de uso



Seguridad y guías de uso

Guías de instalación y preparación del entorno (AGD_PRE):

- Instrucciones de seguridad para **instalación, configuración y preparación del entorno** de manera que el **TOE quede en un estado seguro**. Ejemplos: instalar software, instalar certificado, cambiar clave de acceso por defecto.
- **Instrucciones seguridad para la aceptación segura del TOE**. Ejemplos: comprobar SHA del fichero descargado, comprobar firmas del certificado con el que se ha firmado el ejecutable.

Guías de uso operacional (AGD_OPE)

- Instrucciones para uso **seguro del entorno operacional**. Ej: dar formación a los usuarios administradores del software.
- Instrucciones para **uso seguro de las interfaces del TOE**. Ej: utilizar una contraseña fuerte, cerrar sesión en la web al f

Perfiles de protección

Perfiles de protección

- **Los perfiles de protección** son “plantillas” de Common Criteria para determinados tipos de producto que afrontan la **solución a un problema de seguridad parecido** con una aproximación similar.

Perfiles de protección

- Hay PPs para sistemas operativos, dispositivos de firma digital, dispositivos de acceso a la red, bases de datos, etc.

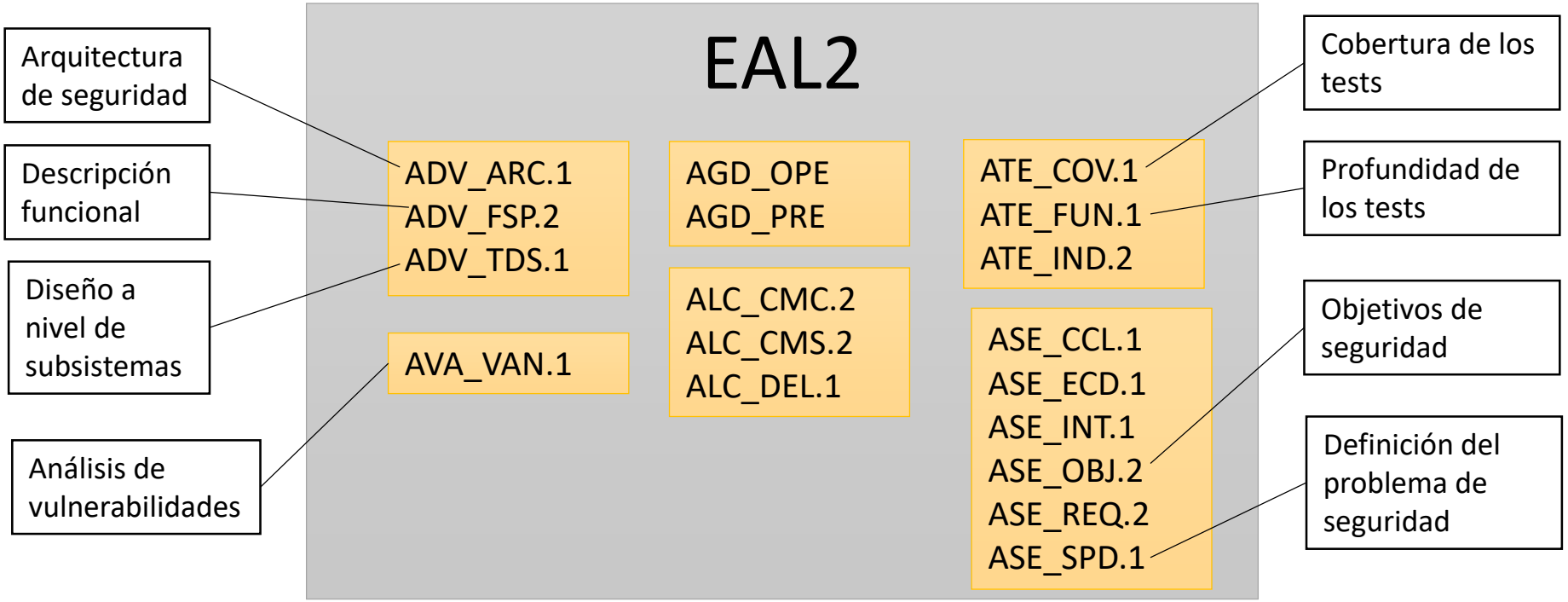
  https://www.commoncriteriaportal.org/pps/

- ⊞ Access Control Devices and Systems – 3 Protection Profiles
- ⊞ Biometric Systems and Devices – 2 Protection Profiles
- ⊞ Boundary Protection Devices and Systems – 11 Protection Profiles
- ⊞ Data Protection – 10 Protection Profiles
- ⊞ Databases – 3 Protection Profiles
- ⊞ ICs, Smart Cards and Smart Card-Related Devices and Systems – 71 Protection Profiles
- ⊞ Key Management Systems – 4 Protection Profiles
- ⊞ Mobility – 4 Protection Profiles
- ⊞ Multi-Function Devices – 2 Protection Profiles
- ⊞ Network and Network-Related Devices and Systems – 12 Protection Profiles
- ⊞ Operating Systems – 2 Protection Profiles
- ⊞ Other Devices and Systems – 48 Protection Profiles
- ⊞ Products for Digital Signatures – 19 Protection Profiles
- ⊞ Trusted Computing – 6 Protection Profiles

Perfiles de protección

Un perfil de protección incluye (1):

- Un **nivel de garantía de evaluación (EAL).** Consiste en el conjunto de Common Criteria que implican actividades documentales, de evaluación, de ciclo de vida, etc.



Perfiles de protección

Un perfil de protección incluye (2):

- Una descripción general (overview) del tipo de producto de seguridad al que se refiere el perfil de protección.
- Cada producto (TOE) que sea conforme con un perfil de protección deberá dar una descripción detallada del producto, con los detalles específicos que no están en la descripción data por el PP.

Ejemplo: BSI-CC-PP-0088-V2 (Database Management Systems)

2 TOE DESCRIPTION

2.1 Product Type

2.2 TOE Definition

2.3 Security Functionality Provided by the TOE

2.4 Optional Security Functionality

2.5 TOE Operational Environment

2.5.1 Enclave

2.5.2 TOE Architectures

2.5.3 TOE Administration

2.1 Product Type

The product type of the Target of Evaluation (TOE) described in this Protection Profile (PP) is a database management system (DBMS).

A DBMS is a computerized repository that stores information and allows authorized users to retrieve and update that information. A DBMS may be a single-user system, in which only one user may access the DBMS at a given time, or a multi-user system, in which many users may access the DBMS simultaneously.

Perfiles de protección

Un perfil de protección incluye (3):

- La **definición del problema de seguridad**. Se da una lista ya fijada de activos, sujetos, amenazas, políticas e hipótesis.
- Los **objetivos de seguridad** del TOE y del entorno. Los objetivos de seguridad ya se dan fijados para un tipo de TOE.
- La lista de SFRs: **requisitos funcionales de seguridad**. Los requisitos funcionales de seguridad ya se dan fijados para un tipo de TOE.

Perfiles de protección

Aplicación al desarrollo de los PPs

- La definición del **problema de seguridad**. Se da una lista ya fijada de activos, sujetos, amenazas, políticas e hipótesis.
- Los **objetivos de seguridad** del TOE y del entorno. Los objetivos de seguridad ya se dan fijados para un tipo de TOE.
- La lista de SFRs: **requisitos funcionales de seguridad**. Los requisitos funcionales de seguridad ya se dan fijados para un tipo de TOE.

Catálogo de productos STIC (CPSTIC)

Catálogo de productos STIC (CPSTIC)

- El CPSTIC es el **catálogo de referencia creado por el CCN** en el que se incluyen productos TIC con **certificación de seguridad** para su adquisición por la **Administración Pública**.
- **Productos cualificados**: aptos para usar en sistemas afectados por el ENS con categoría ALTA.
- **Productos aprobados**: aptos para manejar información clasificada.
- Es **requisito** para la inclusión pasar una certificación **Common Criteria** con los SFRs y SARs indicados para la taxonomía propia del producto.
- Cada taxonomía de producto tendrá unos requisitos diferentes en la certificación CC para poder ser incluido en el catálogo.

CONCLUSIONES

- Common Criteria es un estándar reconocido internacionalmente que permite evaluar el nivel de garantía en la seguridad de un producto.
- Desarrollar un producto contemplando los requisitos de Common Criteria introduce consideraciones de seguridad en todas las fases del desarrollo.
- Common Criteria contempla la seguridad más allá del desarrollo, teniendo en cuenta las distintas fases del ciclo de vida del producto.
- Los perfiles de protección permiten partir del análisis previo de un problema parecido que ya ha sido resuelto desde el punto de vista de la seguridad.
- Confianza para las administraciones: catálogo CPSTIC

Datos de contacto

jtsec: Beyond IT Security

c/ Abeto s/n Edificio CEG Oficina 2B

CP 18230 Granada – Atarfe – Spain

hello@jtsec.es

@jtsecES

www.jtsec.es