Análisis, diseño y ejecución de pruebas a la capa de seguridad del Sistema de Información para el Sistema de Abastecimiento y Seguridad Alimentaria de Bogotá
SI-SAAB fase 2006

Carlos Francisco Álvarez Gallardo

DIRECTOR
Ing. Msc. Henry Diosa

CODIRECTORA
Msc. Zulima Ortiz

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
FACULTAD DE INGENIERÍA
PROYECTO CURRICULAR INGENIERÍA DE SISTEMAS
BOGOTÁ D.C.
2008

AGENDA

- Definición del proyecto de pruebas de seguridad
  - Introducción
  - Formulación del problema
  - Objetivos
  - Justificación
  - Alcances y limitaciones
- Marco referencial, conceptual y teórico
  - Definición del proyecto SI-SAAB
  - Pruebas de software
  - Aspectos relevantes sobre el diseño de seguridad y patrones de seguridad
  - Pruebas de seguridad de software
- Diseño del proceso de pruebas de seguridad
  - Modelos utilizados en el diseño del proceso de pruebas de seguridad
  - Modelo de pruebas de seguridad
  - Integración del proceso de pruebas de seguridad al proceso de pruebas general de SI-SAAB
- Implementación del proceso de pruebas de seguridad
  - Definición del proceso de pruebas de seguridad
  - Definir la estrategia y ambiente de pruebas de seguridad
  - Definir las pruebas de seguridad
  - Ejecución de pruebas de seguridad
  - Mejorar los artefactos de pruebas de seguridad
- Conclusiones
- Referencias

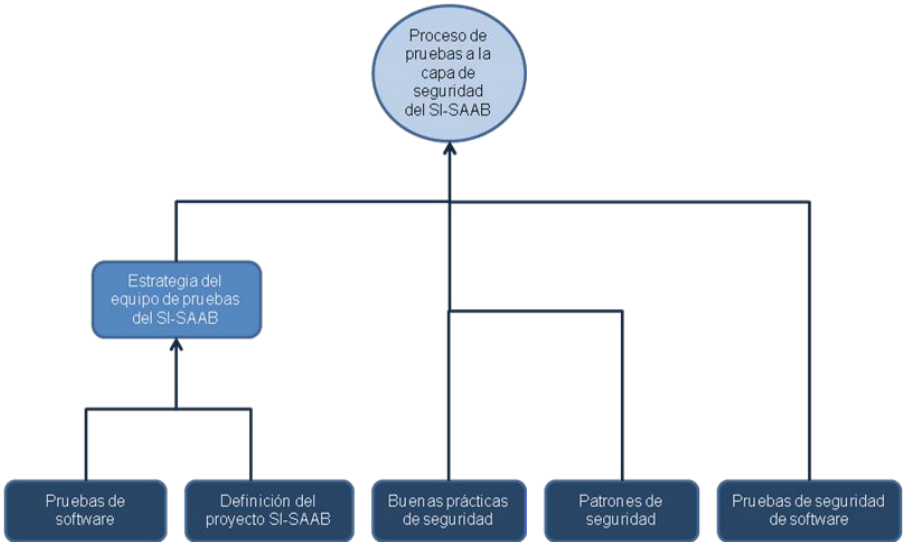
DEFINICIÓN DEL PROYECTO DE PRUEBAS DE SEGURIDAD

INTRODUCCIÓN

Cuando se habla de seguridad en software se habla de un tema de actualidad y que está tomando gran importancia dentro del desarrollo soluciones. Así mismo, la etapa de pruebas en el desarrollo de software se está convirtiendo en una etapa que está cobrando importancia desde las etapas tempranas del modelo de proceso aplicado; sin embargo, particularmente al momento de realizar pruebas a la capa de seguridad se encuentran fuertes vacíos en el ejercicio real del desarrollo de validaciones en sistemas de información.

Es por esta razón que, para poder generar un proceso aplicado de pruebas a la capa de seguridad del SI-SAAB, se ve la necesidad de estudiar algunos temas que sirven como base para poder aplicar o generar enfoques metodológicos en desarrollos de software reales y que permitan ganar experiencia, para generar un proceso acorde con las exigencias de este proyecto.

Dada la naturaleza y dimensión que puede abarcar el proyecto es necesario comprender los conceptos generales que describen al SI-SAAB y los desarrollos con terceros, así como las consideraciones técnicas y tecnológicas referentes a la capa de seguridad y la metodología usada por el equipo de pruebas. Tomando como línea base de análisis los estándares y situación actual sobre el desarrollo de pruebas de software y el desarrollo de software seguro.



Pirámide de definiciones del proceso de pruebas de seguridad para SI-SAAB

FORMULACIÓN DEL PROBLEMA

El problema a tratar es la generación de pruebas a la capa de seguridad del SI-SAAB y su integración dentro del proceso de pruebas de software que se aplica actualmente a cada uno de los subsistemas desarrollados en la fase 2006.

- Las pruebas que se realizan a un sistema sólo se orientan a verificar y validar su funcionalidad mas no a detectar las deficiencias de seguridad [Ness, 2005]

- Como la capa de seguridad en un sistema generalmente está asociada a los requerimientos no funcionales [Kruchten, 2003], sólo se realizan algunos controles de seguridad, de acceso y de protección de datos, pero no se prueba la validez de estas implementaciones.

- Con las pruebas a esta capa se busca comprobar que todo el sistema en su conjunto, sea estable, seguro y confiable, dando por hecho que los diseños de seguridad generados fueron implementados correctamente.

- Es necesario generar una fase de pruebas documentada, según el plan maestro de pruebas, que se centre en verificar que las implementaciones hechas por el equipo en la capa de seguridad satisfacen las necesidades que se pueden tener del software referentes a este aspecto, y que todas las políticas y guías generadas fueron implementadas correctamente.

- Exige analizar metodologías existentes para utilizarlas en la fase de pruebas e incluso, de ser necesario, proponer un desarrollo interno procedimental y aplicado para realizar este tipo de pruebas en el SI-SAAB que pueda servir de estándar para futuras iteraciones.

OBJETIVOS

GENERAL

- Analizar, diseñar y ejecutar pruebas a la capa de seguridad del SI-SAAB desarrollados en la fase 2006 por la Universidad Distrital Francisco José de Caldas.

ESPECÍFICOS

- Evaluar la utilidad de la documentación sobre el proceso de pruebas de seguridad de software y su integración a un proceso general de pruebas de software para definir si se puede adaptar una metodología existente al diseño y aplicación de pruebas a la capa de seguridad del SI-SAAB. En caso contrario, proponer un desarrollo interno procedimental y aplicado de dichas pruebas.
- Analizar los posibles ataques y vulnerabilidades que hayan sido contemplados por el equipo de seguridad del SI-SAAB.
- Analizar los diseños e implementaciones para la capa de seguridad generados por el equipo de seguridad del SI-SAAB.
- De acuerdo a la metodología escogida o el proceso generado establecer un estándar dentro del equipo de pruebas del SI-SAAB sobre la forma de diseñar las pruebas a la capa de seguridad y que pueda ser integrado dentro del plan de pruebas establecido por dicho equipo.
- De acuerdo al estándar establecido diseñar pruebas a la capa de seguridad del SI-SAAB.
- Ejecutar pruebas de seguridad diseñadas para el SI-SAAB.
- Retroalimentar al equipo de pruebas y al equipo de seguridad sobre los defectos encontrados durante la ejecución de las pruebas de seguridad.
- Publicar los resultados de este proyecto para aportar a la comunidad académica y técnica un enfoque de diseño y aplicación práctico sobre pruebas a la capa de seguridad de sistemas de información.

JUSTIFICACIÓN

- En el proyecto SI-SAAB y en general en cualquier tipo de proyecto de software es necesario implementar una capa de seguridad para asegurar la confiabilidad e integridad del sistema. Cuando se generan estas implementaciones se debe asegurar de alguna manera que efectivamente fueron desarrolladas correctamente y que además cubren las diferentes vulnerabilidades que se podrían presentar en el sistema.

- Cuando estos desarrollos de seguridad son hechos por un grupo de personas, es conveniente que un grupo diferente se encargue de probar los diseños generados por el equipo de seguridad para poder tener puntos de vista distintos y que no hayan sido considerados por el grupo de seguridad; lo anterior, con el objeto de retroalimentar sobre los defectos encontrados y así poder desarrollar un sistema más seguro, confiable y eficiente.

- El proyecto SI-SAAB ha planteado la necesidad de probar que el sistema efectivamente es seguro y que en su puesta en producción se minimicen riesgos de privacidad, confidencialidad, no repudiación e integridad generados en la capa de seguridad del sistema.

- En el SI-SAAB es necesario generar un conjunto de pruebas a la capa de seguridad que se puedan integrar al plan de pruebas general desarrollado originalmente, así mismo, un proceso para el desarrollo de estas que pueda ser documentado formalmente para una posterior capacitación de personal que se encargue de esta área y/o desarrollos futuros.

ALCANCES Y LIMITACIONES

Alcances

El proyecto pretende desarrollar un proceso de pruebas de seguridad que pueda ser implementado dentro del SI-SAAB e integrado dentro del plan de pruebas del sistema desarrollado. Los alcances puntuales que se tendrán como referencia para el desarrollo de este proyecto son:

- Implementación y/o diseño de un proceso que sirva como estándar interno para la generación de pruebas a la capa de seguridad en el SI-SAAB fase 2006.
- Integración del proceso establecido al plan de pruebas del SI-SAAB.
- Análisis sobre las implementaciones diseñadas para la capa de seguridad de la fase 2006 por el equipo de seguridad del proyecto.
- Generación de casos de prueba para la capa de seguridad en los subsistemas de negociación del SI-SAAB fase 2006 y para los controles de acceso establecidos.
- Ejecución de las pruebas de seguridad referentes a los controles de acceso y el subsistema de negociación.
- Generación de los reportes de los resultados y defectos encontrados en la ejecución de las pruebas mencionadas anteriormente.

ALCANCES Y LIMITACIONES

Limitaciones

Las limitaciones contempladas para el desarrollo de este proyecto son las siguientes:

- Los diseños de pruebas serán generados con base a la capa de seguridad diseñada para el SI-SAAB fase 2006, pero no se considerarán criterios de seguridad adicionales que no hayan sido tenidos en cuenta en dicha capa.
- Sólo se podrá trabajar con las herramientas que se estén utilizando dentro del equipo de pruebas del SI-SAAB.
- Para el desarrollo del proyecto sólo se podrá consultar y referenciar los documentos autorizados para ser utilizados con este fin por el grupo del SI-SAAB.

MARCO REFERENCIAL, CONCEPTUAL Y TEÓRICO

DEFINICIÓN DEL PROYECTO SI-SAAB

DESCRIPCIÓN DEL SI-SAAB

- Nace de las necesidades informáticas presentadas por el Sistema de Abastecimiento y Seguridad Alimentaria para Bogotá (SAAB) en cuanto al soporte y el apoyo en el control de la calidad y de la transparencia de los flujos de información en los diferentes procesos y servicios diseñados por el equipo interdisciplinario de la Universidad Distrital [Isaza, et al., 2006].

- Este equipo interdisciplinario se conforma en el año 2005 con la vinculación del grupo de investigación ARQUISOFT al proyecto de “Modelamiento integral de la operación logística de las UPZs Lucero-Tesoro” liderado por el Grupo de Investigación GICIC para apoyar el diseño del componente denominado Sistema de Información [Isaza, et al., 2006].

- Conjunto de herramientas de software, hardware y comunicaciones interoperables entre sí con el fin de apoyar todos los procesos operativos, de gestión, de integración administrativos y de flujos de información involucrados en el SAAB y definidos en los documentos de Diseño del Modelo Logístico para el SAAB, realizados por la Universidad Distrital y la UESP (Unidad Ejecutiva de Servicios Públicos) en el marco del Convenio No. 11 del año 2005.

DEFINICIÓN DEL PROYECTO SI-SAAB

DESCRIPCIÓN DEL SI-SAAB

- Componente necesario para el correcto funcionamiento de los procesos operativos, de gestión, administración, integración y control del SAAB, con miras a obtener transparencia, efectividad, eficiencia, seguridad de acceso, confidencialidad, confiabilidad dando soporte a toda la información que fluye por la cadena de abastecimiento y las interacciones de ésta con los diferentes actores del sistema. [Isaza, et al., 2006]

- Carácter público y estatal para responder a las necesidades de información en el abastecimiento y seguridad alimentaria de los usuarios y participantes vinculados y externos del sistema además de las entidades Distritales y Nacionales de aseguramiento de calidad en las políticas de abastecimiento [Isaza, et al., 2006].

- En la fase 2005 del proyecto se genera una aproximación a un modelo de referencia arquitectural de software del SI-SAAB que permitiera gestionar de manera ordenada el análisis, diseño, implementación, puesta en producción y posterior apoyo en la evolución de los sistemas y subsistemas propios del mismo [Isaza, et al., 2006].

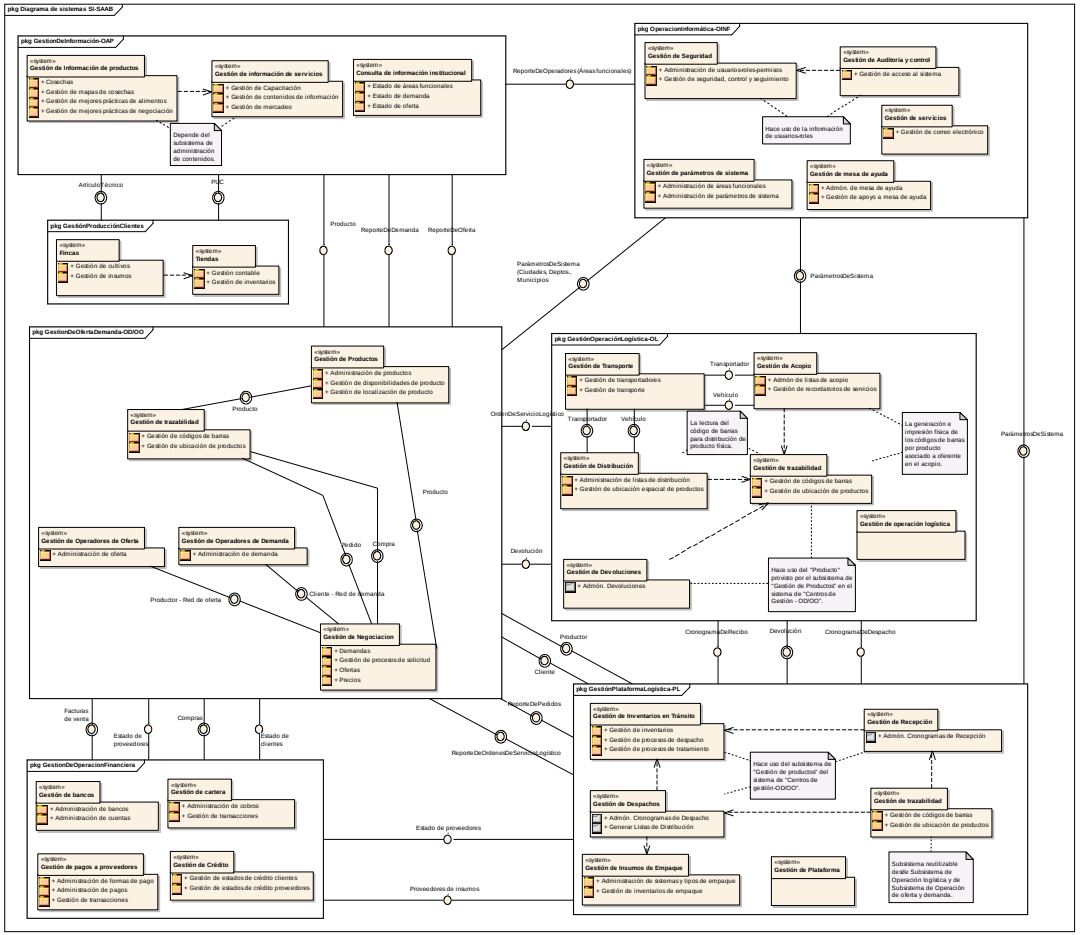
- Dentro de la fase 2006 de acuerdo al modelo incremental e iterativo planteado en el SI-SAAB se trabajan la primeras dos bases incrementales las cuales culminan la fase 2006 con la entrega del primer prototipo operacional [Isaza, et al., 2006].

DEFINICIÓN DEL PROYECTO SI-SAAB

ÁREAS FUNCIONALES DEL SI-SAAB

Las áreas funcionales definidas dentro de la fase 2006 del SI-SAAB son: [Isaza, et al., 2006]

- *Operación logística*
- *Plataforma logística*
- *Centros integrales de gestión de abastecimiento y seguridad alimentaria urbano/rural*
- *Operación institucional*
- *Operación de demanda*
- *Operación de oferta*
- *Operación informática*
- *Operación de acceso público*
- *Operación financiera*



DEFINICIÓN DEL PROYECTO SI-SAAB

SISTEMAS Y SUBSISTEMAS

Se identifican 7 subsistemas a partir de la definición de las áreas funcionales, actores, necesidades presentadas y funcionalidades del sistema son:.

- *Gestión de información*
  - Gestión de información de productos
  - Gestión de información de servicios
  - Gestión de información institucional
- *Gestión de operación logística*
  - Gestión de transporte
  - Gestión de acopio
  - Gestión de distribución
  - Gestión de devoluciones
  - Gestión de trazabilidad
- *Gestión de plataforma logística*
  - Gestión de inventarios en transito
  - Gestión de la recepción
  - Gestión de despachos
  - Gestión de insumos de empaque y embalaje
- *Gestión de oferta y demanda*
  - Gestión de operadores de demanda
  - Gestión de operadores de oferta
  - Gestión de procesos de solicitud
  - Gestión de productos
  - Gestión de negociación
  - Gestión de capacitación
  - Cosechas
- *Operación informática*
  - Gestión de seguridad
  - Gestión de auditoría y control
  - Gestión de maestros de sistema
  - Gestión de recuperación
- *Gestión de producción y clientes*
  - Gestión de fincas
  - Gestión de tiendas
- *Gestión financiera*
  - Gestión de bancos
  - Gestión de cartera
  - Gestión de pagos a proveedores
  - Gestión de crédito

PRUEBAS DE SOFTWARE

DEFINICIONES

- *Pruebas:*

  - Una actividad en la cual un sistema o uno de sus componentes se ejecutan en circunstancias previamente especificadas, los resultados se observan y registran y se realiza una evaluación de algún aspecto [Piattini, 2004].

  - El nombre "prueba", además de la actividad de probar, se puede utilizar para designar "un conjunto de casos y procedimientos de prueba“ [IEEE, 1990].

  - Para Myers, probar es el "proceso de ejecutar un programa con el fin de encontrar errores“ [Myers, 2004].

  - Según Pressman prueba es la ejecución de una aplicación, o un trozo de código, para identificar uno o varios errores [Pressman, 2004].

  - En conclusión, probar es el proceso de operar un sistema o componente bajo ciertas condiciones de forma tal que los resultados que arroja sirvan para observarlos y con base en ellos realizar evaluaciones. También se deriva de estas definiciones que las pruebas no sólo se efectúan al sistema sino que pueden efectuarse a cualquier componente o artefacto del proceso de desarrollo, por ejemplo, la documentación.

PRUEBAS DE SOFTWARE

DEFINICIONES

- *Caso de prueba:* el caso de prueba es una especificación en la cual se describe detalladamente cómo se realizará una prueba.

- *Dato de prueba*: los datos de prueba son las entradas seleccionadas para probar el sistema. Dichos datos algunas veces se generan de forma automática.

- *Error*: La diferencia entre un valor calculado, observado o medido y el valor verdadero, especificado o teóricamente correcto. [IEEE, 1990]

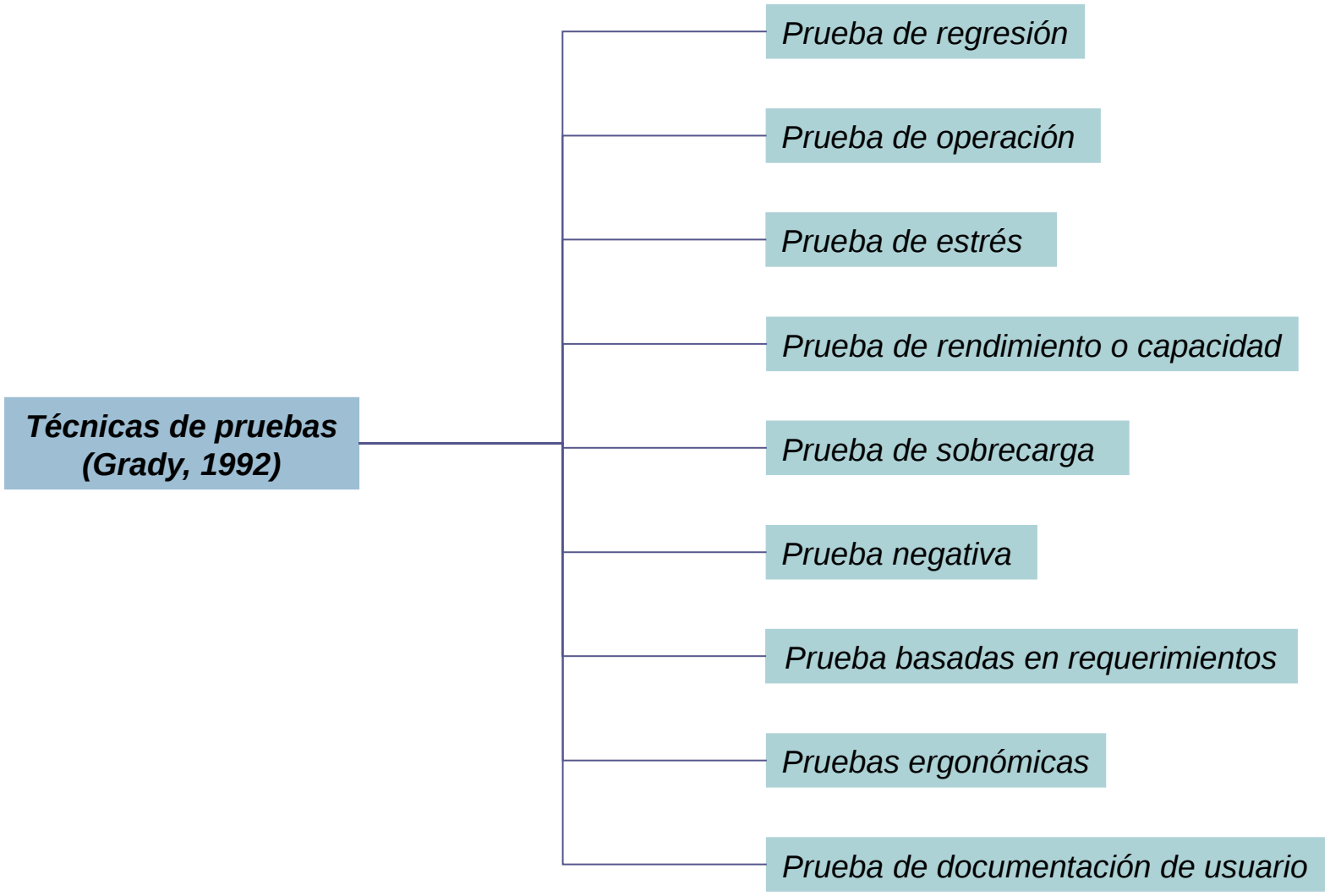
- *Defecto:*

  - Un paso, proceso o definición de dato incorrecto en un programa de computadora. El resultado de una equivocación. [IEEE, 1990].

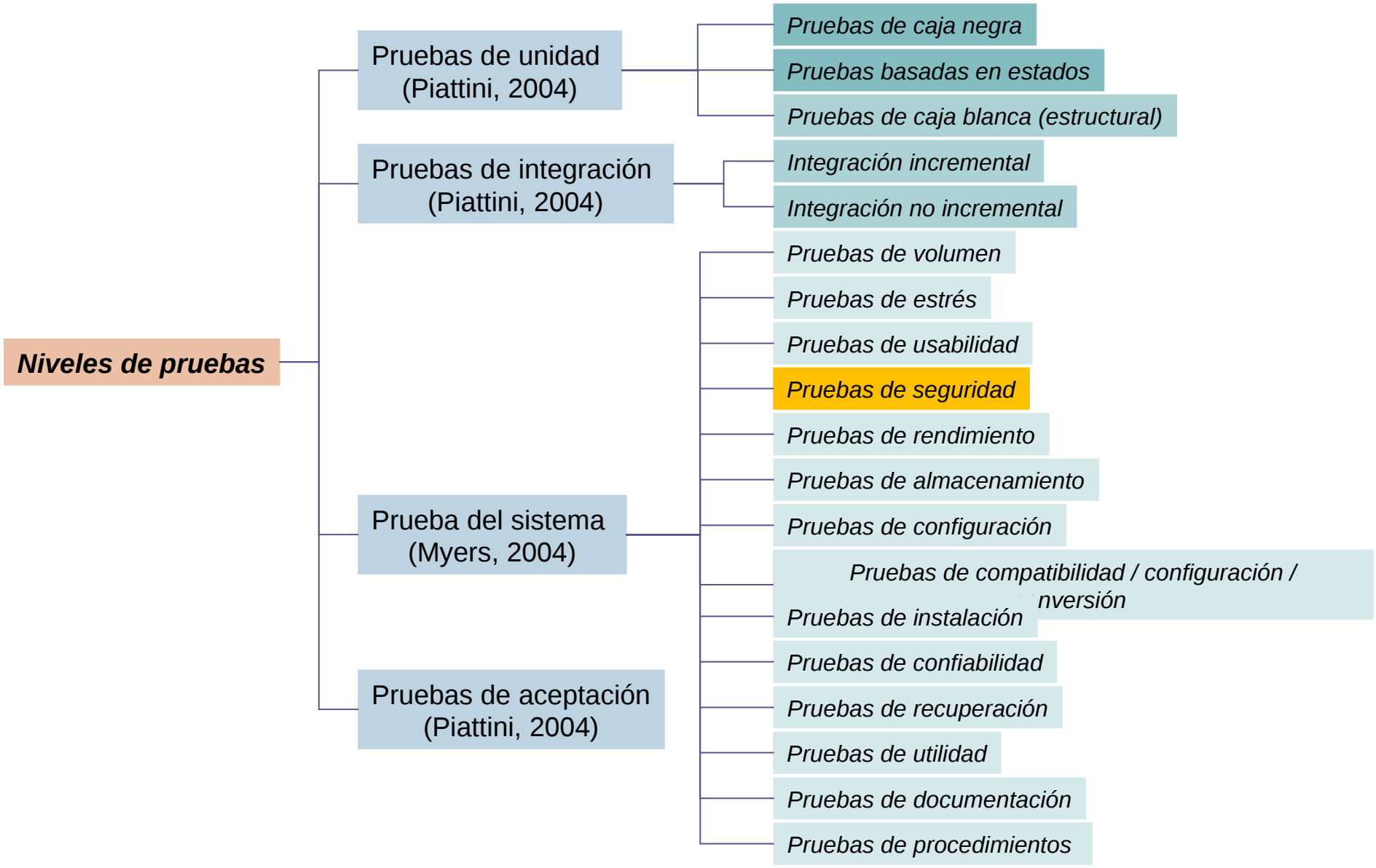
  - Una variación de una característica deseada del producto [Perry, 2000].

- *Falla:* La incapacidad de un sistema o de alguno de sus componentes para realizar las funciones requeridas dentro de los requisitos de rendimiento especificados [Piattini, 2004]

PRUEBAS DE SOFTWARE



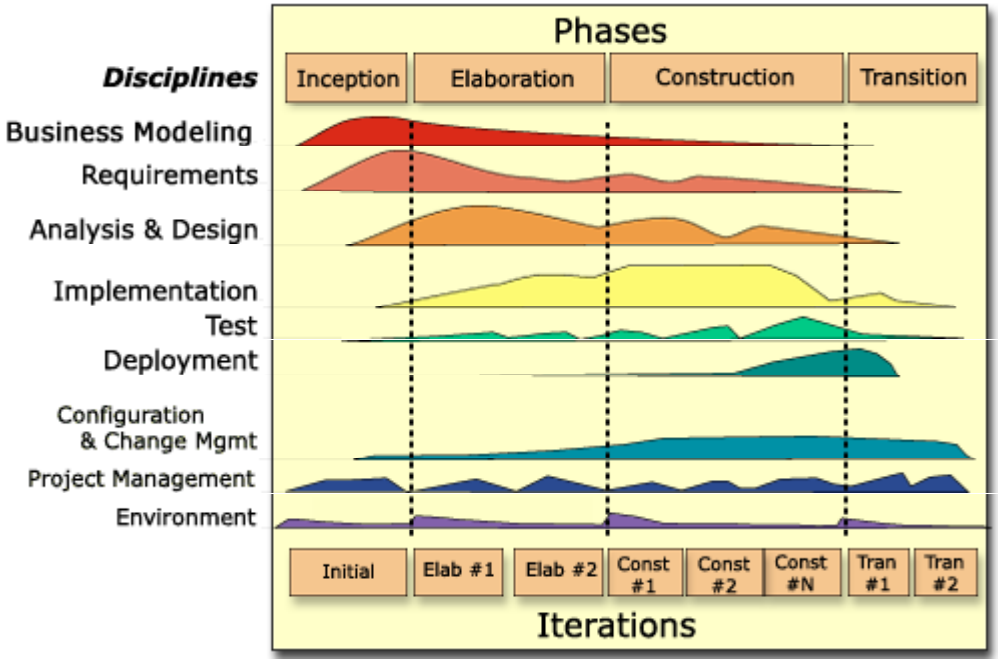
PRUEBAS DE SOFTWARE



PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

- RUP (Rational Unified Process) es un proceso de ingeniería de software. [IBM, 2003]
- Proporciona un acercamiento disciplinado a la asignación de tareas y responsabilidades dentro de un desarrollo de la organización. [IBM, 2003]
- Su meta es asegurar la producción de software de alta calidad que resuelva las necesidades de los usuarios finales dentro de un cronograma y un presupuesto fiables. [IBM, 2003]

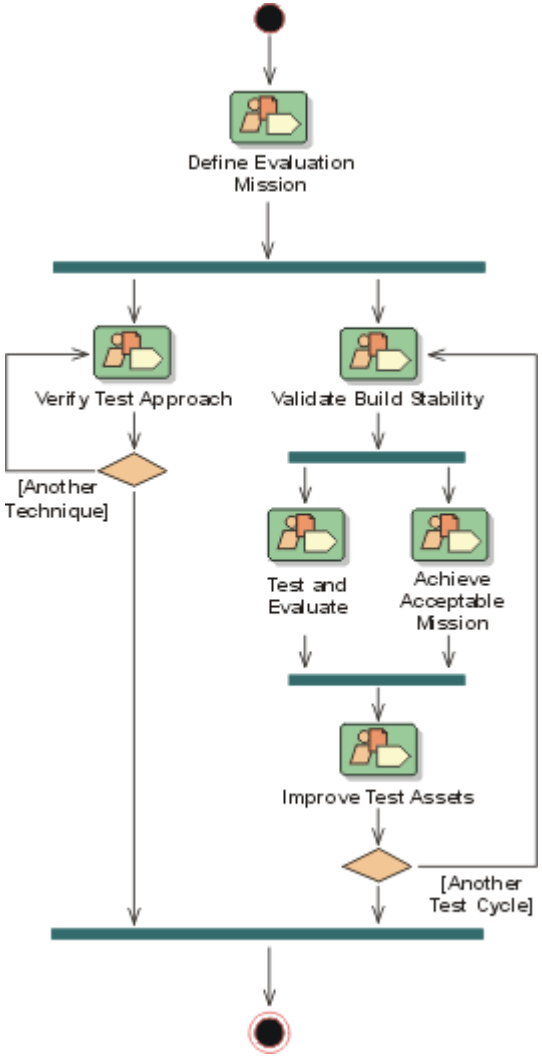


Fases de RUP

Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP



Proceso de pruebas de RUP

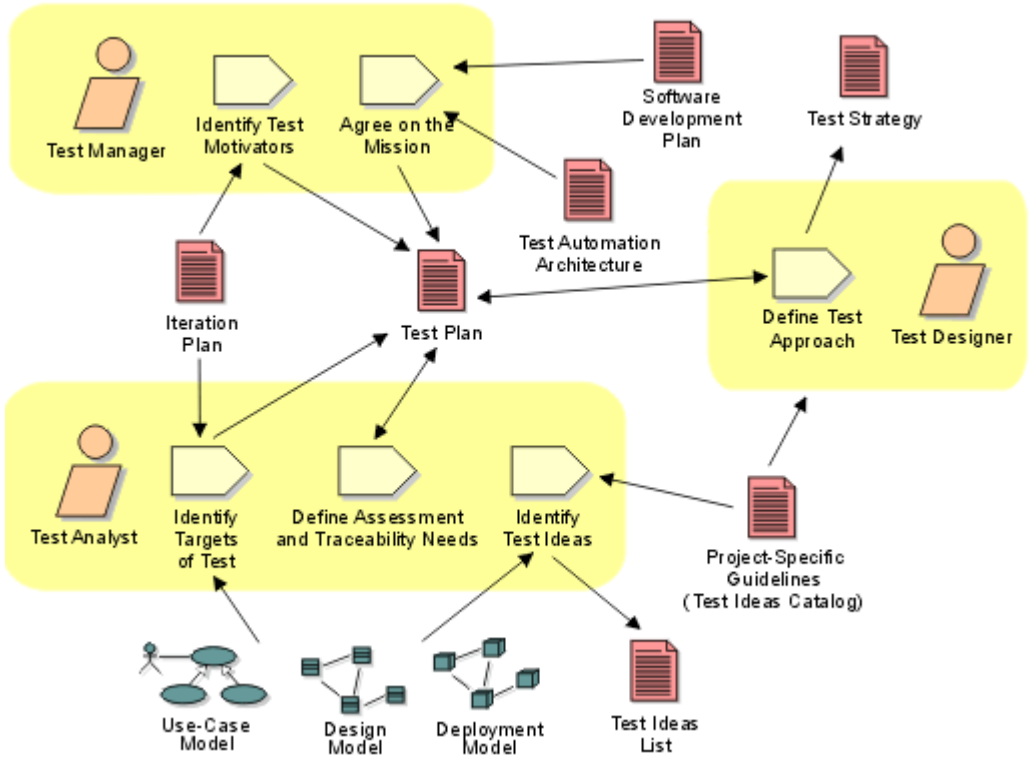
Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

Definición de la misión de evaluación

El propósito de esta fase es el de determinar el enfoque apropiado para cada iteración de pruebas y acordar los objetivos con los clientes los cuales serán parte del proceso de pruebas.



Definición de la misión de evaluación

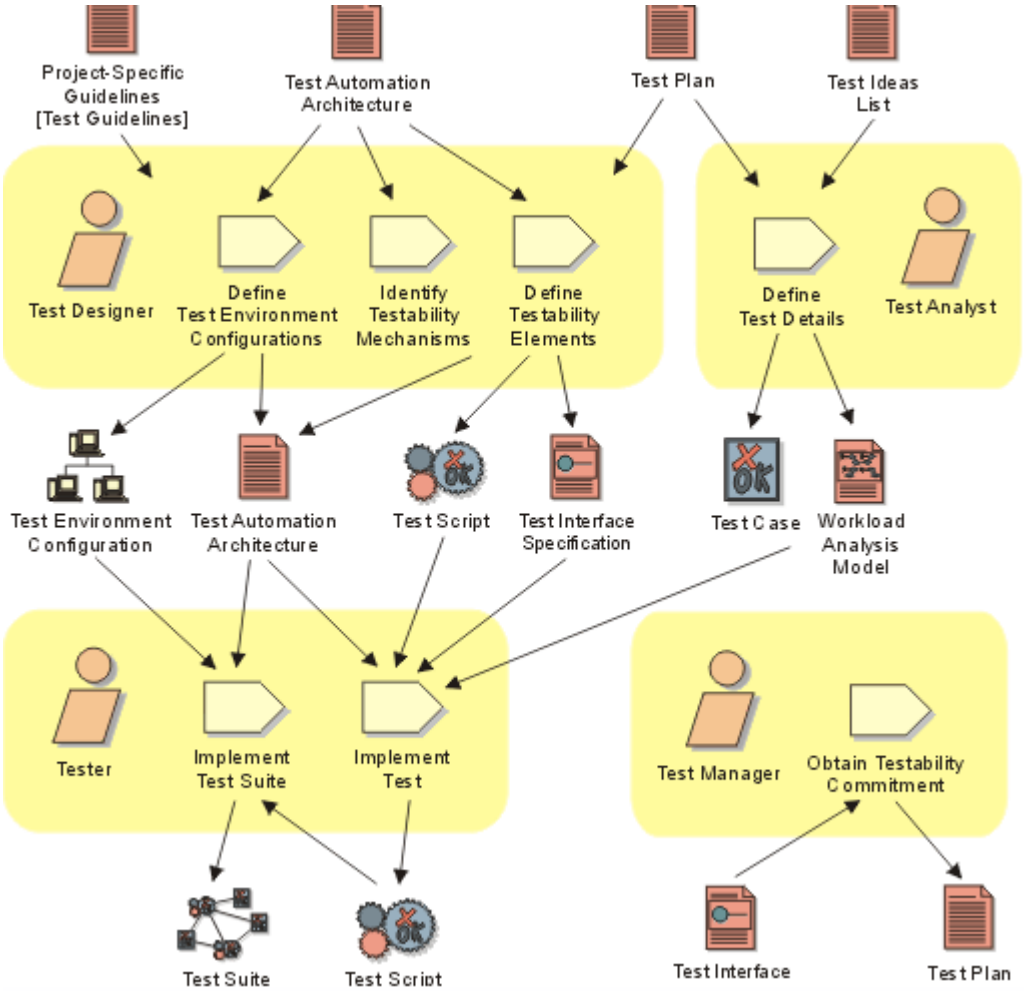
Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

Verificación del enfoque de la prueba

El objetivo de esta fase es el de demostrar que varias técnicas de pruebas, escogidas en la definición del plan de pruebas, pueden facilitar el desarrollo del esfuerzo planeado de pruebas. La idea es demostrar, por medio de la verificación, que el acercamiento hecho efectivamente funciona, se producen resultados exactos y es apropiado con los recursos disponibles.



Verificación del enfoque de la prueba

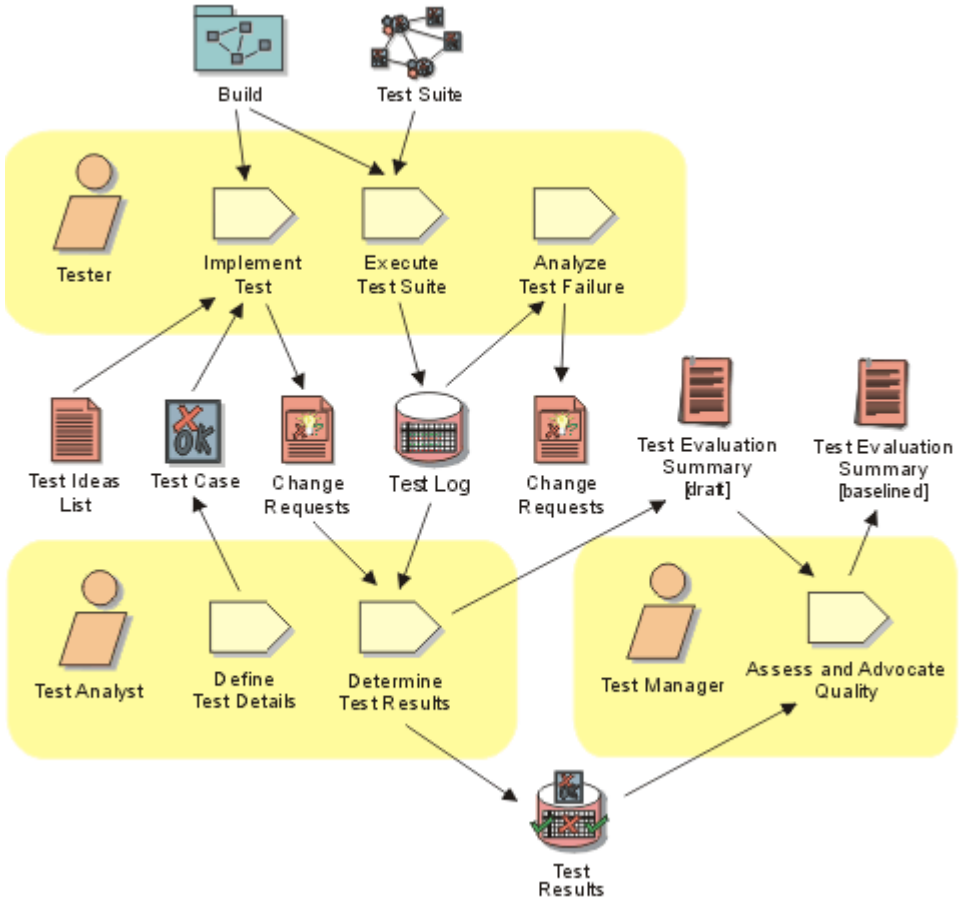
Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

Validación de la estabilidad de la implementación

La idea de esta fase es validar que la implementación hecha es lo suficientemente estable para comenzar pruebas más detalladas y hacer evaluaciones de esfuerzo. Esta fase ayuda a prevenir que los recursos de pruebas se malgasten en esfuerzos infructíferos de pruebas.



Validación de la estabilidad de la implementación

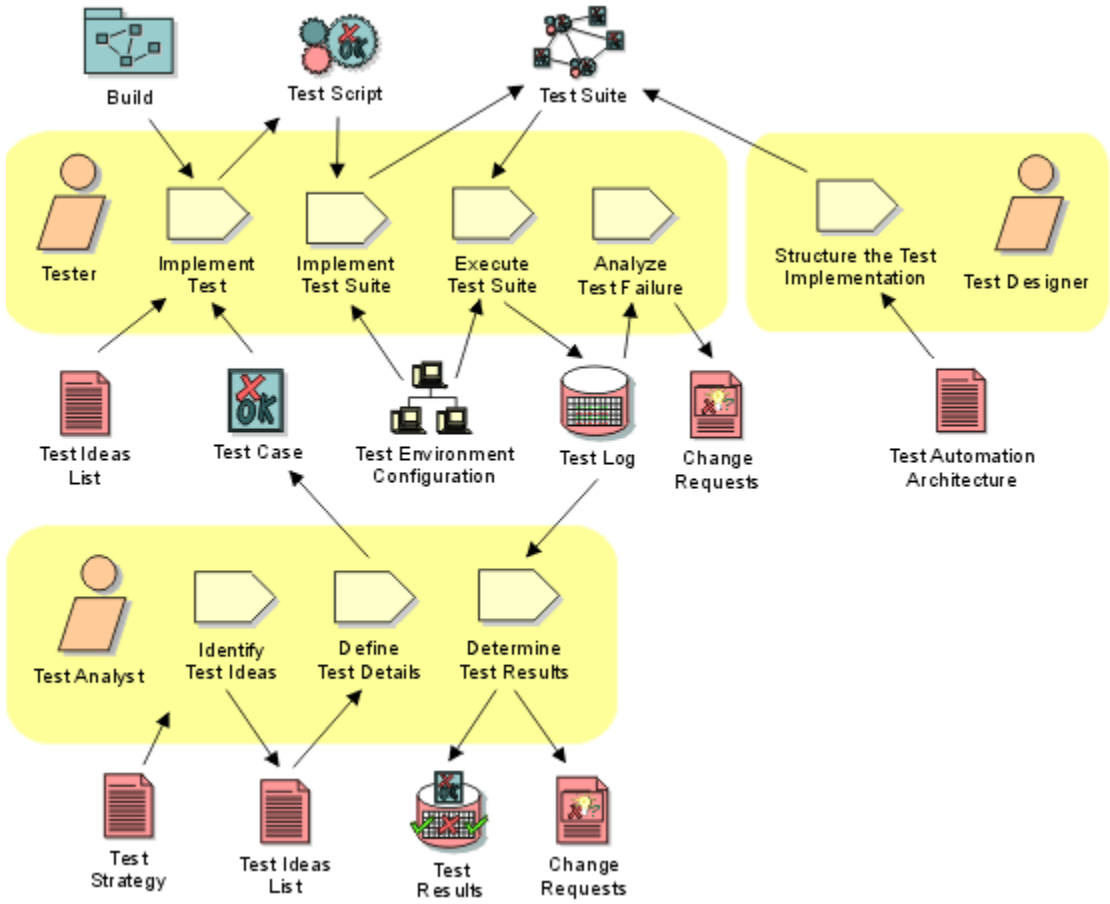
Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

Ejecución de pruebas y evaluación

El objetivo de esta fase es el de asegurar que el desarrollo tanto vertical como horizontal definidos dentro del plan de pruebas son suficientes para realizar una evaluación adecuada de los objetivos propuestos.



Ejecución de pruebas y evaluación

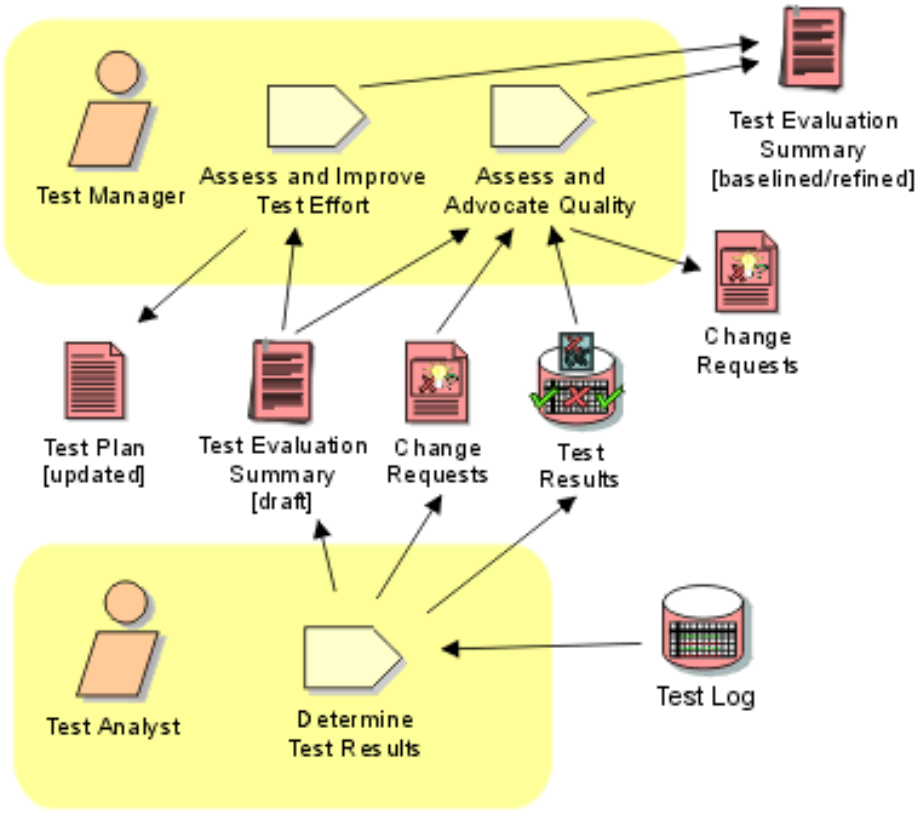
Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

Alcanzar resultados aceptables

El objetivo de esta fase es el de entregar un resultado útil de la evaluación a los clientes relacionados con el esfuerzo de pruebas donde una evaluación aceptable de los resultados es determinada por el aseguramiento del la misión de evaluación. En la mayoría de los casos eso significará que hay que centrar los esfuerzos en ayudar al equipo del proyecto a alcanzar los objetivos de la iteración que se aplican al ciclo actual de la prueba.



Alcanzar resultados aceptables

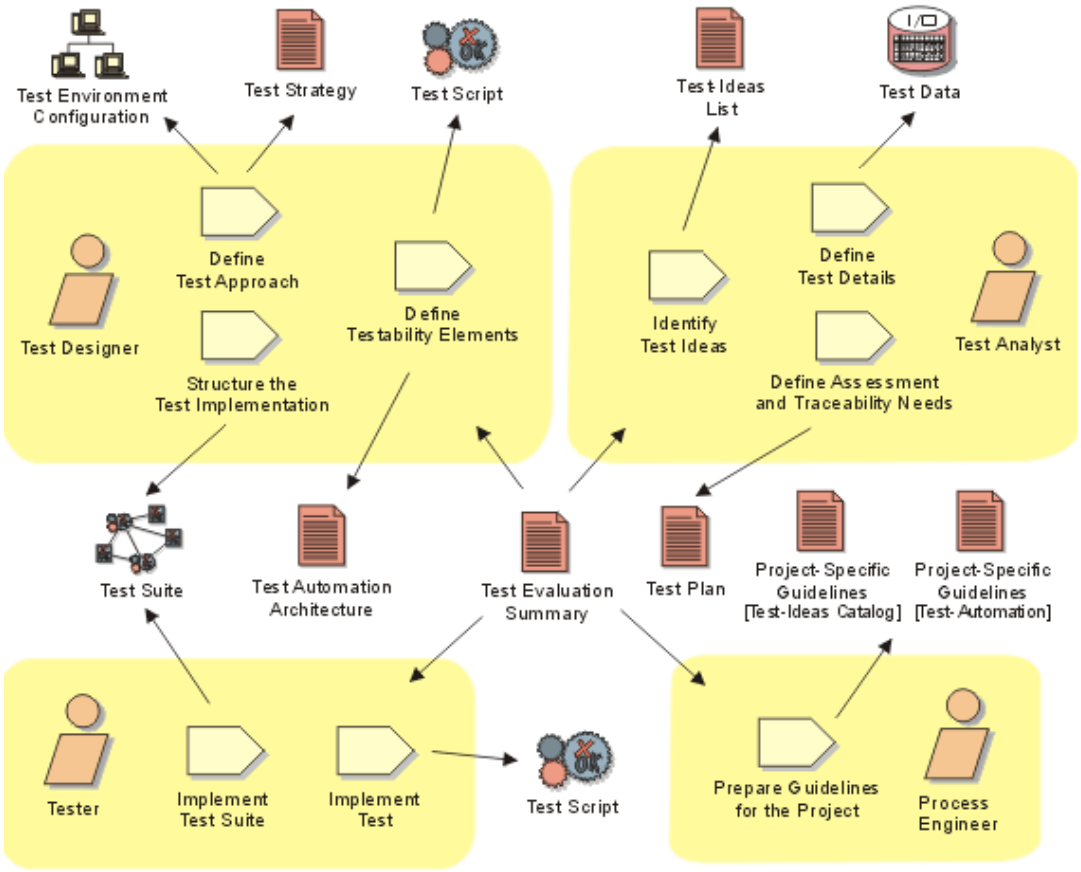
Tomado de [IBM, 2003]

PRUEBAS DE SOFTWARE

PROCESO DE PRUEBAS EN RUP

Mejoramiento de las pruebas

El propósito de esta fase es el de mantener y mejorar las pruebas. Esto es importante especialmente si la intención es reutilizar los activos desarrollados en el ciclo actual de pruebas en ciclos subsecuentes.



Mejoramiento de las pruebas

Tomado de [IBM, 2003]

ASPECTOS RELEVANTES DEL DISEÑO DE SEGURIDAD Y PATRONES DE SEGURIDAD

Cuando se desea generar pruebas a la capa de seguridad el mejor punto de partida es el estudio de la documentación existente sobre formas de implementar seguridad, definiciones, estándares, tipos de ataques y contramedidas existentes. Todo esto vinculado con el desarrollo de software y software seguro.

ATAQUES, SERVICIOS Y MECANISMOS DE SEGURIDAD

Entre los conceptos a tener en cuenta para el desarrollo del proyecto se deben considerar algunos aspectos de seguridad que permiten determinar las necesidades de seguridad que pueden ser implementadas en un sistema de información como el SI-SAAB. Entre estos se consideran los siguientes:

ASPECTOS RELEVANTES DEL DISEÑO DE SEGURIDAD Y PATRONES DE SEGURIDAD

ATAQUES [STALLINGS, 2003]

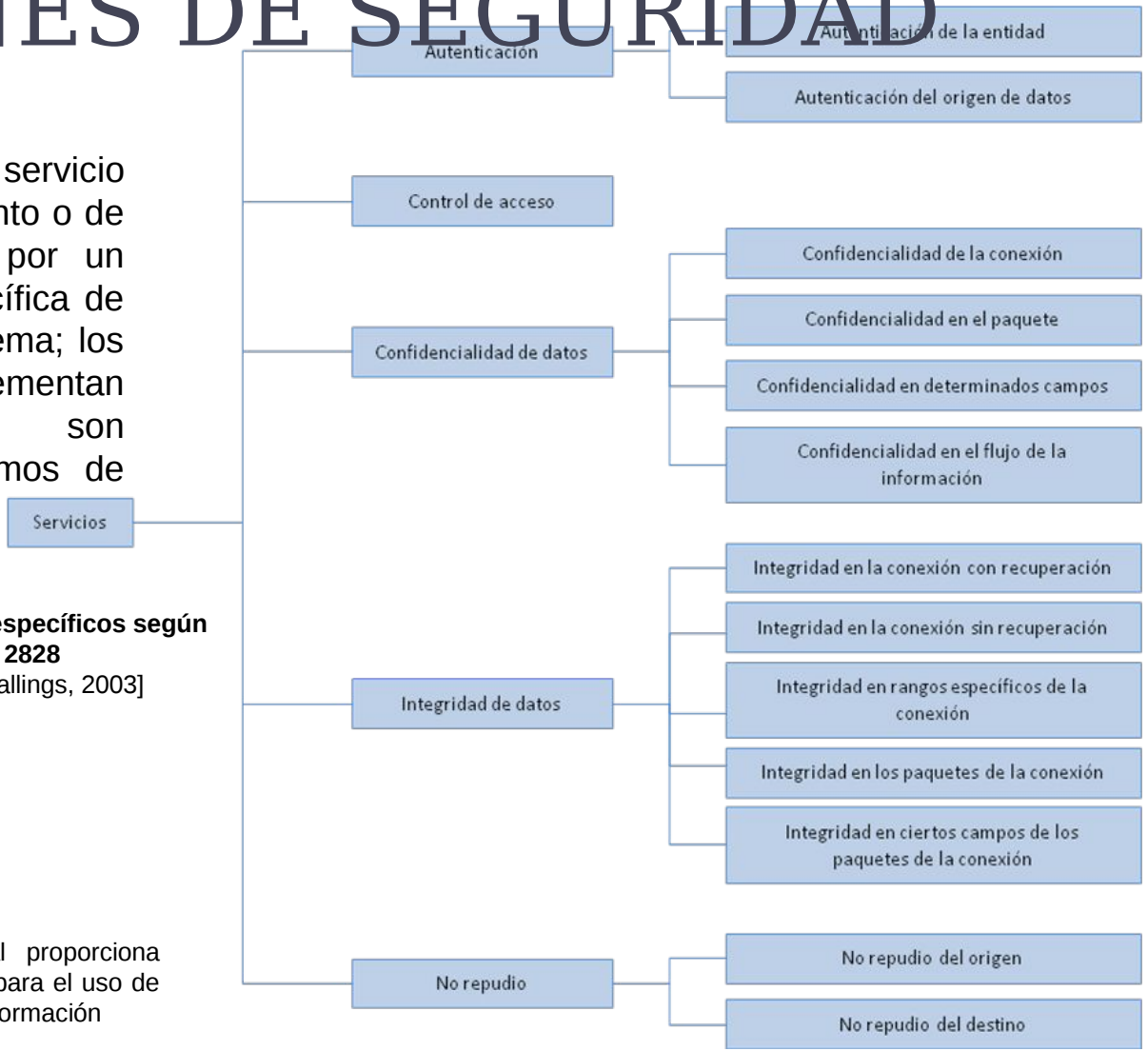
Se define un ataque como un evento exitoso o no, que atente contra el buen funcionamiento del sistema

Lista de ataques de seguridad
Tomado de [Stallings, 2003]

ASPECTOS RELEVANTES DEL DISEÑO DE SEGURIDAD Y PATRONES DE SEGURIDAD

SERVICIOS [STALLINGS, 2003]

Según la RFC 2828[1] se define un servicio como: “un servicio de procesamiento o de comunicación que es brindado por un sistema para dar una clase específica de protección a los recursos del sistema; los servicios de seguridad implementan políticas de seguridad y son implementados por los mecanismos de seguridad”.



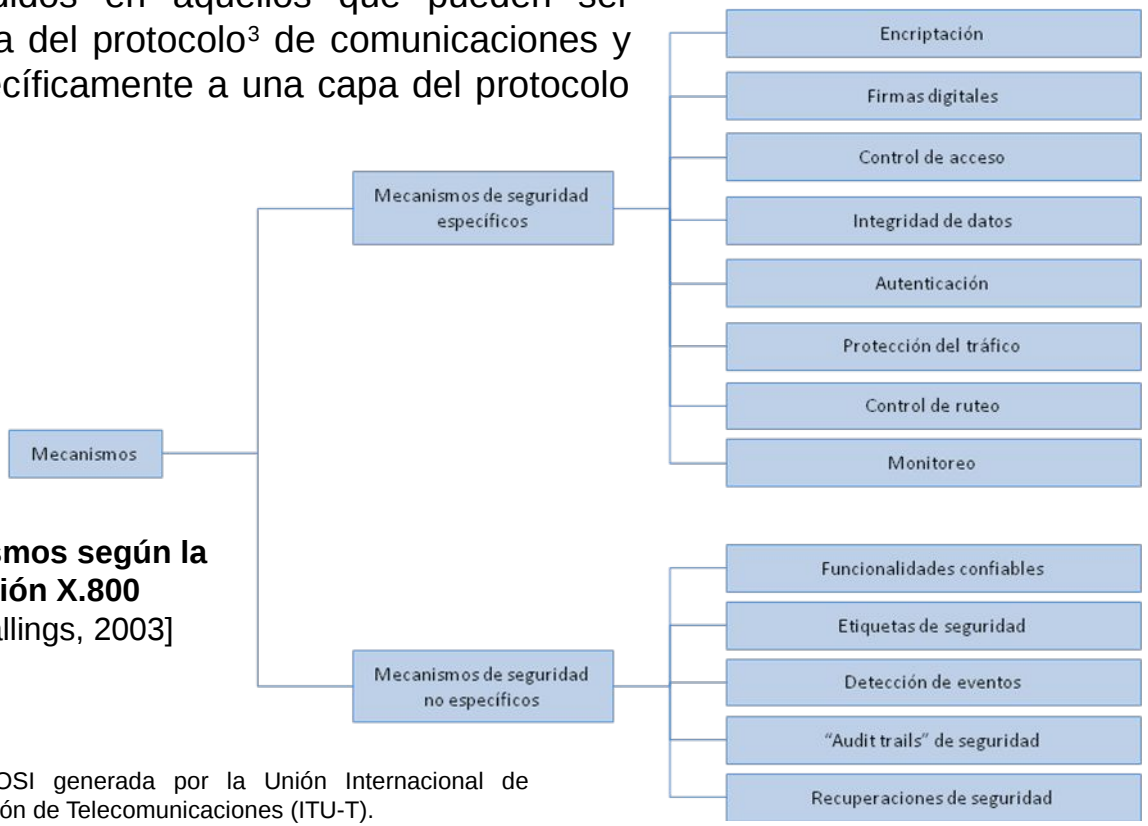
Lista de servicios específicos según la RFC 2828
Tomado de [Stallings, 2003]

1.Glosario de seguridad en Internet el cual proporciona abreviaturas, explicaciones y recomendaciones para el uso de la terminología de seguridad de un sistema de información

ASPECTOS RELEVANTES DEL DISEÑO DE SEGURIDAD Y PATRONES DE SEGURIDAD

MECANISMOS [STALLINGS, 2003]

La lista de mecanismos que se citan a continuación está definida en la recomendación X.800[2] y están divididos en aquellos que pueden ser implementados en una capa específica del protocolo[3] de comunicaciones y aquellos que no están asociados específicamente a una capa del protocolo de servicio de seguridad.



Lista de mecanismos según la recomendación X.800
Tomado de [Stallings, 2003]

2. Recomendación de la arquitectura segura para OSI generada por la Unión Internacional de Telecomunicaciones (ITU) y el Sector de Estandarización de Telecomunicaciones (ITU-T).
3. Protocolo de comunicaciones OSI.

ASPECTOS RELEVANTES DEL DISEÑO DE SEGURIDAD Y PATRONES DE SEGURIDAD

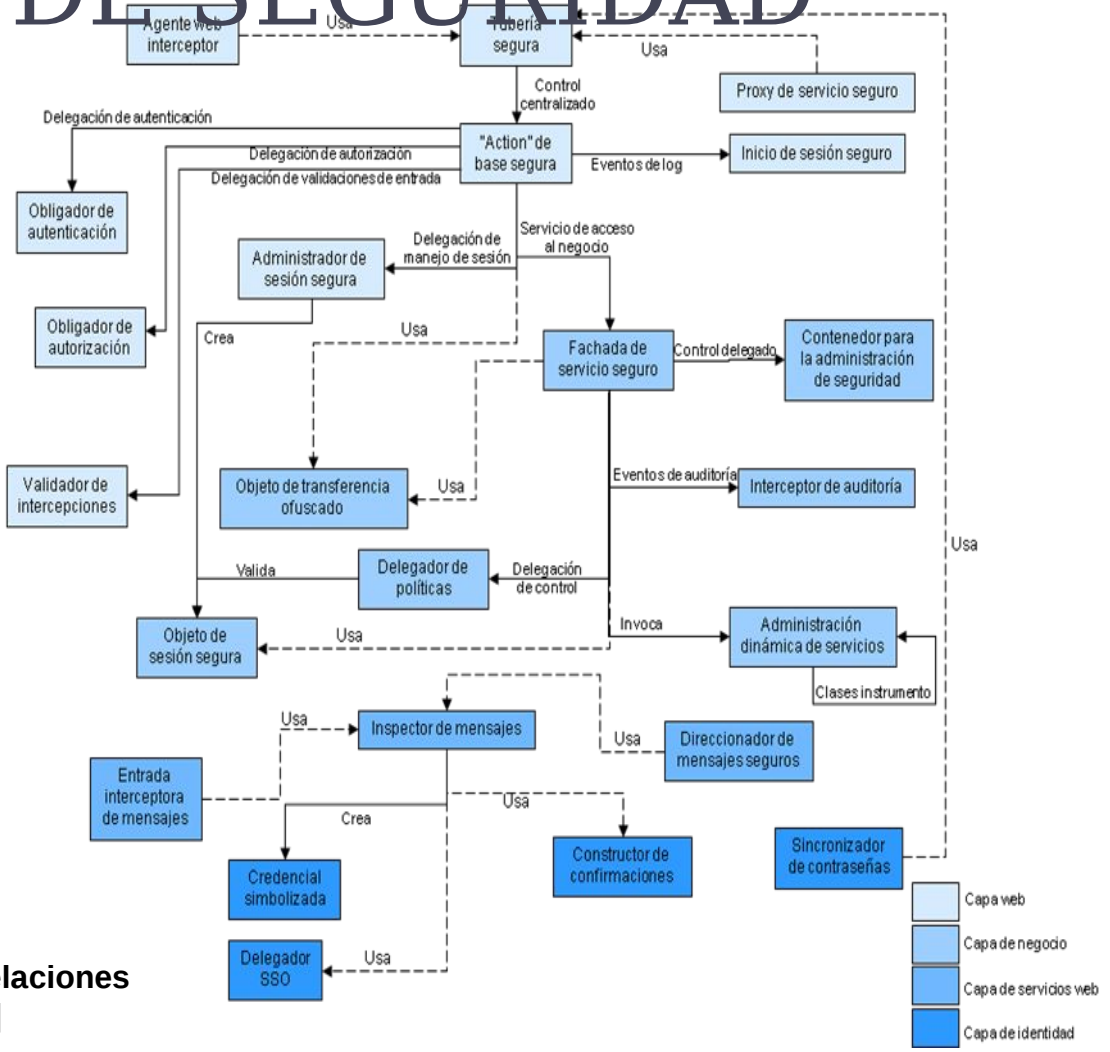
Servicio	Mecanismo							
	Encriptación	Firmas digitales	Control de acceso	Integridad de datos	Autenticación	Protección del tráfico	Control de enrutamiento	Monitoreo
Autenticación de la entidad	X	X			X			
Autenticación del origen de datos	X	X						
Control de acceso			X				X	
Confidencialidad	X						X	
Confidencialidad en el flujo de la información	X					X		
Integridad de datos	X	X		X				
No repudio		X		X				X

Relación entre servicios de seguridad y mecanismos.
Tomado de [Stallings, 2003]

ASPECTOS RELEVANTES DEL DISEÑO DE SEGURIDAD Y PATRONES DE SEGURIDAD

PATRONES DE SEGURIDAD

Los patrones de seguridad describen problemas particulares y recurrentes de seguridad que se dan en contextos específicos, y presentan esquemas genéricos bien definidos para su solución.



Patrones de seguridad y sus relaciones
Tomado de [Steel, 2006]

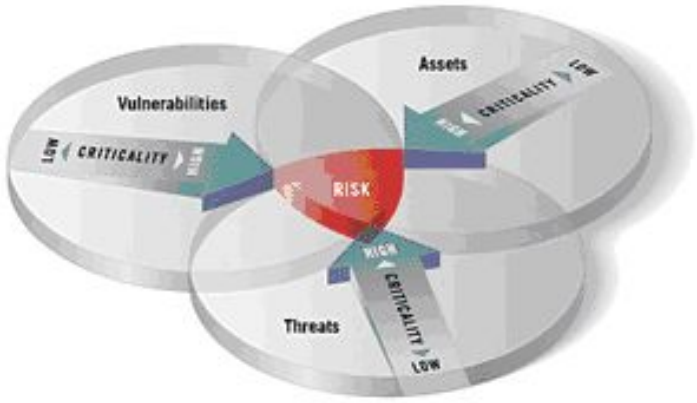
PRUEBAS DE SEGURIDAD DE SOFTWARE
[Wysopal, et al., 2006]

- *Vulnerabilidad:* según la RFC 2828 se define una vulnerabilidad como un defecto o debilidad en el diseño de un sistema, implementación u operación y administración que puede ser explotado o que viola una política de seguridad del sistema. Es la posibilidad de ocurrencia de una amenaza.

- *Amenaza:* es un evento que puede desencadenar un incidente en el sistema, produciendo daños materiales o pérdidas inmateriales en sus activos.

- *Riesgo:* posibilidad de que se produzca un impacto determinado en un activo, en un dominio o en todo el sistema.

- *Activo*: recurso del sistema de información o relacionado con éste, necesario para que el sistema funcione correctamente y alcance los objetivos propuestos. Entre los activos del sistema se tiene la información, el hardware y software y los usuarios del sistema.

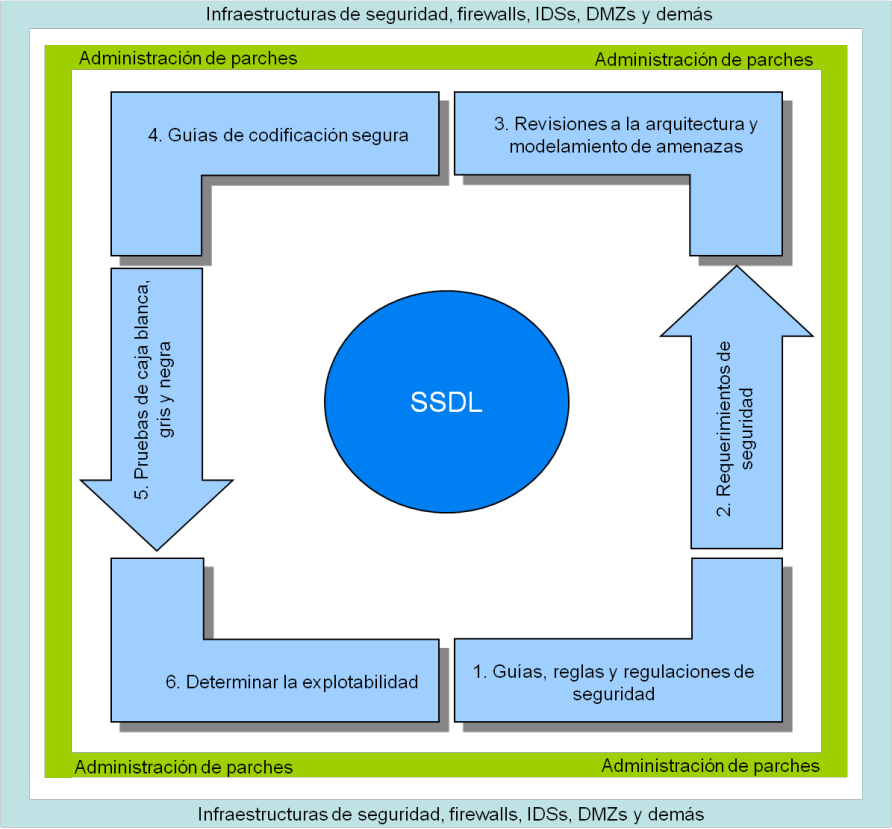


PRUEBAS DE SEGURIDAD DE SOFTWARE

[Wysopal, et al., 2006]

TÉCNICAS DE PRUEBAS DE SEGURIDAD DE SOFTWARE

El ciclo de desarrollo seguro



Relaciones entre SSDL y el ciclo de desarrollo de sistemas

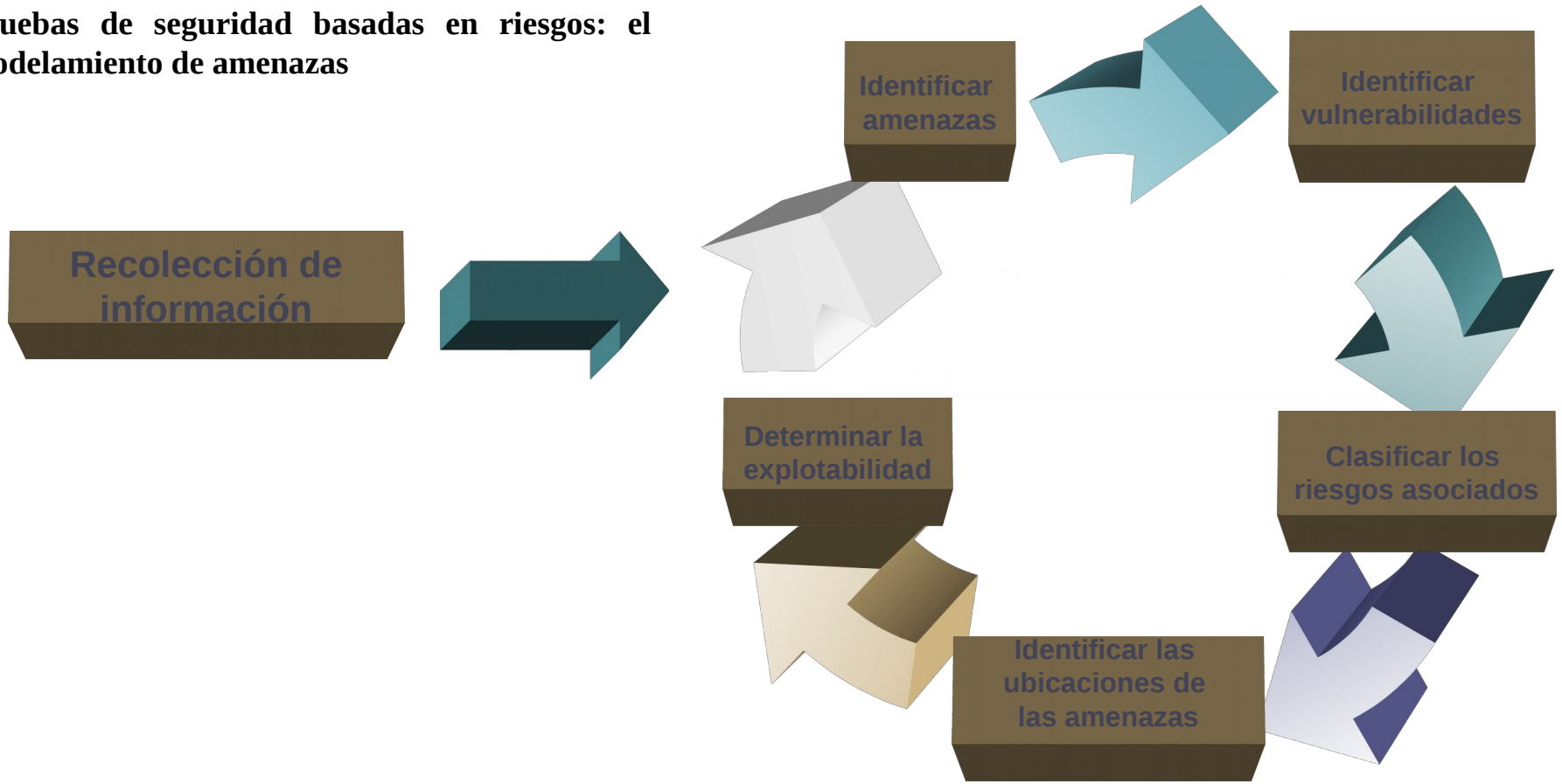
Tomado de [Wysopal, Lucas, Dino, & Elfriede, 2006]

PRUEBAS DE SEGURIDAD DE SOFTWARE

[Wysopal, et al., 2006]

TÉCNICAS DE PRUEBAS DE SEGURIDAD DE SOFTWARE

Pruebas de seguridad basadas en riesgos: el modelamiento de amenazas



DISEÑO DEL PROCESO DE PRUEBAS DE SEGURIDAD

MODELOS UTILIZADOS EN EL DISEÑO DEL PROCESO DE PRUEBAS DE SEGURIDAD

MODELO DE SEGURIDAD BASADO EN PATRONES

Brinda a un ingeniero que no es experto en seguridad herramientas para aplicar seguridad en un desarrollo de software, las cuales en igual medida sirven para tener una idea preliminar al generar un catálogo de ideas de pruebas de seguridad.

CICLO DE DESARROLLO DE SOFTWARE SEGURO (SSDL)

SSDL es tenido en cuenta para el diseño del proceso de pruebas de seguridad debido a que establece ciertos procesos dentro del ciclo de desarrollo de software que permiten al equipo de pruebas identificar distintos aspectos de seguridad tenidos en cuenta durante la fase de análisis y diseño del software que permitirán generar un mejor diseño de las pruebas de seguridad del software.

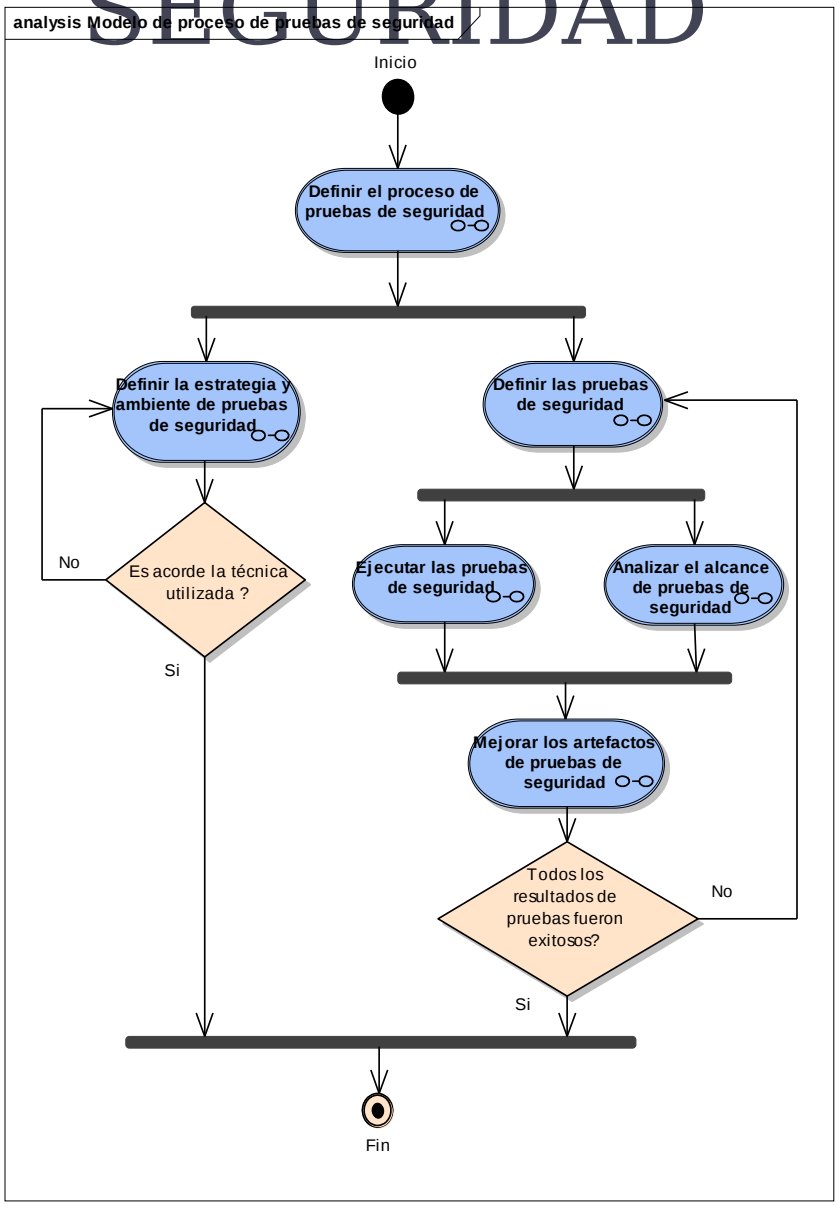
EL MODELAMIENTO DE AMENAZAS

Esta es una metodología que se acomoda a las necesidades del proyecto ya que plantea un proceso documentado para la identificación de una superficie de ataque del sistema y las diferentes amenazas y vulnerabilidades que se puedan encontrar en esta, así mismo presenta una técnica (la técnica DREAD) para la priorización de estas vulnerabilidades y tener un criterio de selección al momento de ejecutar las pruebas de seguridad.

PROCESO DE PRUEBAS DE SOFTWARE EN RUP

Se utiliza RUP como punto de referencia debido a que en el SI-SAAB se utiliza RUP para el desarrollo del sistema de información y más específicamente en el equipo de pruebas. Lo cual garantiza una integración más uniforme al generar el proceso de pruebas de seguridad.

MODELO DE PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD

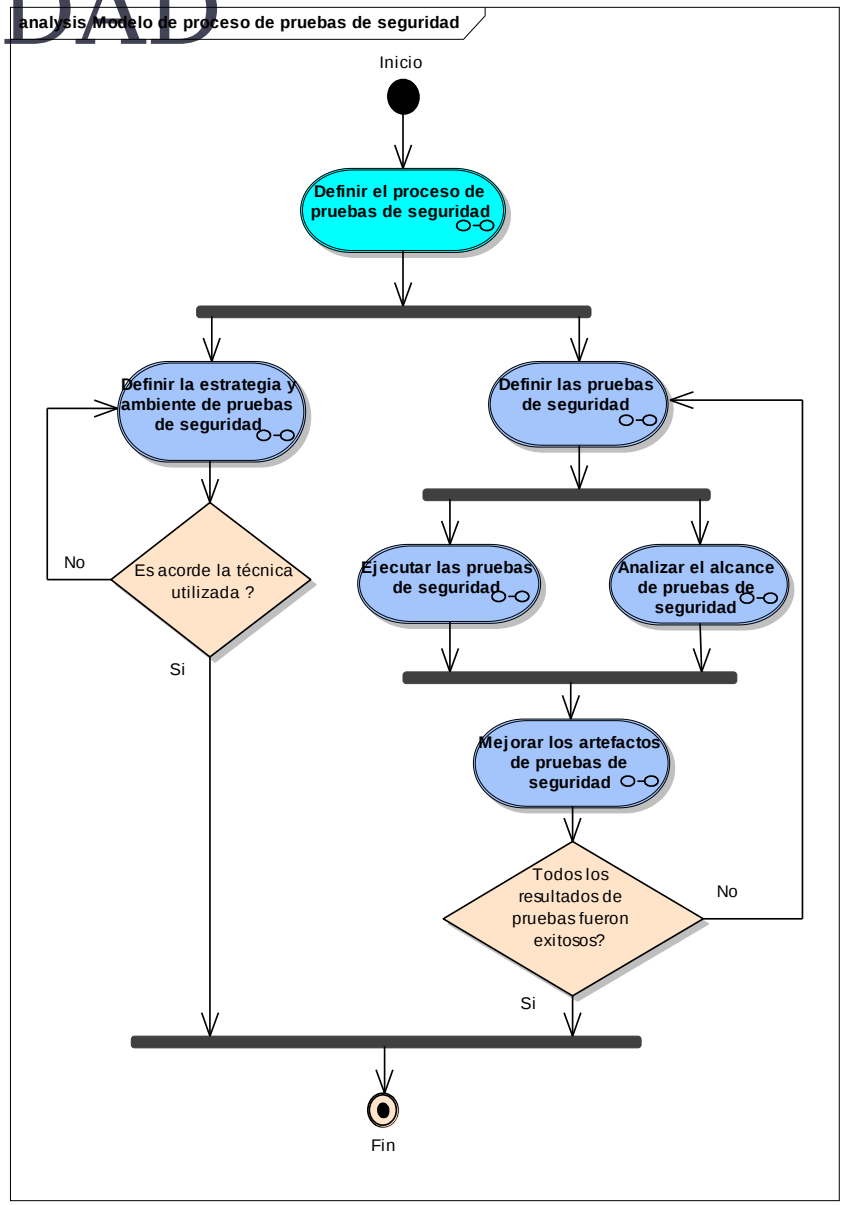
El objetivo de esta primera fase es analizar el desarrollo de software que se está llevando a cabo, los aspectos de seguridad que deben ser tenidos en cuenta para su desarrollo y la estrategia de pruebas a realizar durante el proceso, para establecer el enfoque a seguir durante la fase de pruebas de seguridad.

Artefactos de entrada

- Documentación del sistema (modelos funcionales, estructurales, dinámicos, arquitectura lógica y de implementación).
- Políticas de seguridad.
- Regulaciones legales sobre seguridad.
- Árboles de conocimiento sobre fallas de seguridad en software similares.
- Patrones de seguridad
- Plan maestro de pruebas definido para el desarrollo de software.

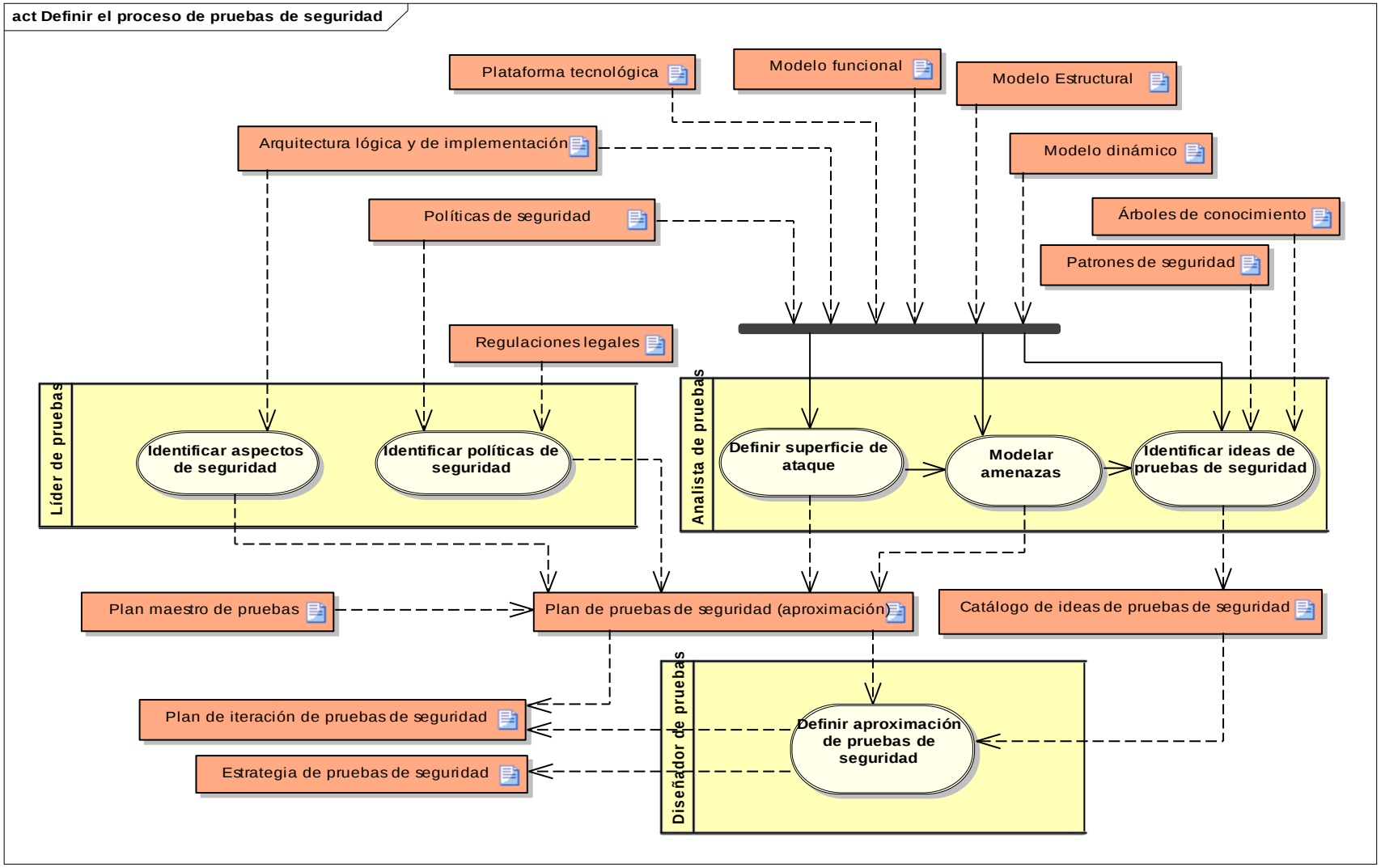
Artefactos generados

- Plan de pruebas de seguridad (aproximación)
- Catálogo de ideas de pruebas de seguridad
- Plan de iteración de pruebas de seguridad
- Estrategia de pruebas de seguridad



MODELO DE PRUEBAS DE SEGURIDAD

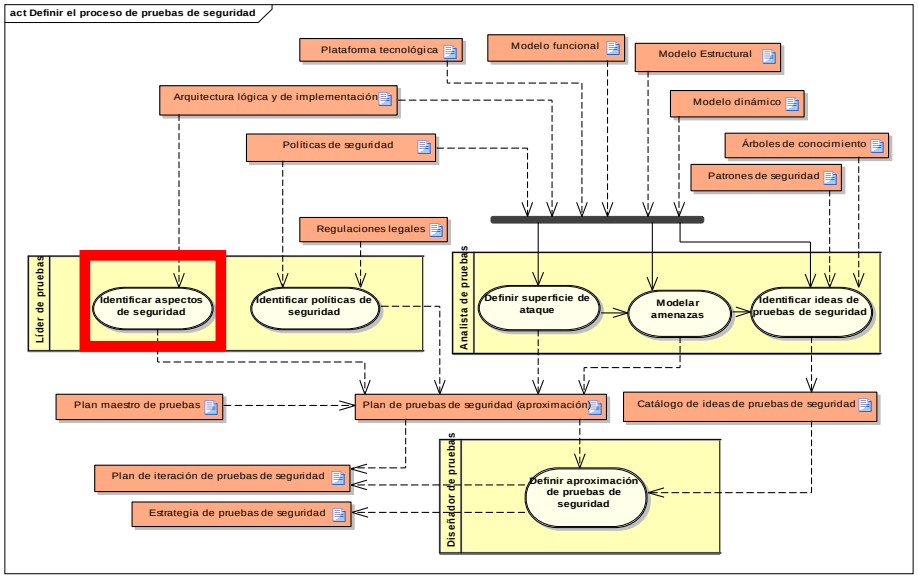
DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

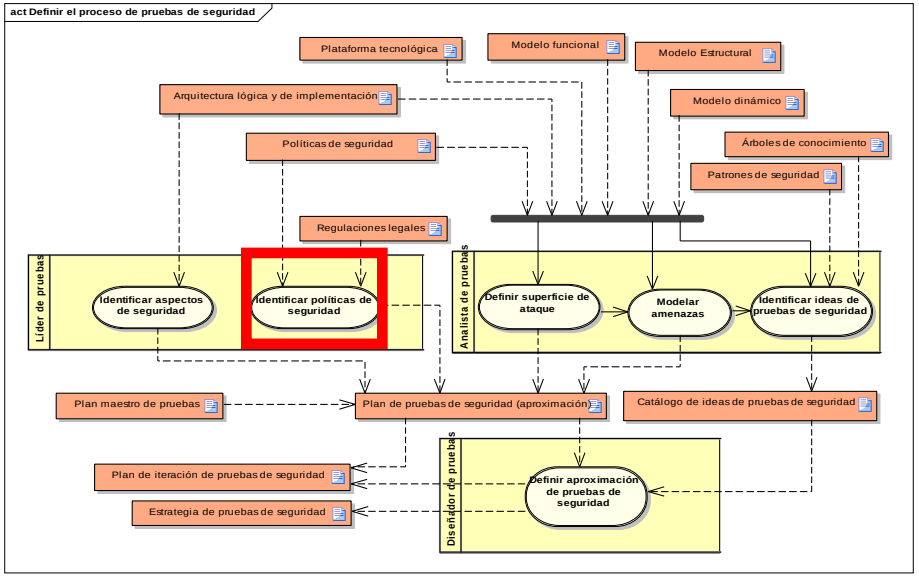
DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD

Identificar aspectos de seguridad	
Propósito Identificar los distintos puntos de seguridad necesarios para el proyecto y que necesitan ser probados.	
Rol Líder de pruebas	
Pasos •Identificar los requerimientos de seguridad asociados al software. •Analizar las técnicas de seguridad definidas para cumplir con los requerimientos de seguridad definidos. •Identificar aquellos requerimientos de seguridad que pueden ser probados de acuerdo al modelo de seguridad implementado.	
Artefactos de entrada •Arquitectura lógica y de implementación	*Artefactos resultantes* •Plan maestro de pruebas de seguridad



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD

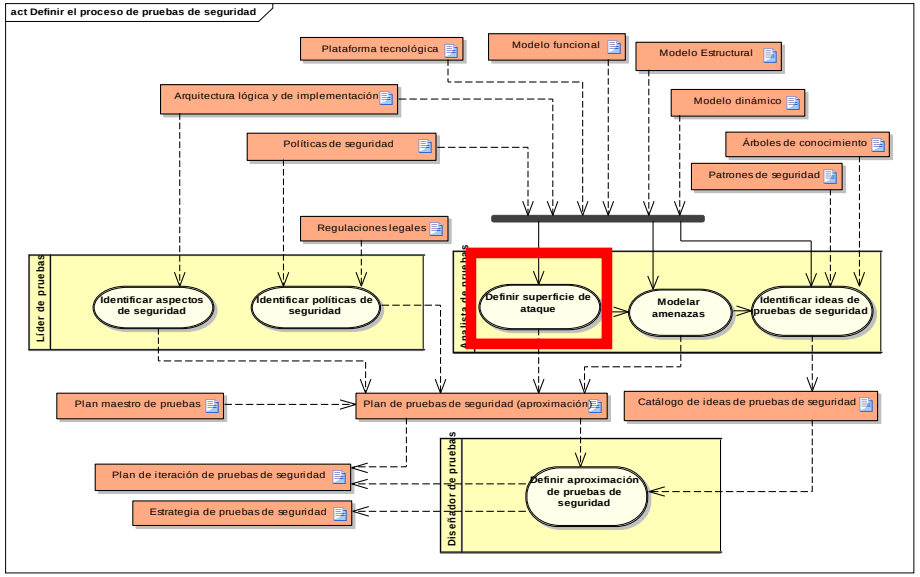


Identificar políticas de seguridad	
Propósito Identificar las distintas normativas y legislaciones tanto nacionales e internacionales referentes a la seguridad que afecten el desarrollo del software.	
Rol Líder de pruebas	
Pasos •Identificar que legislaciones o normas de seguridad pueden afectar el desarrollo del software. •De acuerdo a las normas identificadas realizar un análisis comparativo de cuales fueron tenidas en cuenta en el modelo de seguridad, cuales están en contra del modelo y cuáles no fueron tenidas en cuenta pero son necesarias para el desarrollo. •Identificar aquellas políticas que pueden ser probadas de acuerdo al modelo de seguridad implementado.	
Artefactos de entrada •Regulaciones y normas de seguridad •Políticas de seguridad	*Artefactos resultantes* •Plan maestro de pruebas de seguridad

MODELO DE PRUEBAS DE SEGURIDAD

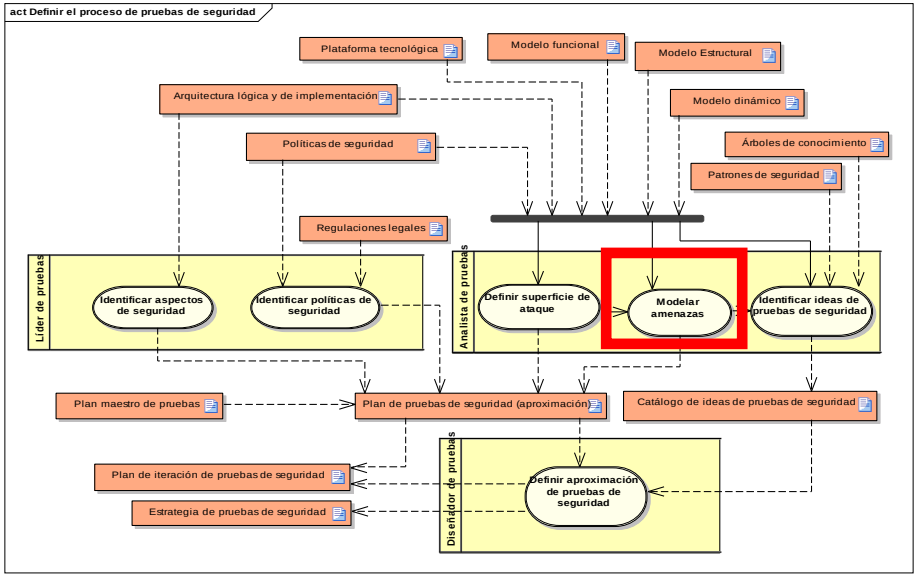
DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD

Definir la superficie de ataque	
Propósito Definir todo el posible campo de acción de un atacante frente al sistema	
Rol Analista de pruebas	
Pasos •Identificar las entradas y salidas del sistema. •Identificar los flujos de información en el sistema. •Definir cuáles de los anteriores poseen algún tipo de protección y cuáles no.	
Artefactos de entrada •Documentación del sistema (modelos funcionales, estructurales, dinámicos y de seguridad, diseño de la plataforma tecnológica)	*Artefactos resultantes* •Plan maestro de pruebas de seguridad



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD

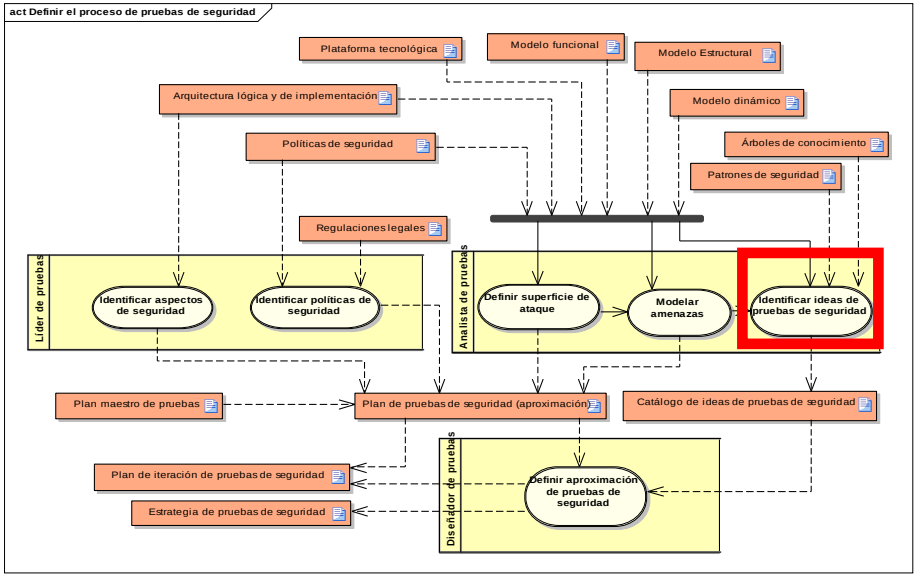


Modelar amenazas	
Propósito Su objetivo es plantear las posibles amenazas basado en el diseño e implementación de la aplicación así como establecer una priorización de las mismas.	
Rol Analista de pruebas	
Pasos •Identificar las ubicaciones de las posibles amenazas. •Identificar las amenazas individuales que se encuentran en cada ubicación. •De acuerdo a las amenazas encontradas se debe definir cualquier vulnerabilidad en alguna de esas ubicaciones. •Por medio del modelo DREAD clasificar la severidad de las amenazas. •Establecer una priorización de las amenazas encontradas.	
Artefactos de entrada •Documentación del sistema (modelos funcionales, estructurales, dinámicos y de seguridad, diseño de la plataforma tecnológica)	*Artefactos resultantes* •Plan maestro de pruebas de seguridad

MODELO DE PRUEBAS DE SEGURIDAD

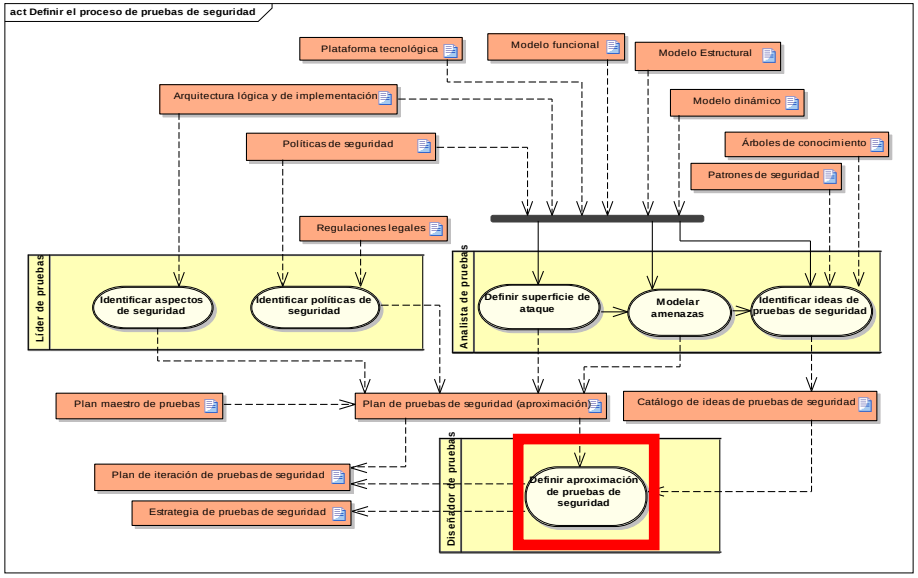
DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD

Identificar ideas de pruebas de seguridad	
Propósito Su objetivo es plantear las posibles pruebas que se pueden llegar a realizar al sistema.	
Rol Analista de pruebas	
Pasos •De acuerdo a la priorización de amenazas planteadas en la actividad anterior definir cuáles de estas pueden ser probadas. •De las amenazas seleccionadas anteriormente plantear conjuntos de pruebas que verifiquen la ausencia o presencia de dicha amenaza en el sistema. •Realizar un estudio entre los arboles de conocimiento y los patrones de seguridad para identificar ideas que no se hayan contemplado hasta el momento.	
Artefactos de entrada •Documentación del sistema (modelos funcionales, estructurales, dinámicos y de seguridad) •Arboles de conocimiento •Patrones de seguridad	*Artefactos resultantes* •Catálogo de ideas de prueba de ataque



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR EL PROCESO DE PRUEBAS DE SEGURIDAD



Definir aproximación de pruebas de seguridad	
Propósito Su objetivo es definir la estrategia a llevar a cabo durante el proceso de pruebas de seguridad.	
Rol Diseñador de pruebas	
Pasos •Definir los tipos de pruebas a llevar a cabo. •Establecer la configuración del ambiente de pruebas de seguridad. •Definir los puntos de corte entre iteración e iteración. •Definir los aspectos de seguridad a probar en cada iteración. •Asignar los recursos físicos, tecnológicos, de información y humanos a los diferentes subsistemas a probar. •Realizar una estimación de tiempo de la duración de cada iteración.	
Artefactos de entrada •Plan de pruebas de seguridad (aproximación) •Catálogo de ideas de pruebas de seguridad	*Artefactos resultantes* •Estrategia de pruebas de seguridad •Plan de iteración de pruebas de seguridad

MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD

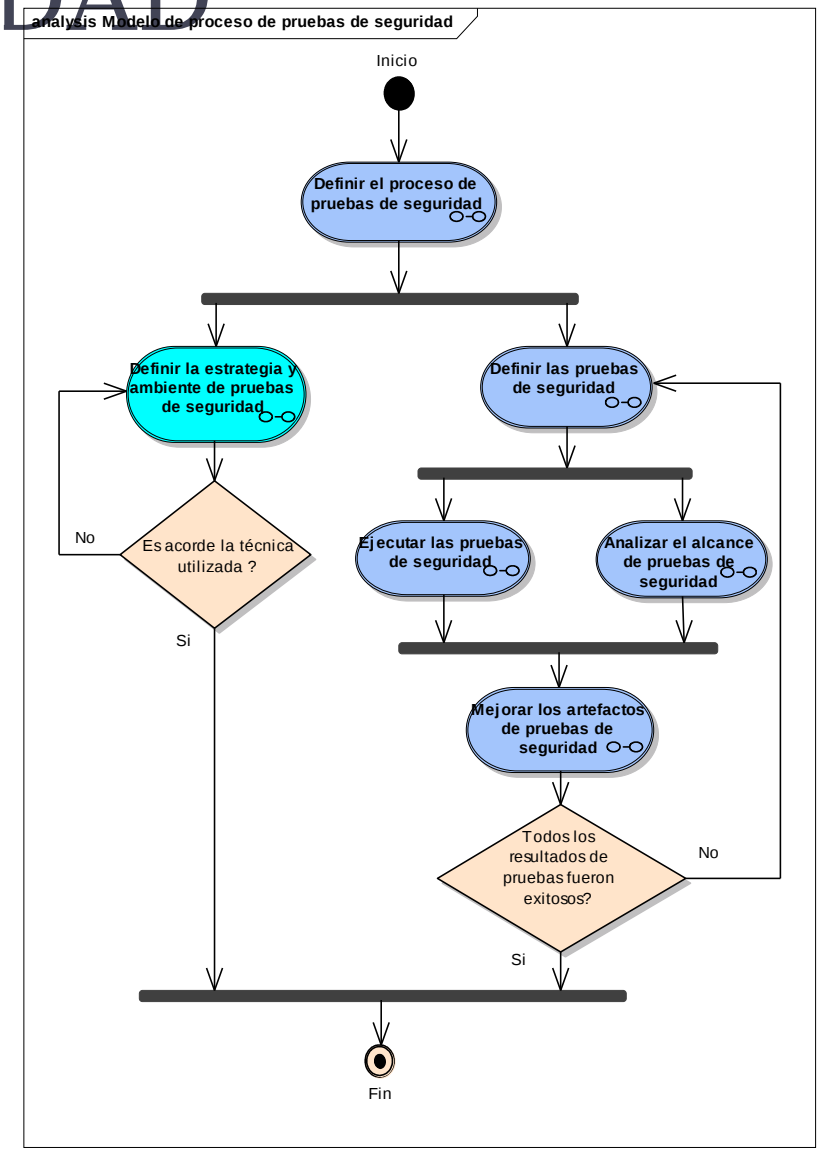
El objetivo de este flujo es, basado en la primera aproximación del plan de pruebas de seguridad, definir el ambiente de pruebas de seguridad, la estrategia de pruebas de seguridad a tener en cuenta y generar un plan de pruebas de seguridad más completo. En este punto se genera un ciclo de verificación para asegurar que las técnicas seleccionadas para llevar a cabo el proceso de pruebas de seguridad son acordes con las necesidades que se tienen dentro del desarrollo del sistema en cuanto a las verificaciones de seguridad.

Artefactos de entrada

- Plan de pruebas de seguridad (aproximación)

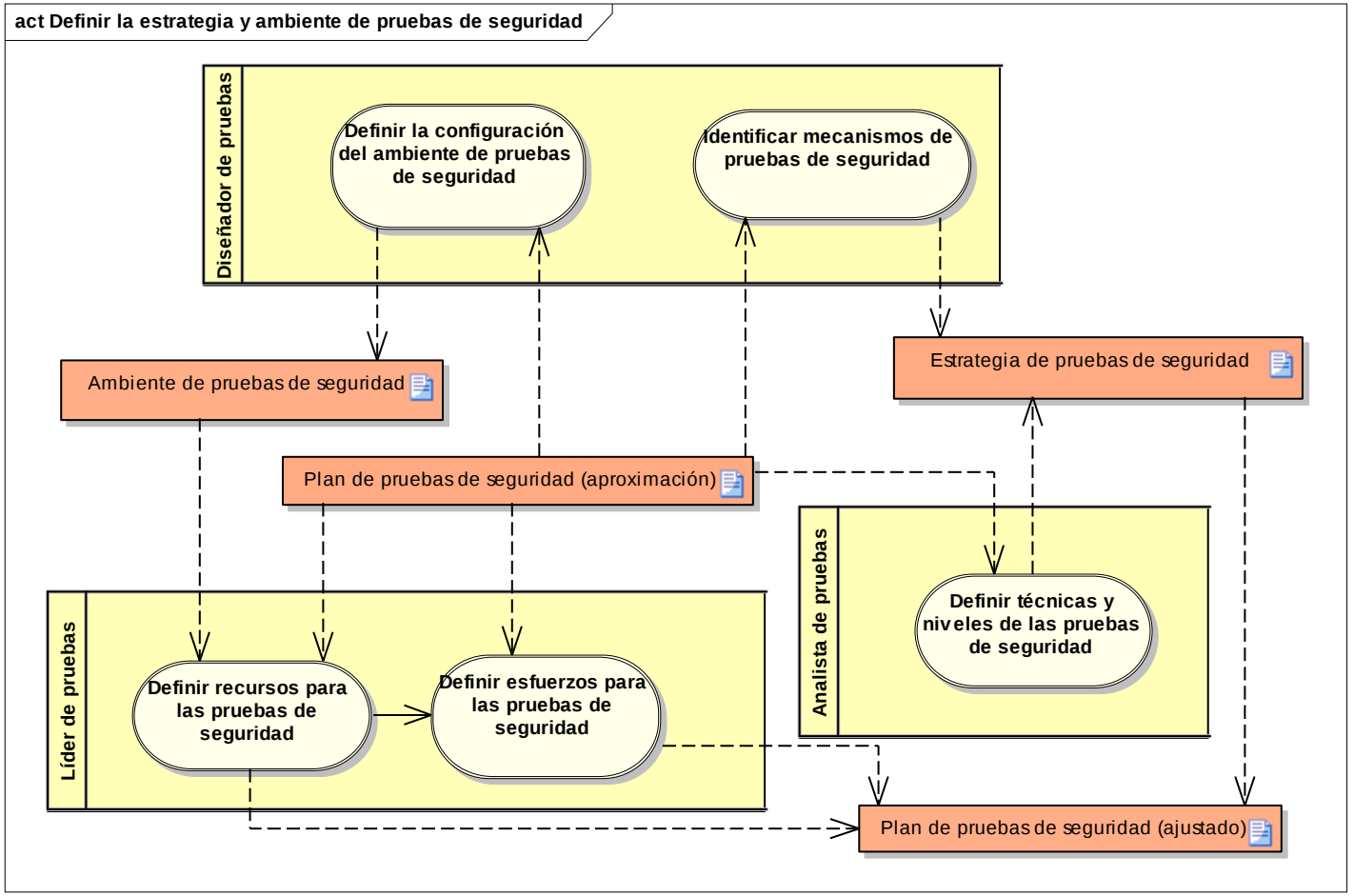
Artefactos generados

- Ambiente de pruebas de seguridad
- Estrategia de pruebas de seguridad
- Plan de pruebas de seguridad (revisado)



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD

Definir la configuración del ambiente de pruebas de seguridad

Propósito

Su objetivo es definir las necesidades de hardware y software necesarias para tener un ambiente controlado para la ejecución de las pruebas de seguridad.

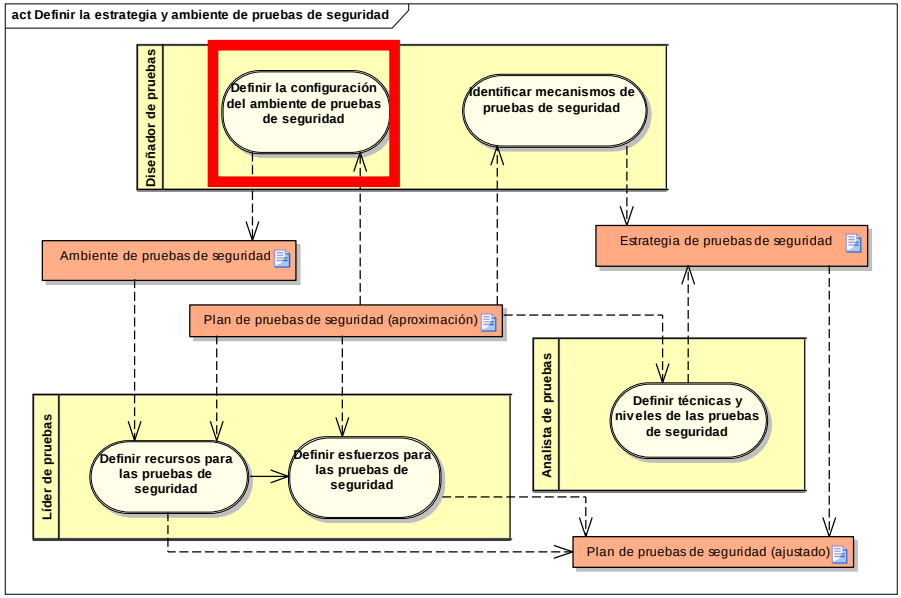
Rol

Diseñador de pruebas

Pasos

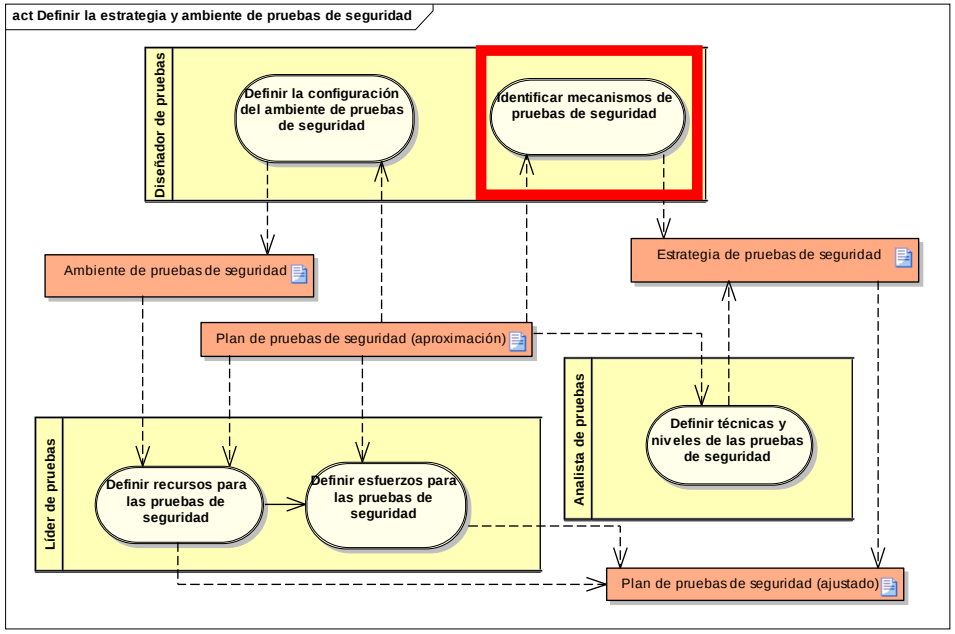
- De acuerdo a la aproximación del plan de pruebas generado en la fase anterior analizar las necesidades de hardware y software que serán utilizados en la ejecución de las pruebas de seguridad.
- Realizar un estudio comparativo sobre los distintos recursos de hardware y software existentes que suplen las necesidades del plan de pruebas de seguridad y escoger aquellos más acordes con las necesidades del proceso.
- Generar un documento donde explique detalladamente cada recurso seleccionado, su configuración dentro del ambiente de pruebas y su funcionamiento específico para el desarrollo de las pruebas de seguridad.

Artefactos de entrada	*Artefactos resultantes*
• Plan de pruebas de seguridad	• Documento de especificación del ambiente de pruebas de seguridad. • Implementación del ambiente de pruebas de seguridad



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD



Identificar mecanismos de pruebas de seguridad	
Propósito Su objetivo es definir la forma en que se pueden realizar los distintos tipos de pruebas de seguridad, especialmente aquellas ya definidas y asimiladas por el proyectos (como las obtenidas de los árboles de conocimiento)	
Rol Diseñador de pruebas	
Pasos •De acuerdo a la aproximación del plan de pruebas generado en la fase anterior analizar las ideas de pruebas de seguridad tenidas en cuenta. •Clasificar y agrupar esas ideas de pruebas de seguridad en grupos con características comunes. •Generar mecanismos que sirvan de apoyo para ejecutar cada uno de los distintos grupos de pruebas de seguridad establecidos en el paso anterior.	
Artefactos de entrada •Plan de pruebas de seguridad	*Artefactos resultantes* •Estrategia de pruebas de seguridad

MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD

Definir técnicas y niveles de pruebas de seguridad

Propósito

Su objetivo es definir las técnicas y niveles de pruebas a utilizar para la implementación y ejecución de las pruebas de seguridad

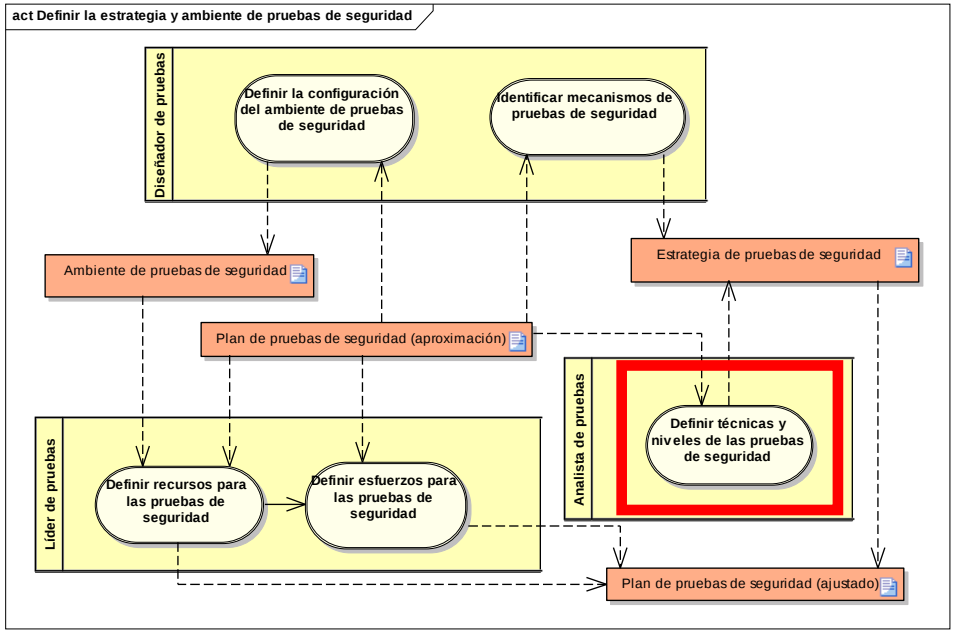
Rol

Analista de pruebas

Pasos

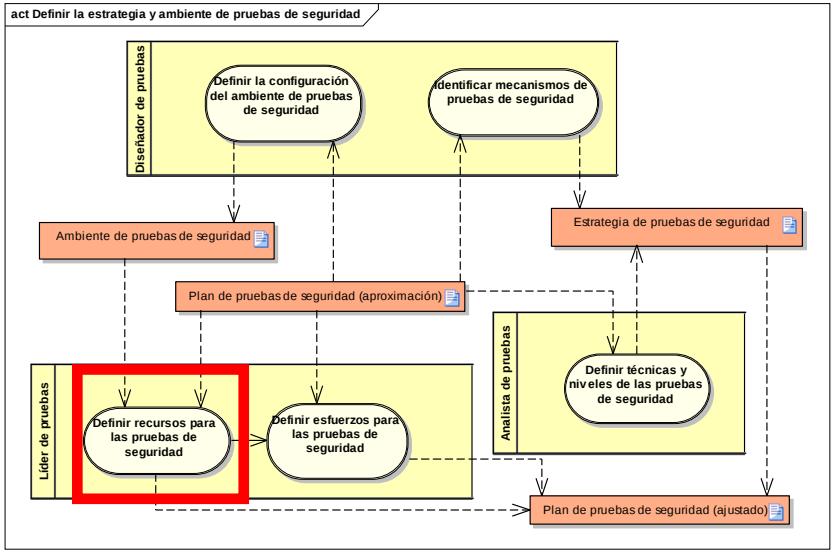
- Tomando como base las técnicas y niveles de pruebas utilizados en las pruebas de software convencionales, definir cuáles de ellos son apropiados para el desarrollo de las pruebas de seguridad del proyecto.
- De acuerdo a las técnicas y niveles definidos, establecer, si es necesario, las variaciones correspondientes para la implementación de la pruebas de seguridad.
- Definir una estrategia donde se especifica el orden y el ciclo iterativo que tendrán las técnicas y niveles de pruebas seleccionados.

Artefactos de entrada	*Artefactos resultantes*
• Plan de pruebas de seguridad	• Estrategia de pruebas de seguridad • Plan de pruebas de seguridad (ajustado)



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD

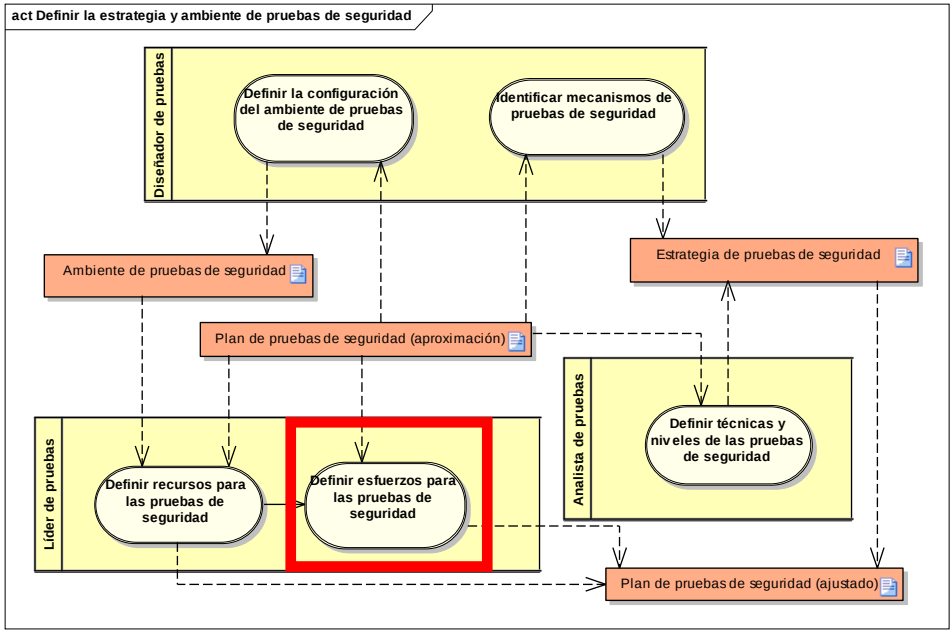


Definir recursos para las pruebas de seguridad	
Propósito Su objetivo es definir, de acuerdo a los recursos necesarios definidos en la documentación del ambiente de pruebas de seguridad, aquellos que van a ser implementados definitivamente en el ambiente de pruebas de seguridad de acuerdo a su disponibilidad tanto, técnica, tecnológica, económica y que se ajusten al cronograma propuesto.	
Rol Líder de pruebas	
Pasos •Definir el recurso humano y los roles que se tendrá disponible para el proceso de pruebas de seguridad •Definir que instalaciones (edificios, oficinas, etc.) se tendrán a disposición para el proceso de pruebas de seguridad. •Definir los recursos de hardware (servidores, terminales, conectividad, periféricos, etc.) que se tendrán a disposición para el proceso de pruebas de seguridad. •Definir los recursos de software (sistemas operativos, software específico para la ejecución de pruebas, software para el diseño de casos de prueba de ataque y reporte de defectos, etc.)que se tendrán a disposición para el proceso de pruebas de seguridad.	
Artefactos de entrada •Documentación del ambiente de pruebas de seguridad •Plan de pruebas de seguridad	*Artefactos resultantes* •Documento de especificación de recursos a utilizar en la fase de pruebas de seguridad •Plan de .pruebas de seguridad ajustado

MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LA ESTRATEGIA Y AMBIENTE DE PRUEBAS DE SEGURIDAD

Definir esfuerzos para las pruebas de seguridad	
Propósito Su objetivo es definir los tiempos y las cargas laborales que se tendrán de acuerdo a los recursos disponibles y el cronograma propuesto para el proceso de pruebas de seguridad	
Rol Líder de pruebas	
Pasos •Realizar una asignación de carga laboral a cada uno de los participantes del proceso de pruebas de seguridad. •Definir los horarios de trabajo de cada uno de los participantes. •Definir los recursos físicos que se darán a cada participante para la elaboración de su trabajo. •Establecer los horarios de funcionamiento y disponibilidad de los distintos recursos físicos.	
Artefactos de entrada •Plan de pruebas de seguridad	*Artefactos resultantes* •Cronograma de tiempos y tareas •Plan de pruebas de seguridad ajustado



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LAS PRUEBAS DE SEGURIDAD

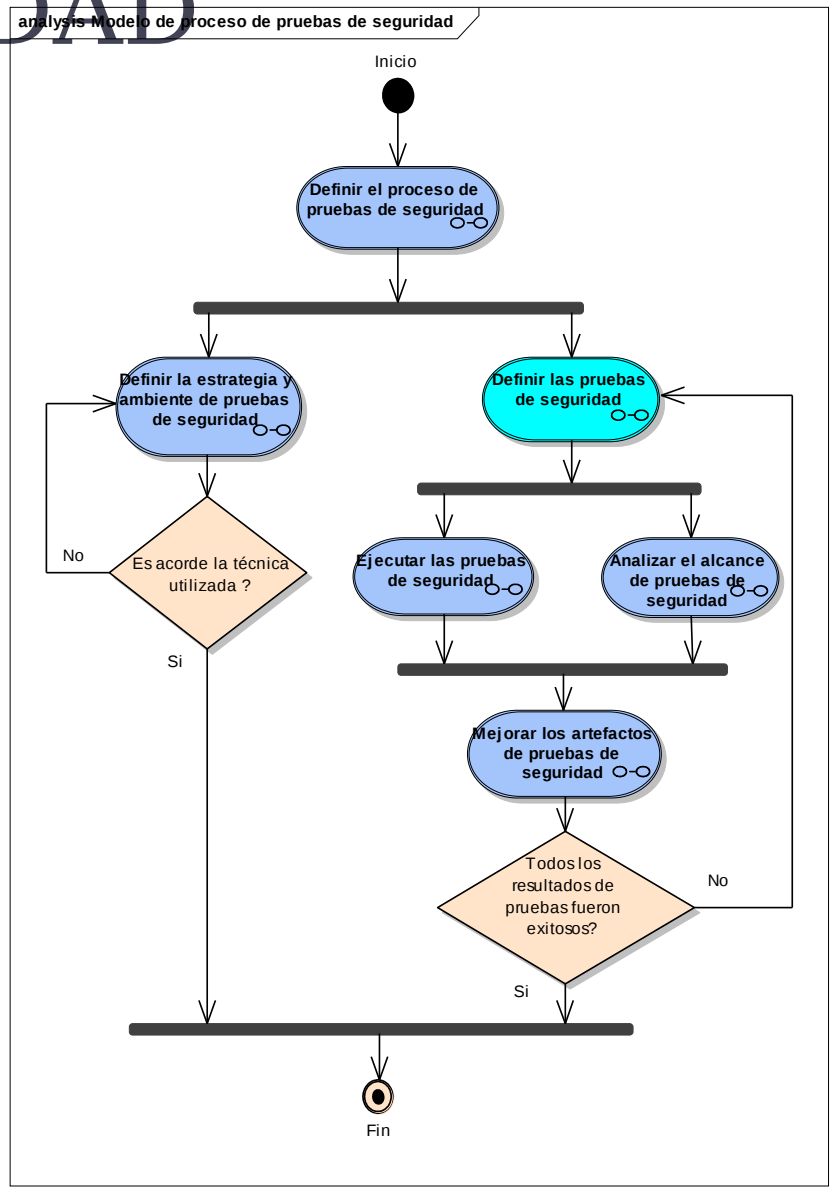
El objetivo de esta fase es determinar que pruebas de seguridad se van a realizar y bajo que niveles de calidad de seguridad va a ser medido el software para su aceptación.

Artefactos de entrada:

- Estrategia de pruebas de seguridad
- Catálogo de ideas de pruebas de seguridad

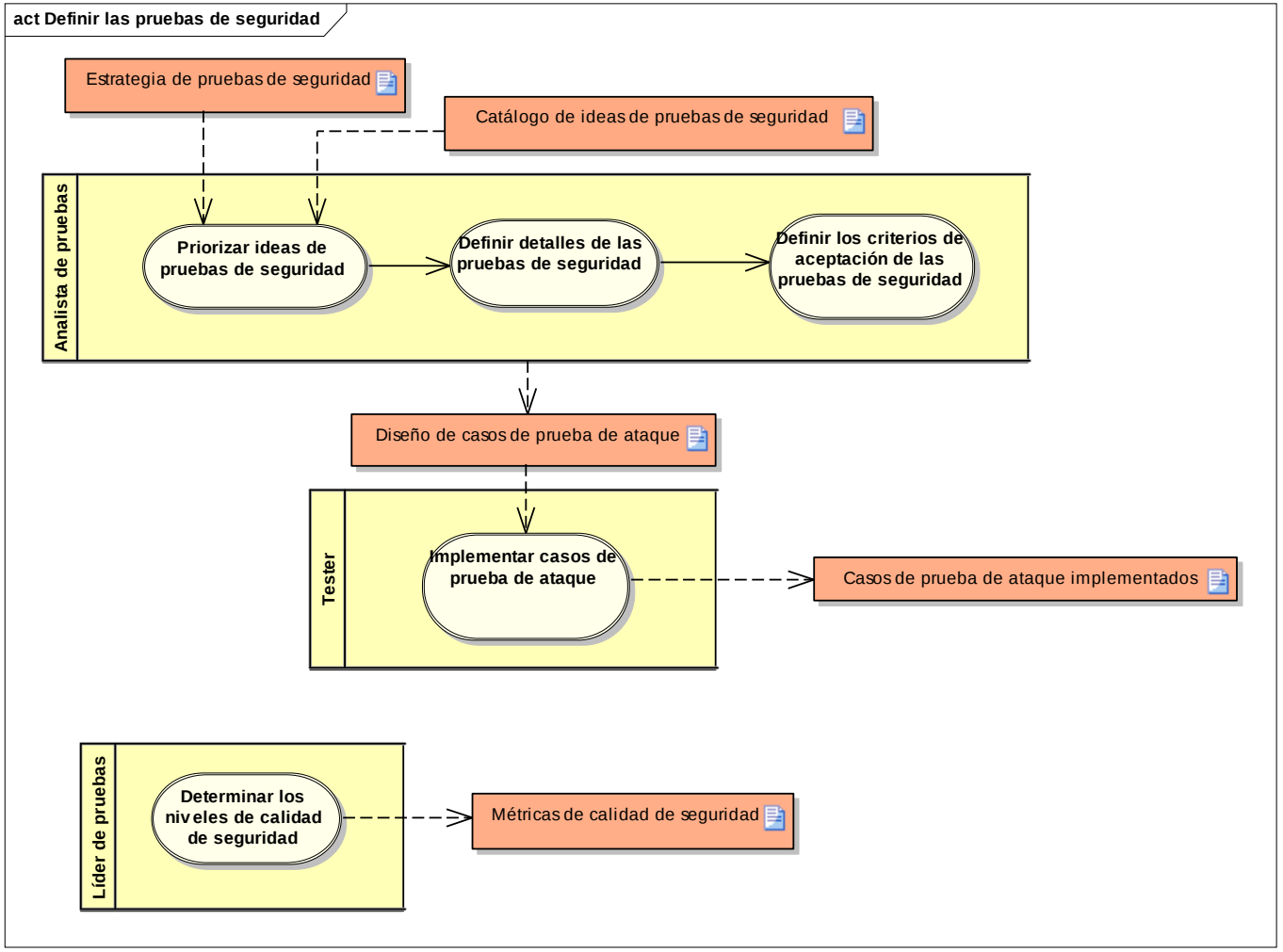
Artefactos generados:

- Diseño de casos de prueba de ataque
- Casos de prueba de ataque implementados
- Métricas de seguridad



MODELO DE PRUEBAS DE SEGURIDAD

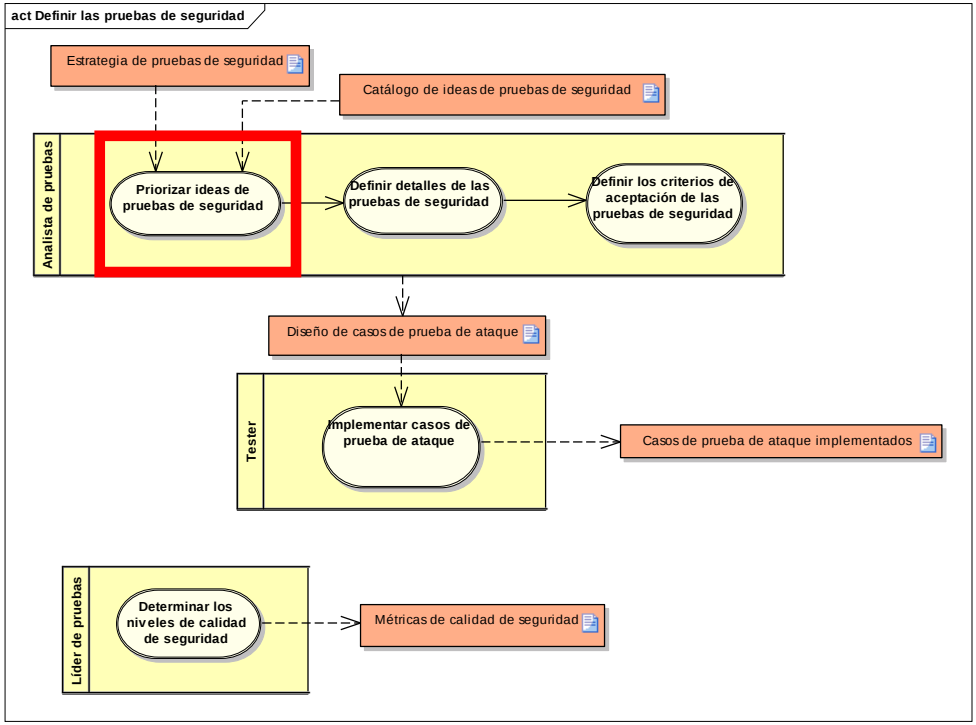
DEFINIR LAS PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

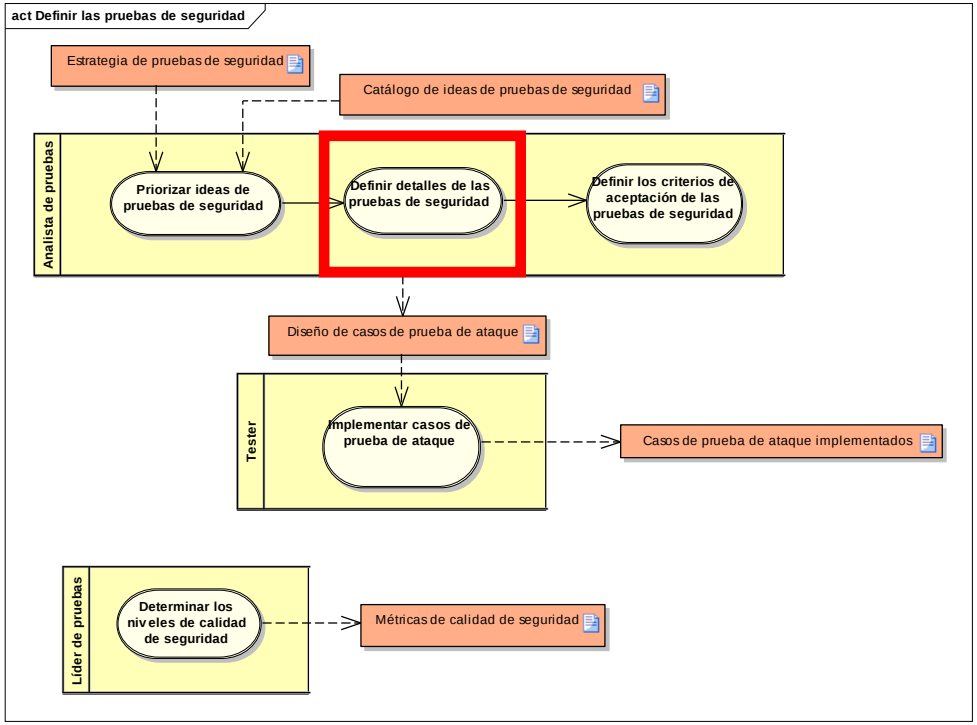
DEFINIR LAS PRUEBAS DE SEGURIDAD

Priorizar ideas de pruebas de seguridad	
Propósito Su objetivo es definir, dentro de todo el conjunto de ideas de pruebas de ataque definido, cuales son las más importantes para realizar.	
Rol Analista de pruebas	
Pasos •De acuerdo al modelamiento de amenazas realizado, asociar las ideas de pruebas de seguridad con la (s) correspondiente (s) amenaza (s). •Tras haber realizado la asociación, las ideas de pruebas tomarán la misma clasificación de prioridad que se obtuvo con las correspondientes amenazas.	
Artefactos de entrada •Estrategia de pruebas de seguridad. •Catálogo de ideas de pruebas de seguridad.	*Artefactos resultantes* •Diseño de casos de prueba de ataque



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LAS PRUEBAS DE SEGURIDAD

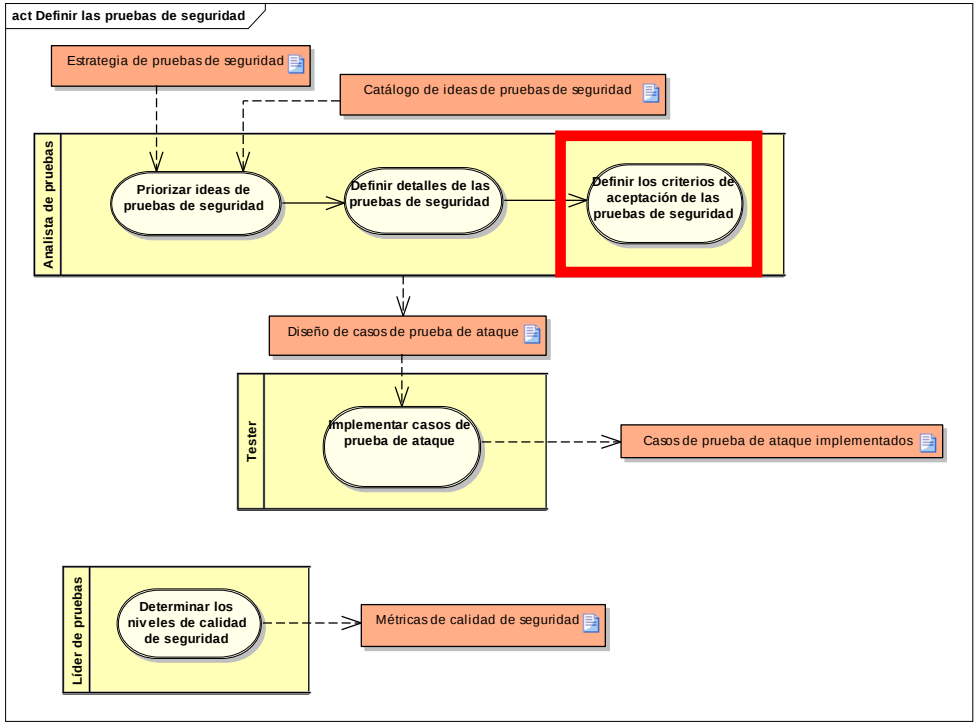


Definir detalles de pruebas de seguridad	
Propósito Su objetivo es definir con mayor detalle las ideas de pruebas de seguridad que fueron seleccionadas para ser implementadas.	
Rol Analista de pruebas	
Pasos •Por cada una de las ideas de pruebas seleccionadas para implementar definir bajo que escenarios se ejecutarán esas pruebas y una descripción más detallada de la forma y propósito de la prueba.	
Artefactos de entrada •Catálogo de ideas de pruebas de seguridad.	*Artefactos resultantes* •Diseño de casos de prueba de ataque

MODELO DE PRUEBAS DE SEGURIDAD

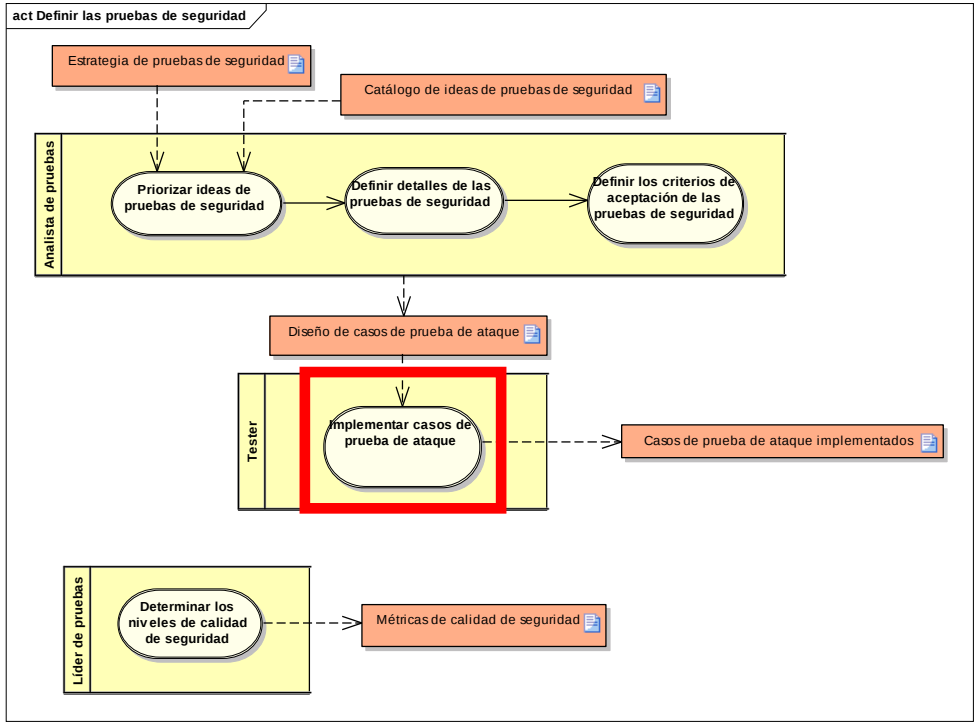
DEFINIR LAS PRUEBAS DE SEGURIDAD

Definir los criterios de aceptación de las pruebas de seguridad	
Propósito Su objetivo es definir los criterios sobre los cuales se considerará que cada prueba fue realizada exitosamente.	
Rol Analista de pruebas	
Pasos •Por cada una de las ideas de pruebas seleccionadas para implementar definir bajo qué criterios se decidirá si la prueba fue realizada exitosamente o si esta falló.	
Artefactos de entrada •Catálogo de ideas de pruebas de seguridad	*Artefactos resultantes* •Diseño de casos de prueba de ataque



MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LAS PRUEBAS DE SEGURIDAD

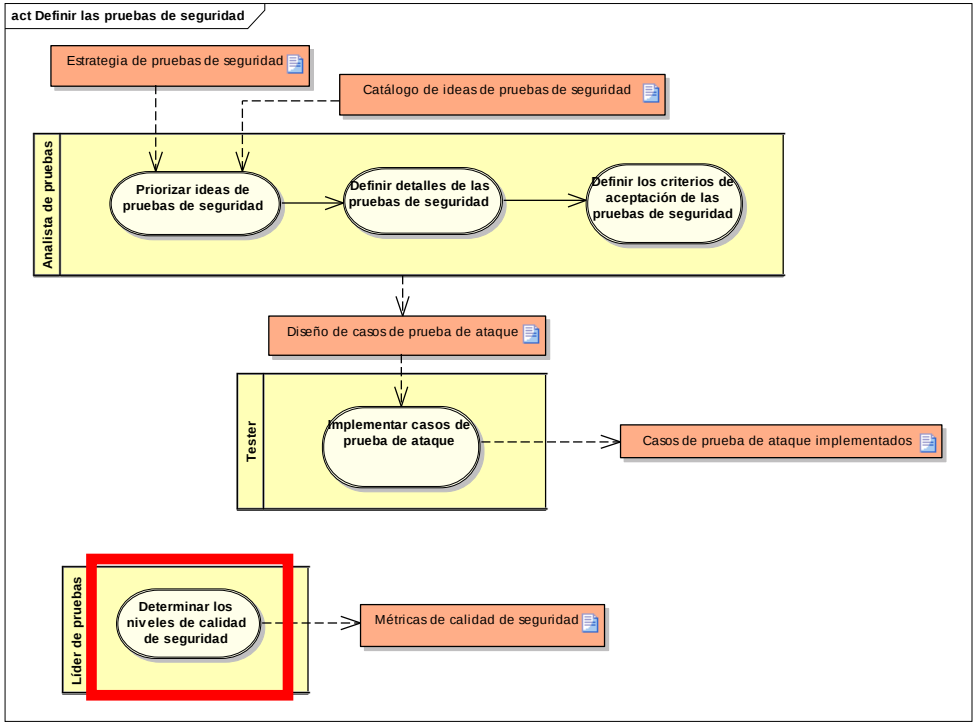


Implementar casos de prueba de ataque	
Propósito Su objetivo es definir detalladamente cada caso de prueba de ataque a implementar.	
Rol Probador	
Pasos •Por cada uno de los casos de prueba de ataque definidos definir los siguientes puntos: o Precondiciones del sistema antes de la ejecución del caso de prueba de ataque. oPoscondiciones del sistema después de la ejecución del caso de prueba de ataque. oPasos detallados de la prueba y resultado esperado del sistema por cada paso. oConjunto de datos a utilizar durante la ejecución del caso de prueba de ataque.	
Artefactos de entrada •Diseño de casos de prueba de ataque	*Artefactos resultantes* •Casos de prueba de ataque implementados.

MODELO DE PRUEBAS DE SEGURIDAD

DEFINIR LAS PRUEBAS DE SEGURIDAD

Determinar los niveles de calidad de seguridad	
Propósito Su objetivo es definir bajo que niveles de calidad referentes a la seguridad será evaluado el sistema.	
Rol Líder de pruebas	
Pasos •Definir bajo que normas y estándares de seguridad vigentes va a ser medido el sistema, teniendo en cuenta el modelo de seguridad definido. •De acuerdo a las normas y estándares definidos establecer que criterios afectan al sistema. •Determinar elementos cuantificables o cualificables por cada uno de los criterios definidos. •Establecer niveles de aceptación de cada uno de esos criterios o conjuntos de criterios. •Determinar la forma de evaluar cada uno de esos criterios.	
Artefactos de entrada •Normas y estándares de seguridad vigentes. •Aspectos no funcionales del sistema.	*Artefactos resultantes* •Métricas de seguridad.



MODELO DE PRUEBAS DE SEGURIDAD

EJECUTAR LAS PRUEBAS DE SEGURIDAD

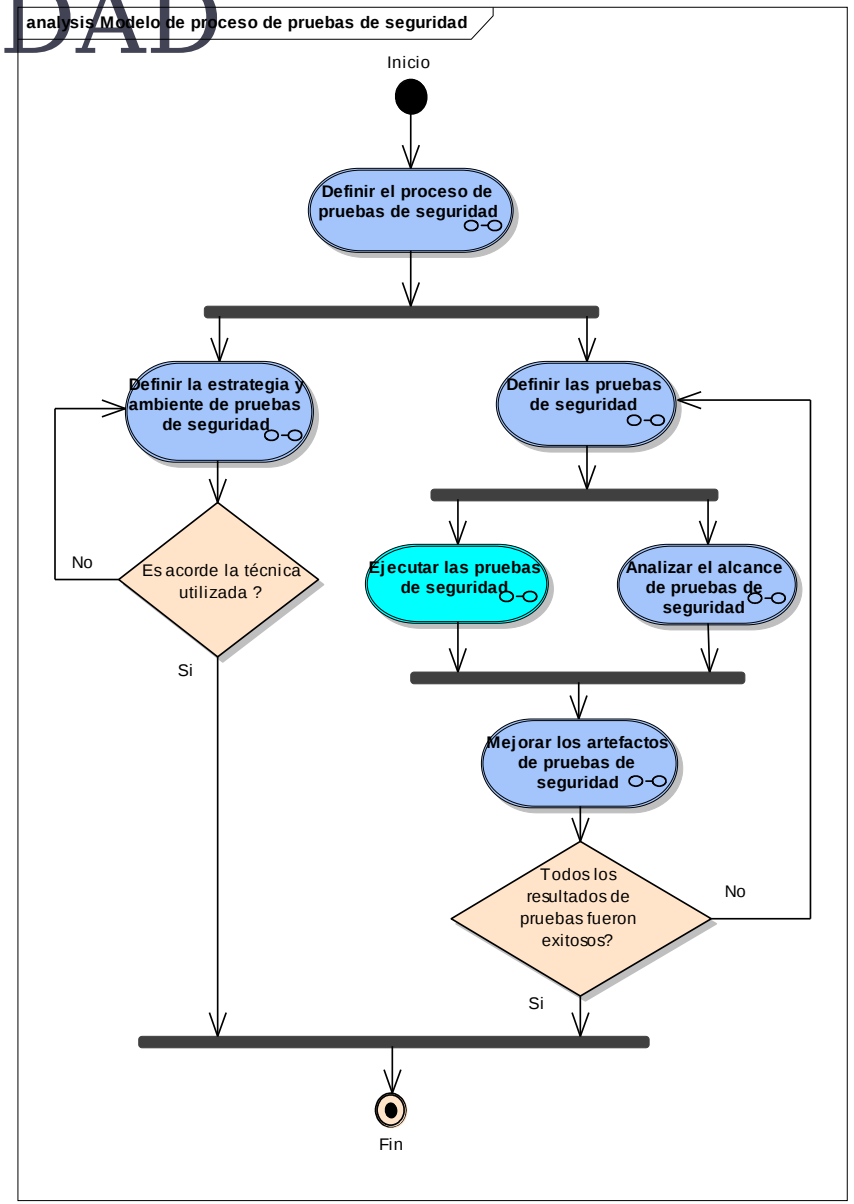
El objetivo de esta fase es llevar a cabo el conjunto de casos de prueba definidos en las fases anteriores, es decir, ejecutar y evaluar los resultados de ejecución de las pruebas definidas para el sistema, así como determinar si un componente es seguro o no.

Artefactos de entrada

- Ambiente de pruebas de seguridad
- Casos de prueba de ataque
- Prototipo funcional

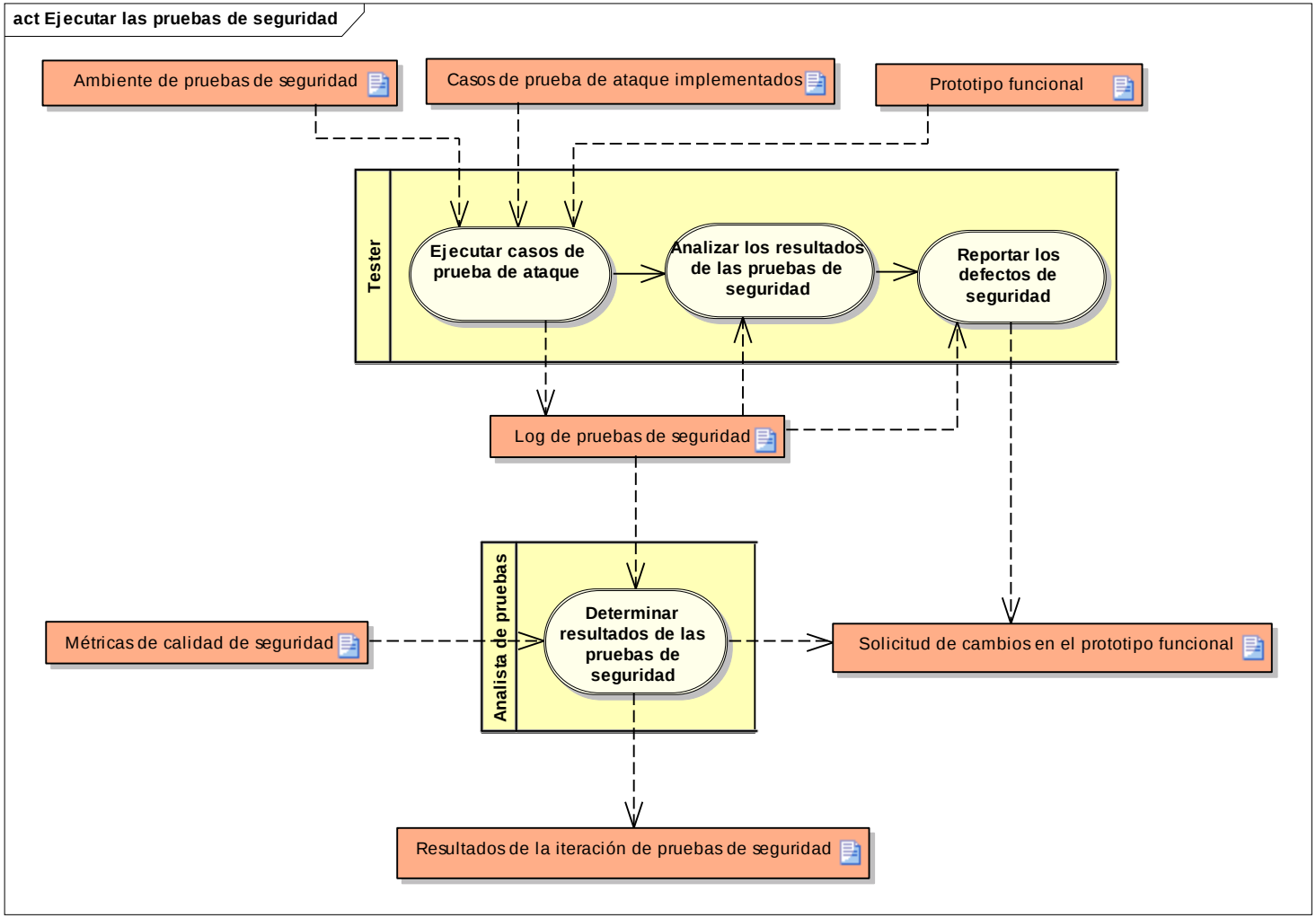
Artefactos generados

- “Logs” de pruebas de seguridad
- Resultados de pruebas de seguridad
- Solicitud de cambios en el prototipo funcional



MODELO DE PRUEBAS DE SEGURIDAD

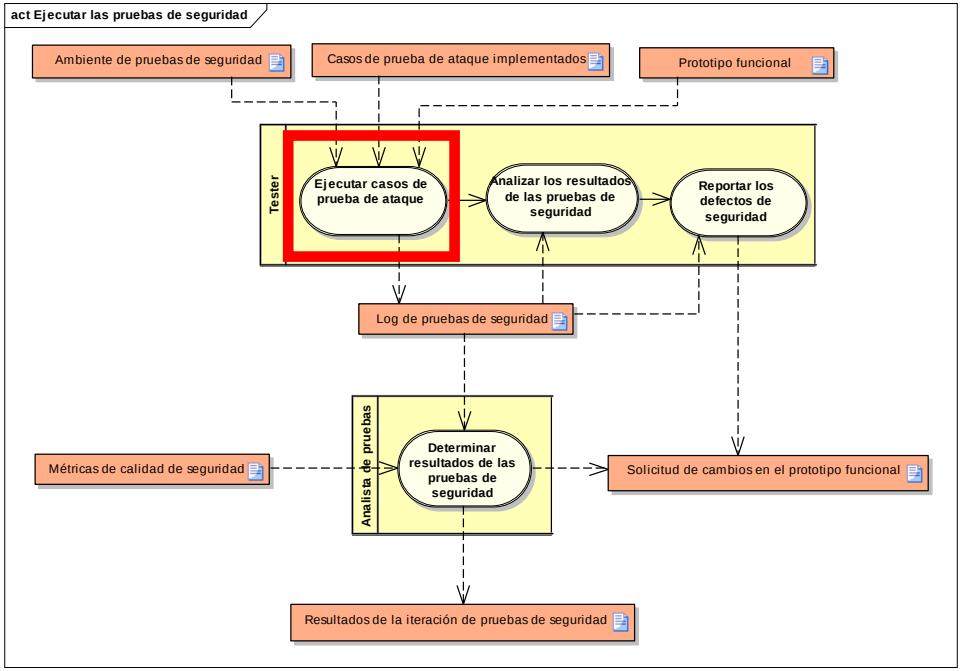
EJECUTAR LAS PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

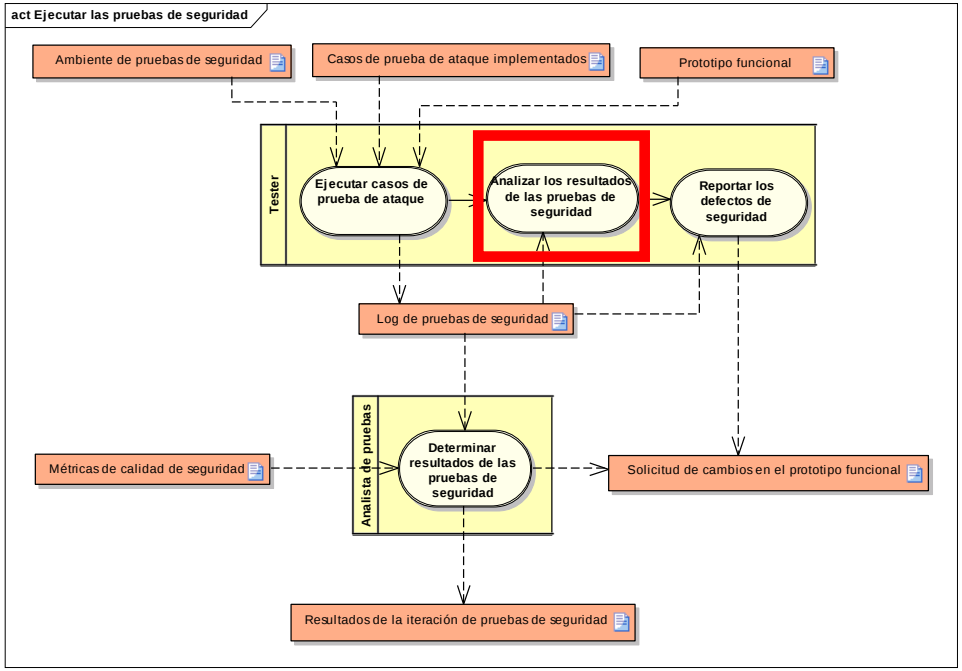
EJECUTAR LAS PRUEBAS DE SEGURIDAD

Ejecutar casos de prueba de ataque	
Propósito Su objetivo es ejecutar todo el conjunto de pruebas de seguridad definidas dentro del proceso.	
Rol Probador	
Pasos •Ejecutar la implementación de cada caso de prueba de ataque por cada componente a probar •Documentar cada entrada y paso realizado durante la prueba. •Documentar cada resultado obtenido durante la prueba. •Documentar los resultados finales obtenidos al ejecutar la prueba.	
Artefactos de entrada •Ambiente de pruebas de seguridad •Casos de prueba de ataque •Prototipo funcional	*Artefactos resultantes* •“Logs de pruebas de seguridad”.



MODELO DE PRUEBAS DE SEGURIDAD

EJECUTAR LAS PRUEBAS DE SEGURIDAD



Analizar los resultados de pruebas de seguridad	
Propósito Su objetivo es realizar un análisis sobre los resultados obtenidos al ejecutar las pruebas y determinar si la prueba fue exitosa o fallida.	
Rol Probador	
Pasos •Analizar los resultados obtenidos por cada caso de prueba de ataque. •De acuerdo al conjunto de resultados obtenidos determinar si el caso de prueba de ataque fue exitoso o fallido.	
Artefactos de entrada •“Logs de pruebas de seguridad”.	*Artefactos resultantes* •Resumen de resultados de las pruebas de seguridad.

MODELO DE PRUEBAS DE SEGURIDAD

EJECUTAR LAS PRUEBAS DE SEGURIDAD

Reportar los defectos de seguridad

Propósito

Su objetivo es generar un informe sobre los defectos encontrados y mejoras necesarias al componente probado para su solución por parte de los desarrolladores o del equipo de seguridad.

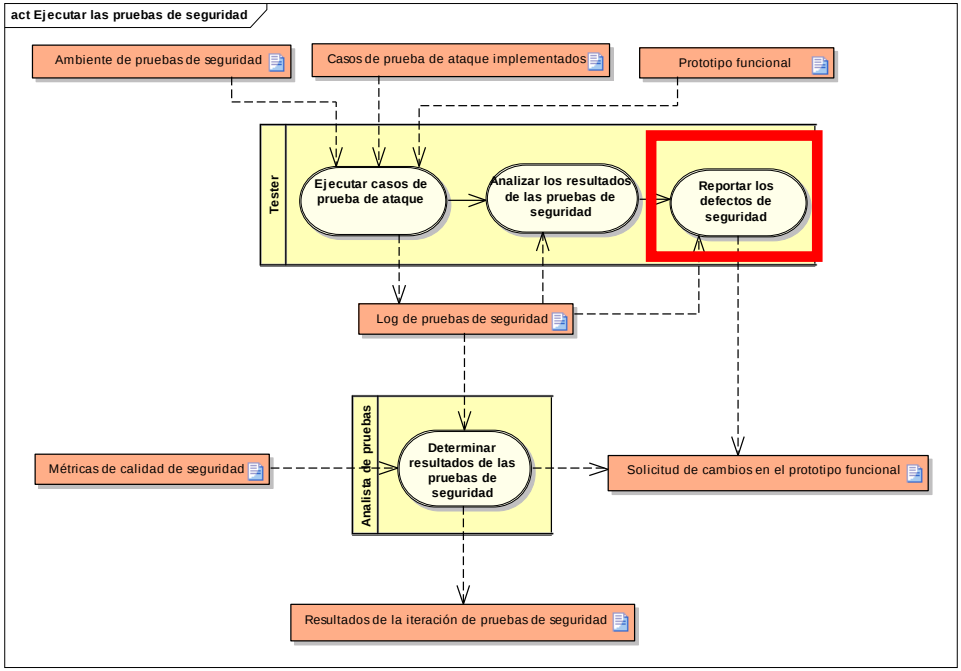
Rol

Probador

Pasos

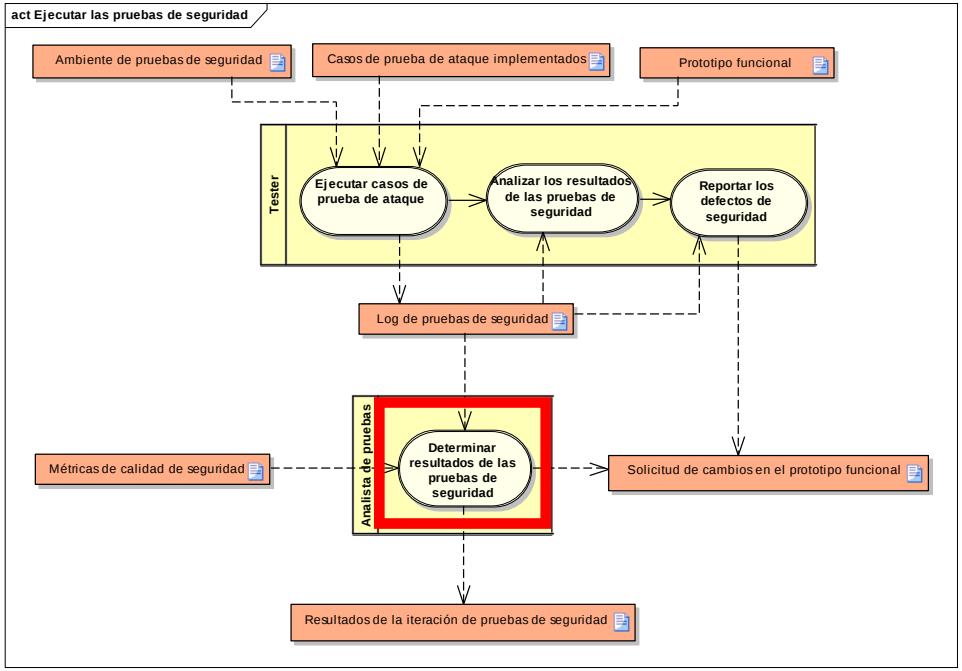
- Por cada defecto encontrado establecer:
  - Ubicación del defecto
  - Descripción del defecto
  - Desarrollo paso a paso para la ejecución del defecto
  - Gravedad del defecto
  - Responsable de la solución del defecto
- Por cada mejora sugerida establecer:
  - Ubicación de la mejora
  - Descripción de la mejora
  - Razones por las cuales se solicita la mejora
  - Responsable de la implementación de la mejora

Artefactos de entrada	*Artefactos resultantes*
• Resumen de resultados de pruebas de seguridad. • "Logs" de pruebas de seguridad.	• Solicitud de cambios en el prototipo funcional.



MODELO DE PRUEBAS DE SEGURIDAD

EJECUTAR LAS PRUEBAS DE SEGURIDAD



Determinar resultados de pruebas de seguridad	
Propósito Su objetivo es definir si la funcionalidad probada es segura o si por el contrario es necesario realizar mejoras en la misma.	
Rol Analista de pruebas	
Pasos •De acuerdo a los resultados obtenidos en todo el conjunto de pruebas asociado a la funcionalidad determinar si esta es segura o no. •En caso de que la funcionalidad probada sea considerada como insegura, establecer que mejoras se deben realizar a la misma y priorizar la implementación de las mismas.	
Artefactos de entrada •Métricas de seguridad. •Solicitud de cambios al prototipo funcional. •Reporte de resultados de pruebas.	*Artefactos resultantes* •Solicitud de cambios en el prototipo funcional. •Resultados de la iteración de pruebas de seguridad

MODELO DE PRUEBAS DE SEGURIDAD

ANALIZAR EL ALCANCE DE LAS PRUEBAS DE SEGURIDAD

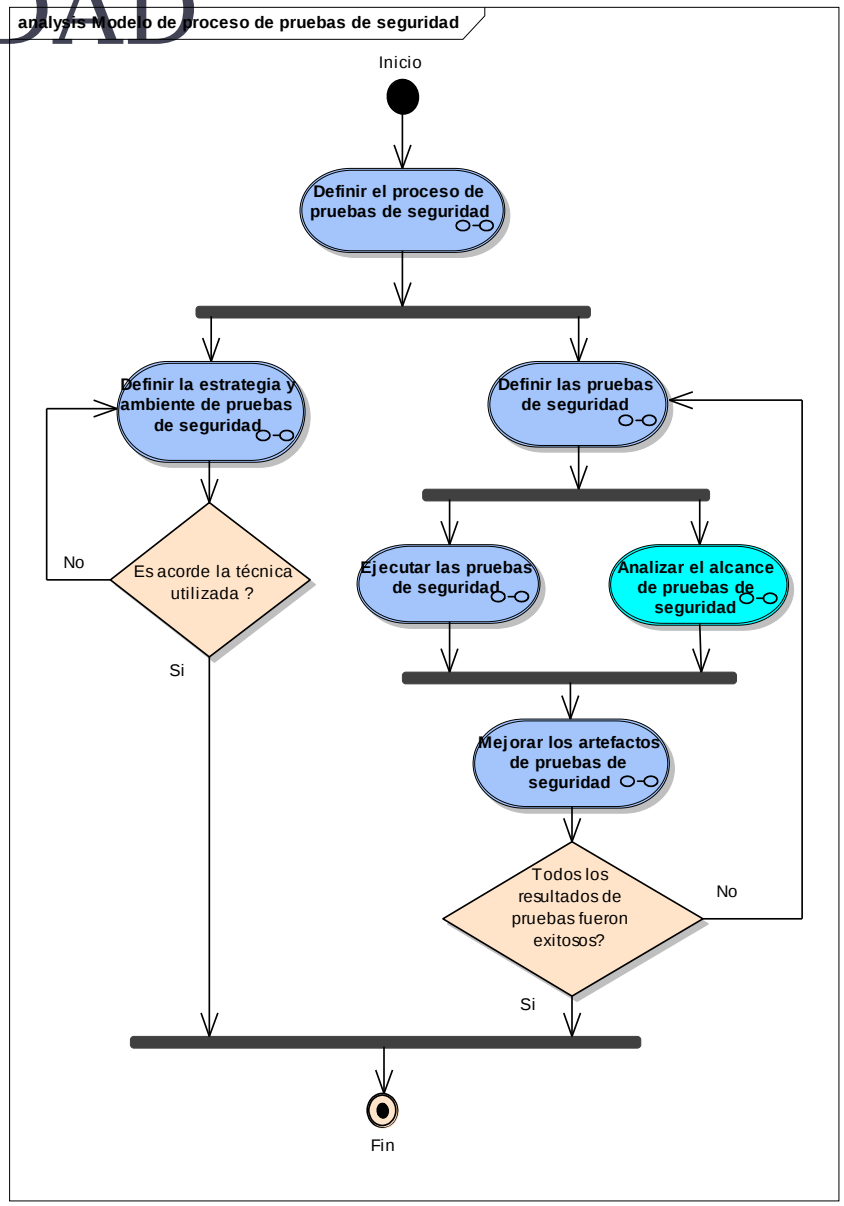
El objetivo de esta fase es el determinar el nivel de cobertura que tuvieron las pruebas en la iteración y si es necesario para una siguiente iteración mejorar los casos de prueba de ataque y/o la estrategia de pruebas de seguridad.

Artefactos de entrada

- Solicitud de cambios en el prototipo funcional
- Log de pruebas de seguridad
- Resultados de la iteración de pruebas de seguridad
- Métricas de calidad de seguridad
- Prototipo funcional

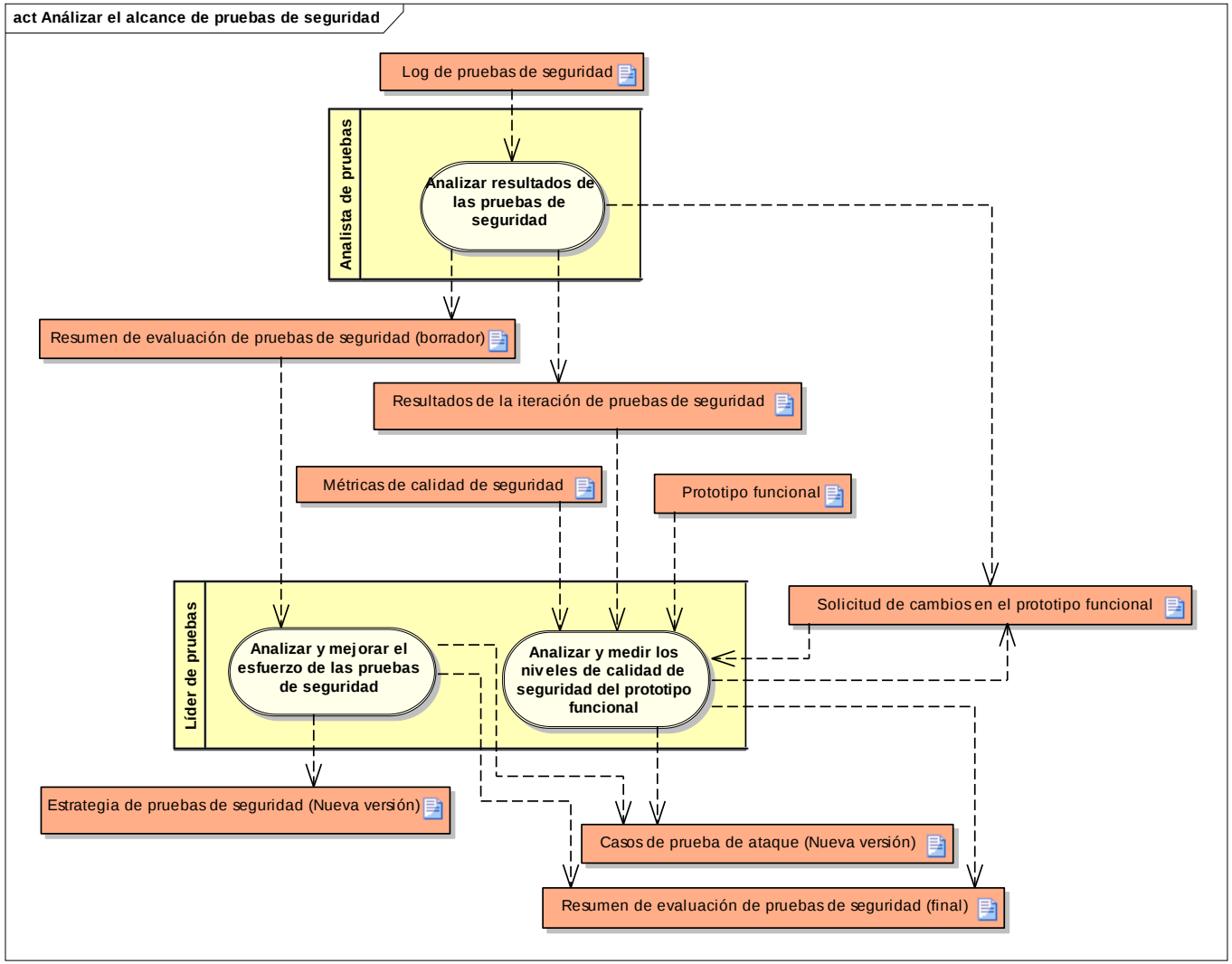
Artefactos generados

- Estrategia de pruebas de seguridad (nueva versión)
- Casos de prueba de ataque (nueva versión)
- Resumen de evaluación de pruebas de seguridad (final)
- Solicitud de cambios en el prototipo funcional



MODELO DE PRUEBAS DE SEGURIDAD

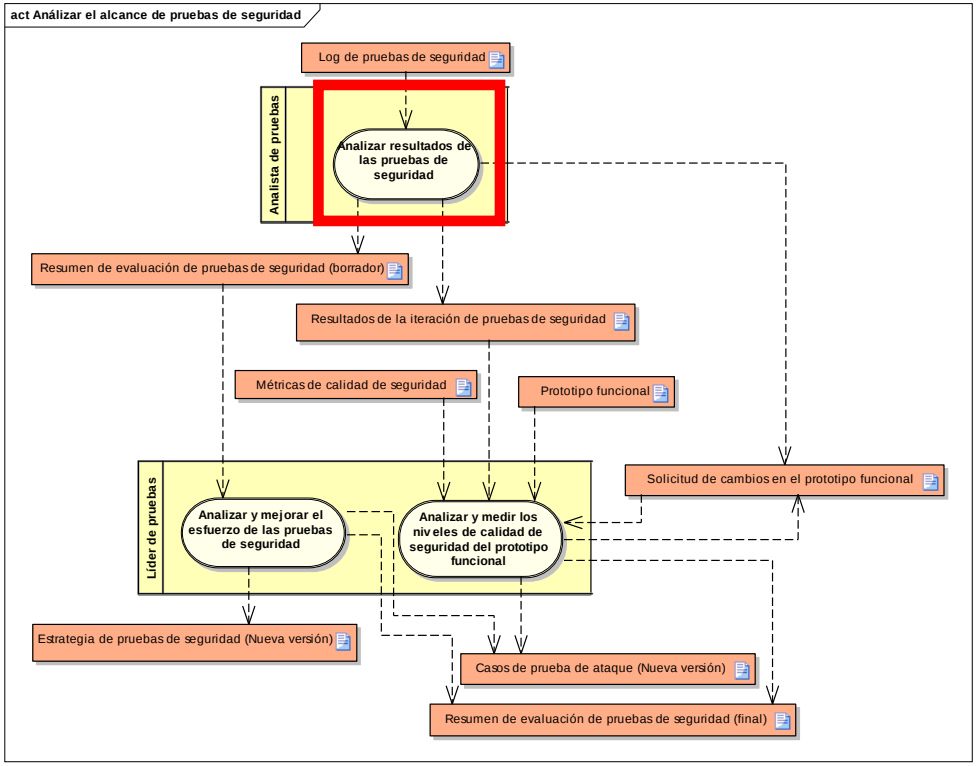
ANALIZAR EL ALCANCE DE LAS PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

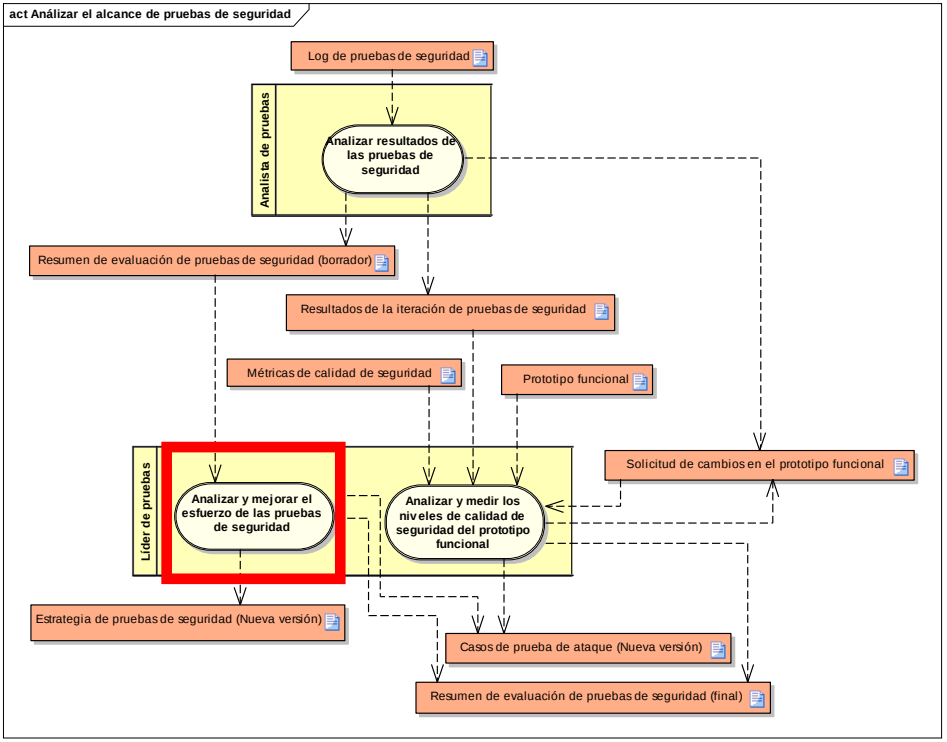
ANALIZAR EL ALCANCE DE LAS PRUEBAS DE SEGURIDAD

Analizar resultados de pruebas de seguridad	
Propósito Su objetivo es revisar los resultados obtenidos en la iteración de las pruebas de seguridad para así poder generar una evaluación del proceso llevado a cabo de acuerdo a los resultados esperados vs. Resultados obtenidos.	
Rol Analista de pruebas	
Pasos •Revisar los distintos resultados obtenidos por cada caso de prueba de ataque y el resultado esperado de cada uno de ellos. •Generar un resumen sobre los casos de ataque ejecutados en la iteración discriminando por casos de prueba de ataque exitosos, fallidos y cancelados. •Generar un reporte de los defectos encontrados con las respectivas solicitudes de cabios en el prototipo funcional.	
Artefactos de entrada •Log de pruebas de seguridad	*Artefactos resultantes* •Resumen de evaluación de pruebas de seguridad (borrador) •Resultados de la iteración de las pruebas de seguridad •Solicitud de cambios al prototipo funcional



MODELO DE PRUEBAS DE SEGURIDAD

ANALIZAR EL ALCANCE DE LAS PRUEBAS DE SEGURIDAD



Analizar y mejorar el esfuerzo de pruebas de seguridad

Propósito

Su objetivo es definir si el esfuerzo empleado en la iteración es suficiente o si por el contrario es excesivo, para realizar una mejor distribución del mismo en las futuras iteraciones.

Rol

Líder de pruebas

Pasos

- Revisar la asignación de carga laboral de cada uno de los participantes del proceso de pruebas de seguridad.
- Revisar los horarios de trabajo de cada uno de los participantes.
- Revisar los recursos físicos que se utilizaron por cada participante para la elaboración de su trabajo.
- Revisar los horarios de funcionamiento y disponibilidad de los distintos recursos físicos.
- Definir si estos esfuerzos fueron suficientes o si por el contrario falto una mejor distribución de los mismos o son excesivos.
- Redefinir la asignación de cargas laborales, horarios y recursos físicos de acuerdo a las necesidades encontradas en la iteración

Artefactos de entrada

- Resumen de evaluación de pruebas de seguridad (borrador)

Artefactos resultantes

- Estrategia de pruebas de seguridad (nueva versión)
- Casos de prueba de ataque (nueva versión)
- Resumen de evaluación de pruebas de seguridad (final)

MODELO DE PRUEBAS DE SEGURIDAD

ANALIZAR EL ALCANCE DE LAS PRUEBAS DE SEGURIDAD

Analizar y medir los niveles de calidad de seguridad

Propósito

Su objetivo es definir si, en la iteración de pruebas de seguridad, los subsistemas probados cumplen con los requisitos mínimos de calidad definidos.

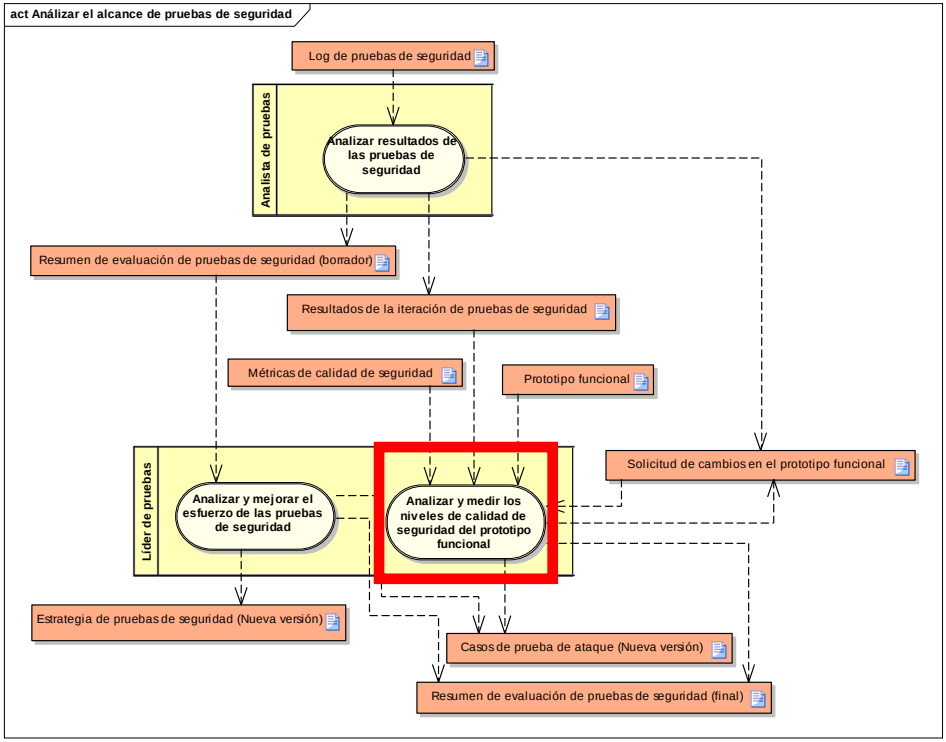
Rol

Líder de pruebas

Pasos

- Definir por cada subsistema los elementos cuantificables o cualificables a utilizar en las métricas de calidad de seguridad.
- Con base a esos elementos ejecutar las métricas de calidad de seguridad definidas.
- Definir si cada uno de los subsistemas probados en la iteración cumple con los niveles mínimos de calidad de seguridad definidos.
- En caso de que algún subsistema no cumpla con los niveles mínimos, generar un reporte donde se especifique que elementos deben ser mejorados para la siguiente iteración.

Artefactos de entrada	*Artefactos resultantes*
•Solicitud de cambios en el prototipo funcional •Métricas de calidad de seguridad •Prototipo funcional	•Solicitud de cambios en el prototipo funcional •Resumen de evaluación de pruebas de seguridad (final)



MODELO DE PRUEBAS DE SEGURIDAD

MEJORAR LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD

El objetivo de esta fase es definir, de acuerdo al alcance de las pruebas obtenido, que mejoras se pueden realizar en los artefactos de pruebas de seguridad en cuanto a cantidad y calidad de los mismos, para futuras iteraciones.

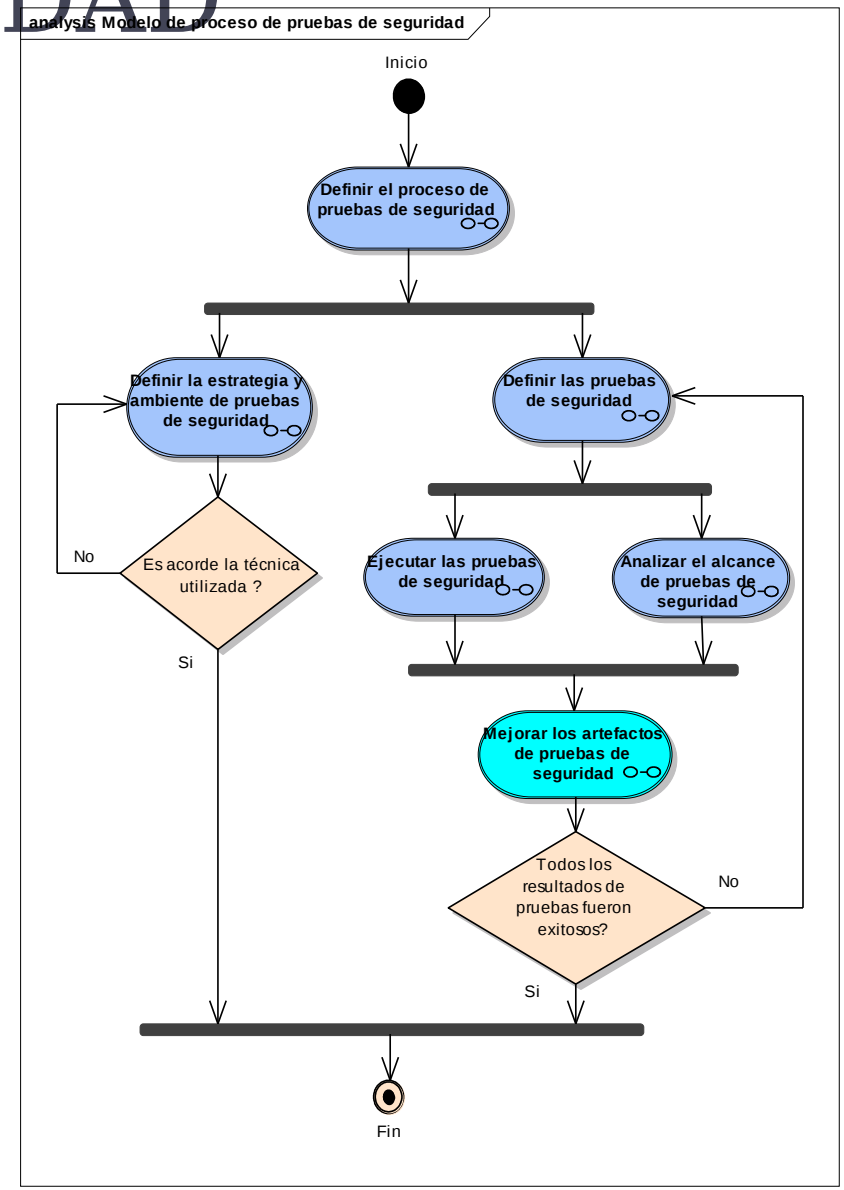
En este punto se genera el último ciclo de verificación en el cual si aún quedan pruebas de seguridad por realizar o defectos de seguridad por solucionar se vuelve a ejecutar todo el proceso de pruebas de seguridad o de lo contrario se da por finalizado el proceso de pruebas de seguridad.

Artefactos de entrada

- Resumen de evaluación de prueba de seguridad (final)

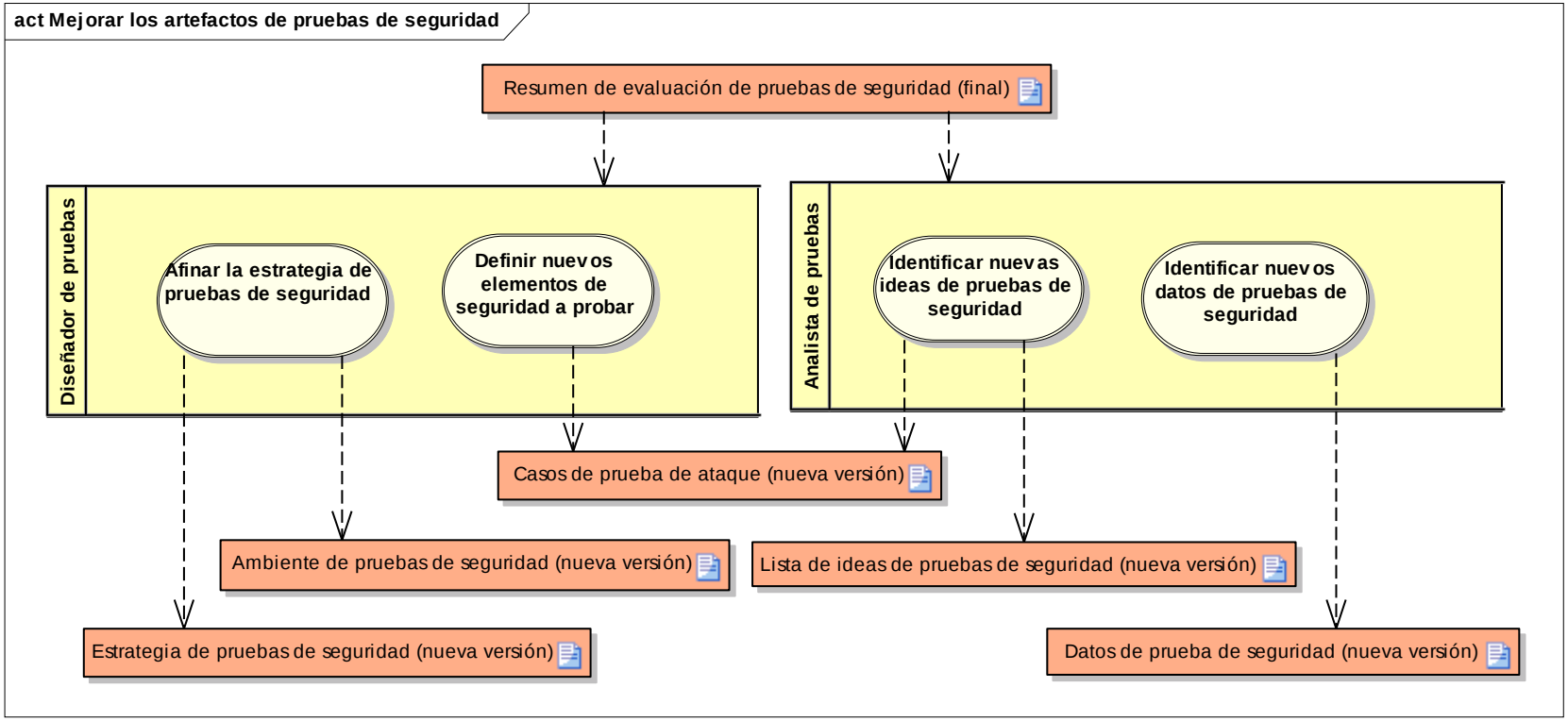
Artefactos generados

- Estrategia de pruebas de seguridad (nueva versión)
- Ambiente de pruebas de seguridad (nueva versión)
- Datos de prueba de seguridad (nueva versión)
- Datos de prueba de seguridad (nueva versión)
- Casos de prueba de ataque (nueva versión)



MODELO DE PRUEBAS DE SEGURIDAD

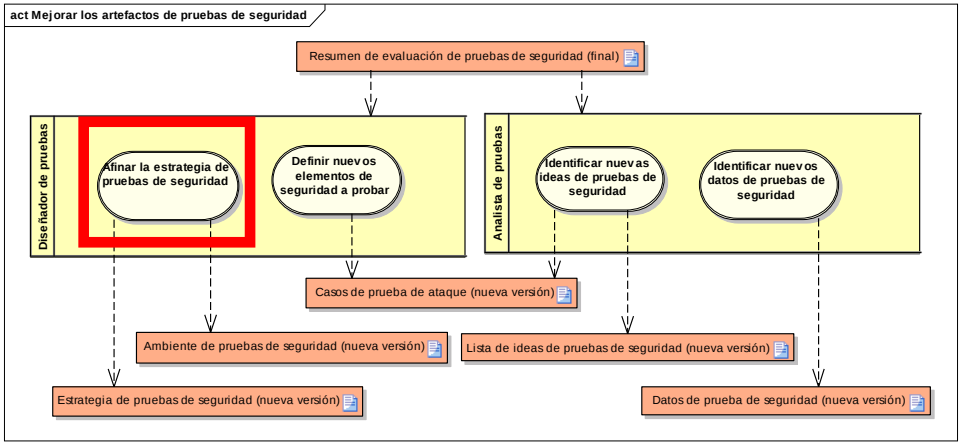
MEJORAR LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD



MODELO DE PRUEBAS DE SEGURIDAD

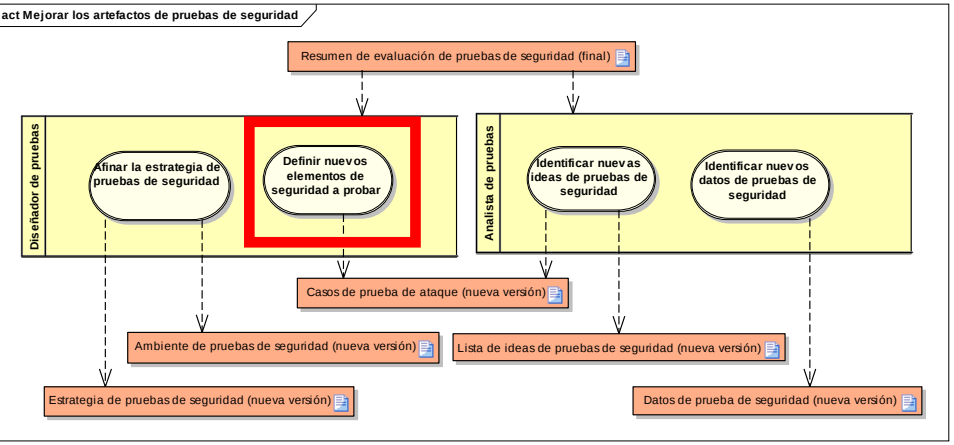
MEJORA DE LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD

Afinar la estrategia de pruebas de seguridad	
Propósito Su objetivo es definir nuevos puntos o nuevas técnicas, detectadas durante la iteración de pruebas, que permitan un mejor desarrollo del proceso en futuras iteraciones.	
Rol Diseñador de pruebas	
Pasos •Definir si el ambiente de pruebas de seguridad utilizado es acorde con las necesidades que se presentaron dentro de la iteración •Definir que técnicas, mecanismos y/o artefactos no utilizados dentro de la iteración pueden ayudar en ciclos futuros. •Analizar si estas técnicas, mecanismos y/o artefactos detectados pueden ser implementados con los recursos disponibles. •Implementar las técnicas, mecanismos y/o artefactos dentro de la estrategia de pruebas.	
Artefactos de entrada •Resumen de evaluación de pruebas de seguridad (final)	*Artefactos resultantes* •Estrategia de pruebas de seguridad (nueva versión)



MODELO DE PRUEBAS DE SEGURIDAD

MEJORA DE LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD



Definir nuevos elementos de seguridad a probar	
Propósito Su objetivo es definir nuevos elementos del prototipo funcional que no fueron considerados dentro de los casos de prueba de ataque y que son necesarios de probar para asegurar la calidad de sistema.	
Rol Diseñador de pruebas	
Pasos •Identificar que elementos no fueron considerados dentro de los casos de prueba de ataque. •Analizar la importancia de probar estos elementos, para el aseguramiento de la seguridad del sistema. •Definir un subconjunto de los elementos identificados que necesiten ser probados. •Generar los casos de prueba de ataque correspondientes para estos nuevos elementos.	
Artefactos de entrada •Resumen de evaluación de pruebas de seguridad (final)	*Artefactos resultantes* •Casos de prueba de ataque (nueva versión)

MODELO DE PRUEBAS DE SEGURIDAD

MEJORAR LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD

Identificar nuevas ideas de pruebas de seguridad

Propósito

Su objetivo es definir nuevas ideas de pruebas de seguridad que bien no fueron tenidas en cuenta por los diseñadores en la iteración de pruebas o que aparecen por el desarrollo de nuevas funcionalidades en el sistema, que no fueron contempladas inicialmente.

Rol

Analista de pruebas

Pasos

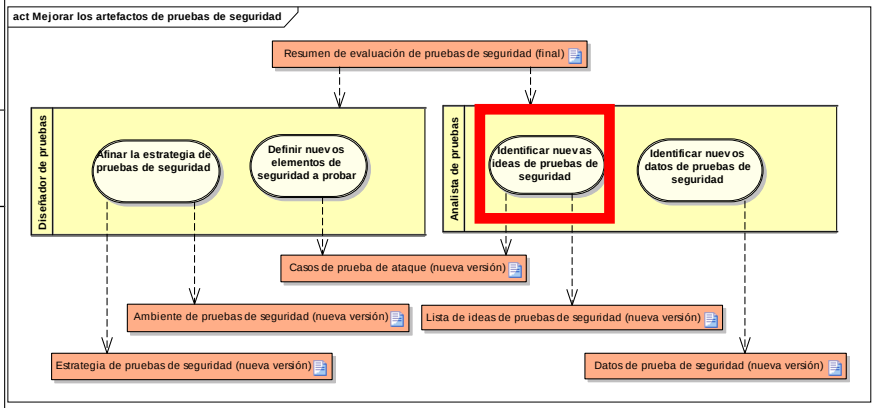
- Identificar si existen nuevas funcionalidades que no fueron contempladas inicialmente.
- Definir un conjunto de ideas de pruebas de seguridad para las nuevas funcionalidades identificadas.
- Analizar los resultados de prueba obtenidos durante la iteración
- Definir si las pruebas de seguridad generadas y ejecutadas durante la iteración son suficientes o si por el contrario existen pruebas, que no fueron consideradas por los diseñadores, y que necesitan ser implementadas
- Definir los nuevos casos de prueba de ataque para aquellas ideas que no fueron tenidas en cuenta en la iteración de pruebas de seguridad

Artefactos de entrada

- Resumen de evaluación de pruebas de seguridad (final)

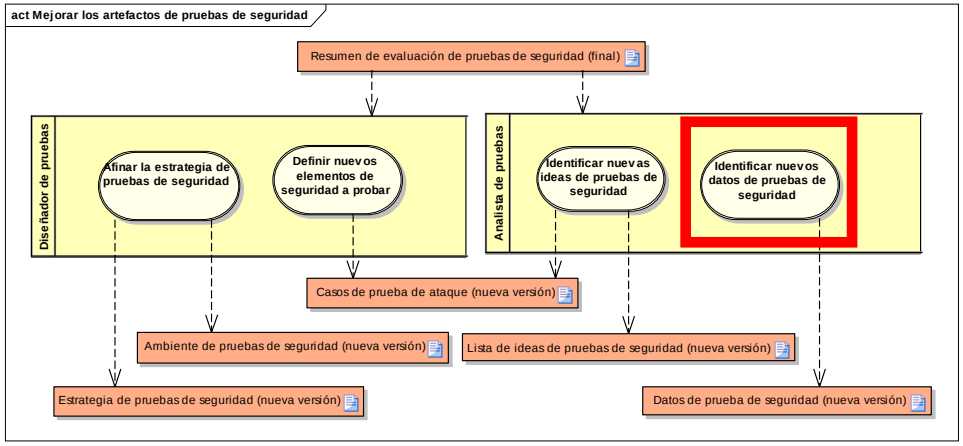
Artefactos resultantes

- Lista de ideas de pruebas de seguridad (nueva versión)
- Casos de prueba de ataque (nueva versión)



MODELO DE PRUEBAS DE SEGURIDAD

MEJORAR LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD



Identificar nuevos datos de pruebas de seguridad

Propósito

Su objetivo es definir nuevos datos en los casos de prueba de ataque que se encontraron dentro de la iteración, pero que no habían sido incluidos inicialmente en los casos de prueba de ataque y que generan resultados relevantes en el proceso de pruebas de seguridad.

Rol

Analista de pruebas

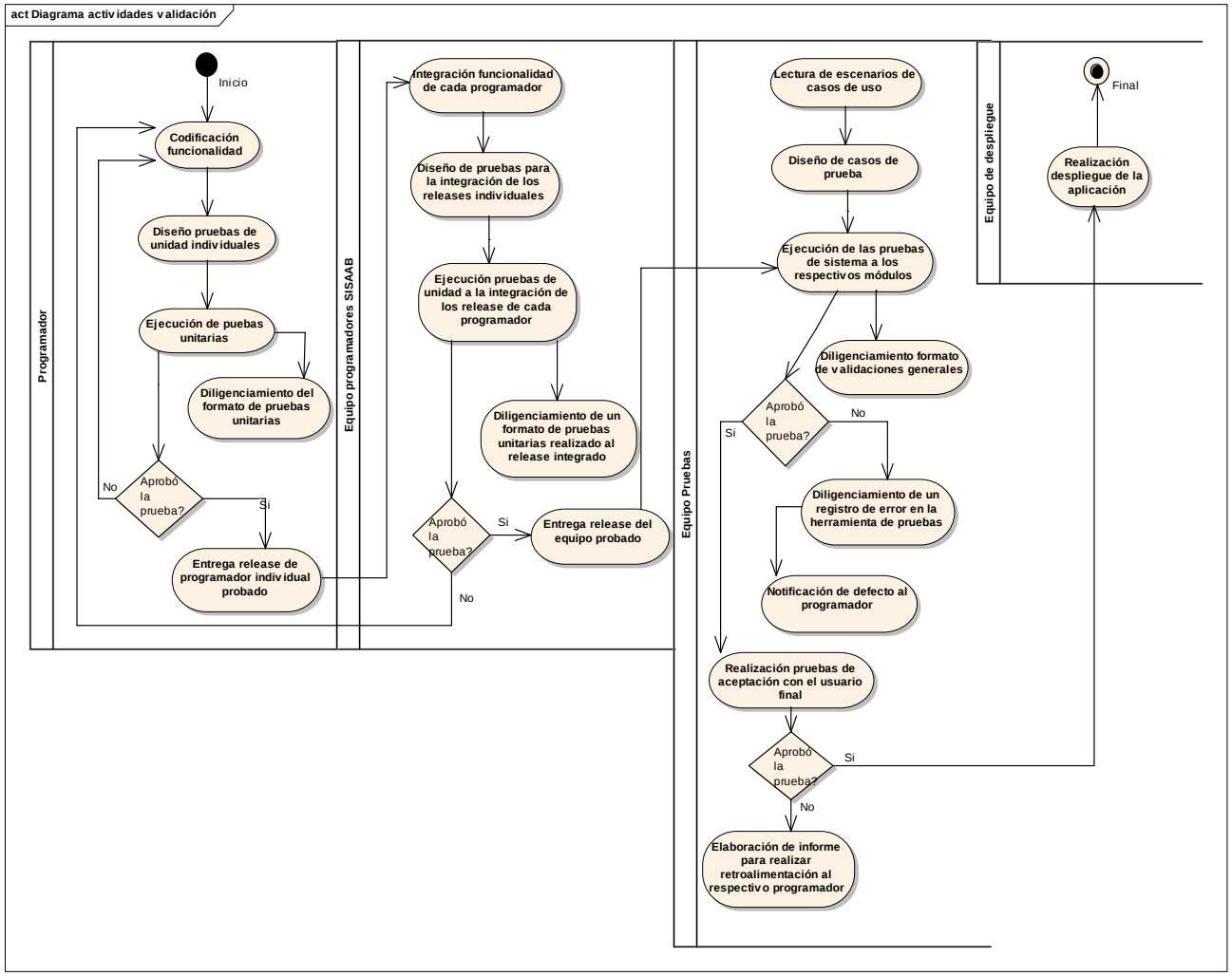
Pasos

- Analizar los resultados de pruebas en busca de nuevos datos de prueba que no fueron contemplados en la documentación de los casos de prueba de ataque y que causaron resultados relevantes en la iteración de pruebas de seguridad.
- Incluir los datos identificados en la especificación de los casos de prueba de ataque para que sean tenidos en cuenta en las futuras iteraciones.

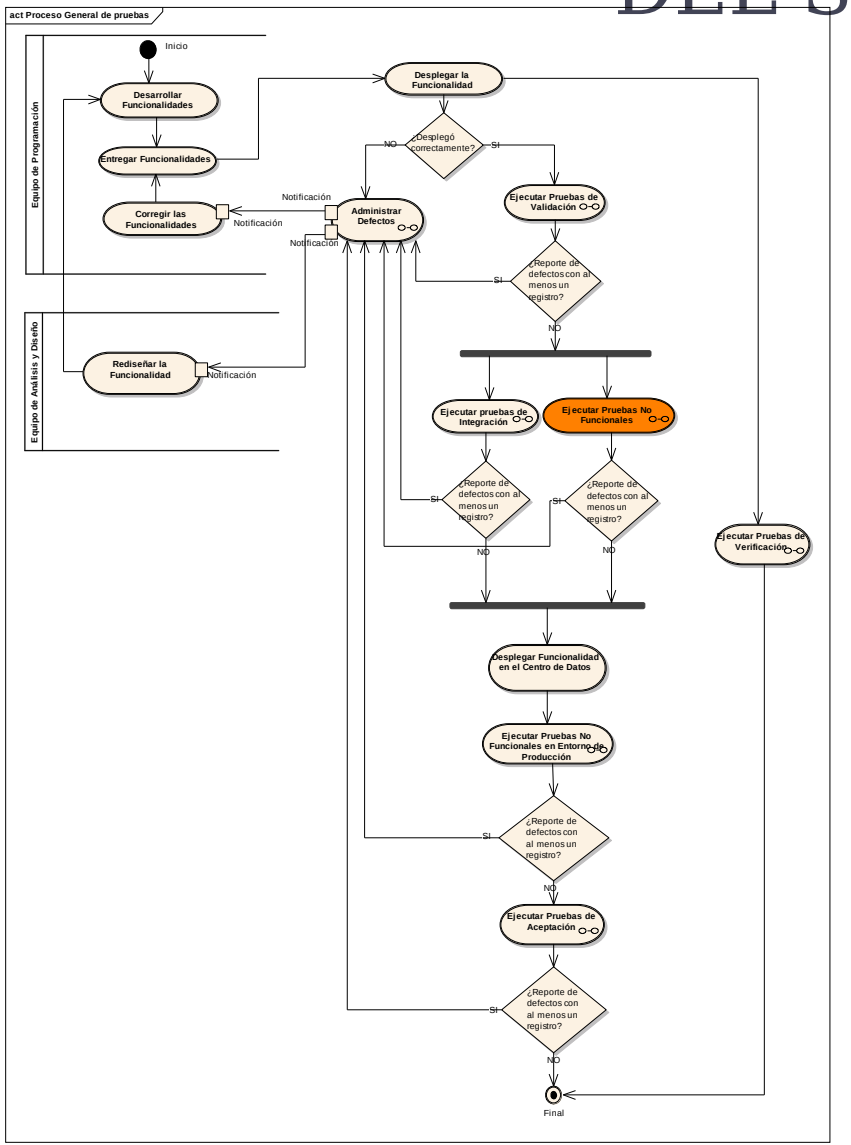
Artefactos de entrada	*Artefactos resultantes*
• Resumen de evaluación de pruebas de seguridad (final)	• Datos de prueba de seguridad (nueva versión)

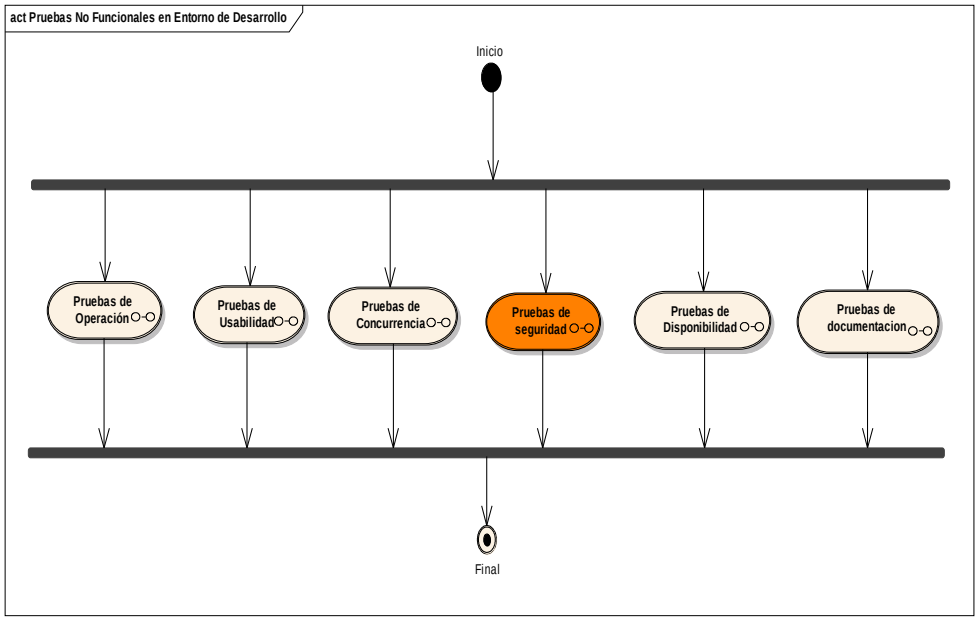
INTEGRACIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD AL PROCESO DE PRUEBAS DEL SI-SAAB

Al analizar el plan de pruebas del SI-SAAB, se determinó que se había tenido en cuenta una fase para generar las pruebas de seguridad, las cuales deberían encontrarse dentro del proceso de validación, sin embargo esta fase no se encuentra en el diagrama de procesos a tener en cuenta, es por esto que la integración del proceso de pruebas de seguridad debe ser realizada desde este punto.



INTEGRACIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD AL PROCESO DE PRUEBAS DEL SI-SAAB





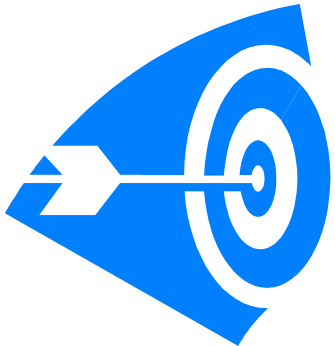
IMPLEMENTACIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD

DEFINICIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD

Los temas que se presentan a continuación hacen referencia a las distintas actividades definidas en el proceso de pruebas de seguridad referentes a la fase de definición del proceso y que dan como resultado el documento SISAAB-PRU-D24 PLAN DE PRUEBAS DE SEGURIDAD [V1.0][Iter1][2007-05-22] [Alvarez, et al., 2007]

Definir superficie de ataque

La lista que se presenta a continuación identifica aquellos subsistemas que son identificados como objetivos de las pruebas de seguridad para esta primera iteración, conocidos como la superficie de ataque del sistema.



- Subsistema de gestión de oferta y demanda
  - *Gestión de negociación*
    - Precios
    - Ofertas
    - Demandas
- Subsistema de operación informática
  - *Gestión de seguridad*
    - Administración de usuarios – roles – permisos
    - Gestión de seguridad control y seguimiento
  - *Gestión de auditoría y control*
    - Gestión de acceso al sistema
  - *Gestión de recuperación*
    - Administración de bases de datos
    - Administración de aplicaciones

DEFINICIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD

MODELAR AMENAZAS – IDENTIFICAR AMENAZAS [MEIER, 2005]

Categoría	Amenazas y ataques
Validación de datos de entrada	• "Buffer overflows" • "Cross-site scripting" • "SQL injection" • Ataques de conversión • Manipulación de cadenas de consulta • Manipulación de campos • Manipulación de "Cookies" • Manipulación de las cabeceras HTTP
Autenticación	• Escuchas de red ("Network eavesdropping") • Ataques de fuerza bruta • Ataques diccionario • Ataques de repetición de "Cookies" • Hurto de credenciales
Autorización	• Elevación de privilegios • Acceso a datos confidenciales • Tratar de forzar datos • Ataques de engaño
Administración de excepciones	• Revelación de detalles sensibles del sistema • Ataques de denegación de servicio

Categoría	Amenazas y ataques
Administración de configuración	• Acceso no autorizado a interfaces de administración • Acceso no autorizado a sitios de almacenamiento de configuraciones • Pasar a texto claro secretos de configuración • Falta de responsabilidades individuales
Datos sensibles	• Acceso a datos sensibles almacenados • Acceso a datos sensibles en la memoria • Escuchas de red ("Network eavesdropping") • Accesos de información
Administración de sesión	• Secuestro de sesión • Duplicación de sesión • Ataques del "Hombre en el medio"
Criptografía	•Perdidas de llaves de desencripción •Rompimiento de la encriptación
Auditoría y gestión de "logs"	•Negación de la realización de acciones por parte del usuario •Explotabilidad de la aplicación sin rastreo •El atacante cubre sus huellas

DEFINICIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD

MODELAR AMENAZAS - IDENTIFICAR VULNERABILIDADES [MEIER, 2005]

Categoría	Vulnerabilidades
Validación de datos de entrada	• Uso de entradas no validadas en el flujo de salida de HTML • Uso de entradas no validadas para la generación de consultas SQL • Confiar en la validación del lado del cliente • Usar nombres de archivos de entrada, URLs o nombres de usuarios para decisiones de seguridad • Usar los filtros de la aplicación sólo para entradas maliciosas • Búsqueda de malos patrones de entrada conocidos • Confiar en la lectura de datos de las bases de datos, archivos compartidos u otros recursos de red • Fallar en la validación de las entradas de todas las Fuentes, incluyendo “cookies”, parámetros de las cadenas de consulta, cabeceras HTTP, bases de datos y recursos de red

Categoría	Vulnerabilidades
Autenticación	• Uso de contraseñas débiles • Almacenamiento en texto claro de las credenciales en los archivos de configuración • Paso en texto claro de las credenciales por la red • Permitir el aumento de privilegios de las cuentas • Permitir largos periodos de vida de la sesión • Mezclar la personalización con la autenticación
Autorización	• Confiar en un solo mecanismo de entrada • Fallar en el aseguramiento de los recursos del sistema contra identidades de aplicación • Fallar en limitar el acceso a las bases de datos a procedimientos específicos de almacenamiento • Usar una separación inadecuada de los privilegios

DEFINICIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD

MODELAR AMENAZAS – IDENTIFICAR VULNERABILIDADES [MEIER, 2005]

Categoría	Vulnerabilidades
Administración de configuración	• Uso de interfaces de administración inseguras • Usar configuraciones de almacenamiento inseguras • Almacenar en texto claro los datos de configuración • Tener demasiados administradores • Usar cuentas con demasiados privilegios en procesos y servicios
Datos sensibles	• Almacenar secretos cuando no hay necesidad • Almacenar secretos en el código • Almacenar secretos en texto claro • Paso de datos privados en texto claro por la red
Administración de sesión	• Paso de los identificadores de sesión por canales no encriptados • Permitir largos periodos de vida de la sesión • Tener estados inseguros de almacenamiento • Colocar identificadores de sesión en cadenas de consulta

Categoría	Vulnerabilidades
Criptografía	• Uso de una criptografía personalizada • Usar el algoritmo incorrecto o un tamaño de clave que sea demasiado pequeño • Fallar en el aseguramiento de las llaves de encriptación • Usar la misma clave por un largo periodo de tiempo • Distribuir las claves de una manera insegura
Administración de excepciones	• Fallar en el uso de un manejo estructurado de excepciones • Revelar demasiada información al cliente
Auditoría y sesión de “logs”	• Fallar en la auditoría de los “logs” • Fallar en la protección de los archivos de auditoría • Fallar en la auditoría a través de las capas de la aplicación

DEFINICIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD

ENTIFICAR IDEAS DE PRUEBAS DE SEGURIDAD

Se planteó un catálogo de ideas de pruebas de seguridad de acuerdo a la superficie de ataque definida y las vulnerabilidades encontradas y priorizadas.

Subsitema				Ideas de pruebas de seguridad									
				Validación de entradas y salidas	Autenticación	Autorización	Configuración	Información sensible	Administración de sesión	Criptografía	Excepciones	Auditoría	Total
Operación informática	Gestión de seguridad	Administración de usuarios	Inicio de sesión	5	6	4	3	4	3	3	4	5	37
			Registro de usuarios	4	0	1	4	3	2	1	3	2	20
			Modificación de usuarios	3	0	1	4	3	3	1	3	2	20
			Eliminación de usuarios	2	0	1	3	1	1	0	3	2	13
			Control de acceso	2	0	1	4	1	1	0	3	2	14
		Gestión de seguridad, control y seguimiento	Generación de logs	0	1	1	1	1	1	0	3	1	9
	Gestión de auditoría y control	Gestión de acceso al sistema	Consulta de logs	1	1	1	1	1	0	0	3	2	10
	Gestión de recuperación		Administración de bases de	2	1	2	2	1	0	1	1	1	11
			Administración de aplicaciones	0	1	1	1	0	0	0	2	1	6
Gestión de oferta y demanda	Gestión de negociación	Registro de oferentes y demandandantes	Registro de oferentes	4	0	0	2	3	2	1	3	2	17
		Registro de ofertas y demandas	Registro de ofertas	1	1	3	0	2	1	0	3	1	12
			Registro de demandas	1	1	3	0	2	1	0	3	1	12
Total				25	12	19	25	22	15	7	34	22	181

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

MBIENTE DE PRUEBAS DE SEGURIDAD

Esta sección presenta todas las necesidades de recursos no humanos para la ejecución de las pruebas de seguridad. La utilización de estos recursos tiene como fin la creación de un ambiente similar al de producción y que permite realizar las pruebas de seguridad en las condiciones reales a las que se estará ejecutando el sistema.

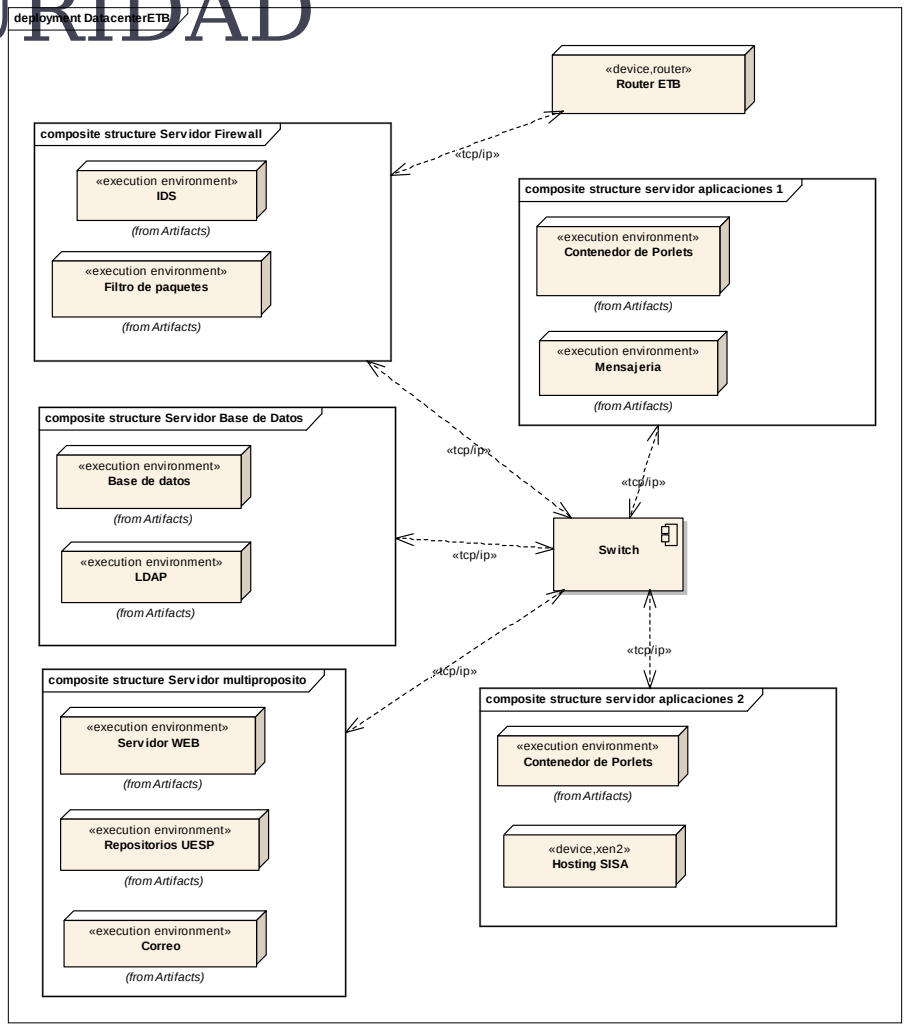


Diagrama de despliegue de servidores del SI-SAAB
Tomado de **(Guerrero, et al., 2006)**

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

MBIENTE DE PRUEBAS DE SEGURIDAD

Recursos del sistema		
Recurso	**Cantidad**	**Descripción**
Servidor	1	Servidor de aplicaciones
	1	Servidor de respaldo
	1	Servidor de bases de datos
	1	Servidor “Firewall”
Computadores	4	Clientes (uno por cada integrante del equipo de pruebas)
“Switch”	1	Comunicación entre los distintos componentes del sistema
“Router”	1	Comunicación entre el ambiente del sistema y los clientes externos
Conexiones de red		Conexiones de red aisladas entre los servidores y el “Switch”
	1	Conexión de red entre el servidor “firewall” y el “router”
		Conexiones entre el “router” y los clientes

Tipo	Descripción	Licencia	Nombre
Sniffers	Sirve para realizar escuchas de red	Libre	Tcpdump
			Ethereal
			Snort
			Dsniff
		Comercial	AeroPeek
			EtherPeak
			Agilent Network Analyzer
			Network General Sniffer
Debuggers	Es utilizado para la compilación de programas y el análisis de código fuente	Libres	Gdb
		Comercial	Microsoft Developer Studio
			SoftIce

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

MBIENTE DE PRUEBAS DE SEGURIDAD

Tipo	Nombre	Versión	Descripción
Sistemas operativos	Linux	Por definir	Sistema operativo por defecto para los servidores
		Por definir	Sistema operativo opcional para los clientes
	Windows	XP profesional	Sistema operativo para los clientes
Navegadores web	Mozilla firefox	1.5	Navegador web
		2.0	Navegador web
	Internet Explorer	6.0	Navegador web
		7.0	Navegador web
Procesadores de texto	Open office	2.0	Herramienta para la generación de documentos
Herramienta del proceso de pruebas	QaTraq	6.6	Diseño de casos de prueba
	Bugzilla	2.22.1	Reporte de defectos
Herramienta del proceso de soporte de desarrollo	Enterprise architect	6.5	Análisis y diseño del sistema
	JDK	1.5	Maquina virtual de java
	JBoss	4.0.4	Servidor de aplicaciones
	LifeRay	4.1	Contenedor de portlets
	pgAdmin	1.4.2	Herramienta de soporte de la base de datos

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

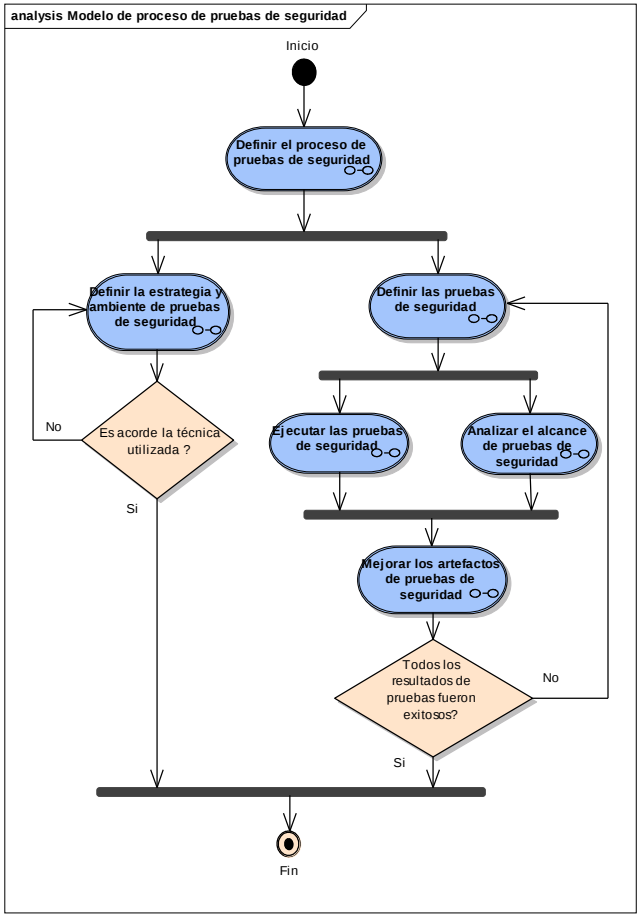
EFINICIÓN DE ESFUERZOS EN LAS PRUEBAS DE SEGURIDAD

Recurso humano		
Rol	**Recursos mínimos recomendados**	**Responsabilidades específicas**
Líder de pruebas de seguridad	1	•Planeación y logística •Definición de la misión •Identificar motivadores •Conseguir los recursos apropiados •Generar los reportes de administración •Defender los intereses de las pruebas de seguridad •Evaluar la efectividad del esfuerzo de pruebas de seguridad
Analista de pruebas de seguridad	2	•Identificar ideas de pruebas de seguridad •Definir detalles de pruebas de seguridad •Determinar los resultados de las pruebas de seguridad •Documentar las solicitudes de cambios •Evaluar la calidad de la seguridad del producto
Diseñador de pruebas de seguridad	4	•Definir el enfoque de las pruebas de seguridad •Verificar las técnicas de pruebas de seguridad •Definir los elementos a probar •Estructurar la implementación de las pruebas de seguridad
Probador	4	•Implementar los conjuntos de casos de prueba de ataque •Ejecutar los conjuntos de casos de prueba de ataque •Generar los “logs” de los resultados de las pruebas de seguridad •Determinar las fallas de seguridad del sistema •Documentar los incidentes

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

EFINIR ESTRATEGIA Y TÉCNICAS DE LAS PRUEBAS DE SEGURIDAD

Estrategia de pruebas de seguridad – Flujo de procesos



DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

EFINIR ESTRATEGIA Y TÉCNICAS DE LAS PRUEBAS DE SEGURIDAD

Estrategia de pruebas de seguridad – Criterios de inicio y terminación

Criterios de inicio de la fase de pruebas de seguridad

- Análisis del modelo de seguridad del SI-SAAB por parte del equipo de seguridad.
- Diseño del modelo de seguridad del SI-SAAB por parte del equipo de seguridad.
- Implementación del modelo de seguridad del SI-SAAB por parte del equipo de seguridad.

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

EFINIR ESTRATEGIA Y TÉCNICAS DE LAS PRUEBAS DE SEGURIDAD

Estrategia de pruebas de seguridad – Criterios de inicio y terminación

Criterios de suspensión y reanudación de la fase de pruebas de seguridad

Suspensión:

- Se están generando modificaciones al modelo de seguridad o al sistema en general que afectan la iteración de pruebas.
- Los recursos necesarios para la ejecución de las pruebas de seguridad no están disponibles temporalmente.
- No se han generado prototipos funcionales sobre los cuales se puedan ejecutar las pruebas de seguridad.
- Suspensión temporal del proyecto

Reanudación:

- Las modificaciones al modelo de seguridad o al sistema en general que afectaban la fase de pruebas de seguridad ya fueron realizadas.
- Ya se puede contar con los recursos necesarios para la ejecución de las pruebas de seguridad que no estaban disponibles temporalmente.
- Se han generado prototipos funcionales sobre los cuales se pueden ejecutar las pruebas de seguridad.
- Reanudación del proyecto.

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

EFINIR ESTRATEGIA Y TÉCNICAS DE LAS PRUEBAS DE SEGURIDAD

Estrategia de pruebas de seguridad – Criterios de inicio y terminación

Criterios de finalización de la fase de pruebas de seguridad

- Todos los casos de prueba de ataque definidos fueron ejecutados y con resultados exitosos.
- No existen nuevas consideraciones de seguridad a tener en cuenta para ser probadas.
- No existen nuevas ideas de pruebas de seguridad que necesiten ser implementadas y probadas.
- Finalización inesperada del proyecto.

DEFINR LA ESTRATEGIA Y AMBIENTE DE LAS PRUEBAS DE SEGURIDAD

EFINIR ESTRATEGIA Y TÉCNICAS DE LAS PRUEBAS DE SEGURIDAD

Estrategia de pruebas de seguridad – Artefactos derivados

Resúmenes de evaluación de pruebas de seguridad: Estos documentos contendrán información acerca de cada iteración de pruebas indicando que funcionalidades fueron probadas, cuales superaron exitosamente las pruebas y cuales requieren ser mejoradas y que porcentaje del total de pruebas de seguridad fue ejecutado en la iteración y en el acumulado de iteraciones.

Reportes de métricas de calidad: Estos reportes indicaran los resultados obtenidos al ejecutar las métricas de calidad en cada iteración dando una perspectiva del nivel de calidad que se va desarrollando paralelamente con el producto

Reporte de defectos: Estos reportes contendrán información detallada de los defectos encontrados al ejecutar las pruebas de seguridad y de las mejoras que se deben realizar a las funcionalidades probadas para que estas superen la fase de pruebas de seguridad.

DEFINR LAS PRUEBAS DE SEGURIDAD

La distribución de los conjuntos de pruebas de seguridad, se ha realizado por subsistema, para poder verificar que un subsistema es seguro respecto a todas las categorías de seguridad. Es decir que todas las categorías de seguridad identificadas en el modelamiento de amenazas realizado, se aplican a cada subsistema en particular y no al sistema como tal. Permitiendo así un desarrollo incremental por subsistema y facilitando el proceso de integración entre los distintos subsistemas.

DEFINR LAS PRUEBAS DE SEGURIDAD

IORIZACIÓN DE IDEAS DE PRUEBAS DE SEGURIDAD

A continuación se presentan el conjunto de ideas de pruebas de seguridad que fueron consideradas para la ejecución por cada subsistema, paralelamente se utiliza la técnica DREAD para darles una priorización de acuerdo a los cinco criterios manejados en esta técnica. Lo cual derivará en que aquellas ideas de pruebas de seguridad que obtengan un mayor grado de prioridad sean detalladas y desarrolladas como los casos de prueba de seguridad. Esto genera el documento de SISAAB-PRU-D36 PRIORIZACIÓN DE IDEAS DE PRUEBAS DE SEGURIDAD (Alvarez, et al., 2007)

Subsistema				Casos de pruebas de seguridad
Operación informática	Gestión de seguridad	Administración de usurios	Inicio de sesión	21
			Registro de usuarios	12
			Modificación de usuarios	11
			Eliminación de usuarios	9
			Control de acceso	8
		Gestión de seguridad, control y seguimiento	Generación de logs	2
	Gestión de auditoría y control	Gestión de acceso al sistema	Consulta de logs	4
	Gestión de recuperación		Administración de bases de datos	3
			Administración de aplicaciones	1
Gestión de oferta y demanda	Gestión de negociación	Registro de oferentes y demandandantes	Registro de oferentes	9
			Registro de demandantes	9
		Registro de ofertas y demandas	Registro de ofertas	7
			Registro de demandas	7
Total				103

DEFINR LAS PRUEBAS DE SEGURIDAD

IPLEMENTACIÓN DE CASOS DE PRUEBA DE ATAQUE

Título:	Titulo del caso de prueba que describe brevemente el objetivo de este				
Versión:	Versión del caso de prueba	**Id:**	Identificador del caso de prueba	**Producto:**	Producto para el cual es generado el caso de prueba
Fecha creación:	Fecha de generación de la versión	**Autor:**	Persona que genera el caso de prueba	**Subsistema:**	Subsistema al cual se aplica el caso de prueba

Requerimientos

Contenido

Descripción

Descripción detallada del objetivo del caso de prueba.

Precondiciones

Conjunto de características que debe tener o en las que debe estar el sistema antes de ejecutar la prueba

Poscondiciones

Conjunto de características en las que debe quedar el sistema una vez ha finalizado la prueba

Criterios de aceptación

Conjunto de reglas que determinaran el éxito o no de la prueba.

Herramientas necesarias

Herramienta utilizada para realizar la prueba (no aplica a todos los casos de prueba)

Pasos de la prueba

Conjunto de pasos detallados que debe seguir el "tester" durante la ejecución de la prueba

Resultados esperados

Conjunto de respuestas que se esperan por la ejecución de cada paso de la prueba.

Título:	Inicio de sesión - Inicio de sesión con datos inválidos				
Versión:	1.0	**Id:**	TCS- 01	**Producto:**	SISAAB
Fecha creación:		**Autor:**	Carlos Álvarez	**Subsistema:**	Administración de usuarios

Contenido

Descripción

El objetivo de este caso de prueba de seguridad es el de verificar las validaciones del sistema en el portlet de inicio de sesión frente al ingreso de tipos de datos inválidos (si es que se maneja algún tipo de dato) en los campos correspondientes

Precondiciones

- No se ha iniciado ninguna sesión en el sistema
- Se ha llamado a la funcionalidad de inicio de sesión

Poscondiciones

- El sistema termina en correcto funcionamiento
- El sistema no ha iniciado ninguna sesión

Criterios de aceptación

- El sistema presenta algún tipo de mensaje donde le informa al usuario que ha ingresado un tipo de dato no admitido por el sistema
- El sistema continúa en correcto funcionamiento

Herramientas necesarias

- N/A

Pasos de la prueba

1. Ingresar en el campo "Acceso" tipos de datos alfanuméricos y especiales y dejar el campo contraseña vacío.
2. Ingresar en el campo "Contraseña" tipos de datos alfanuméricos y especiales y dejar el campo "Acceso" vacío.
3. Dejar los campos de "Acceso" y "Contraseña vacíos"

Resultados esperados

1. El sistema presenta el mensaje "Por favor, introduzca una contraseña válida"
2. El sistema presenta el mensaje "Por favor, introduzca un nombre de usuario válido"
3. El sistema presenta un mensaje donde solicite el ingreso de un nombre de acceso y una contraseña válidos.

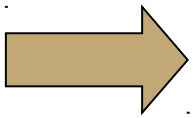
El conjunto completo de casos de prueba de seguridad se encuentran en el documento titulado SISAAB-PRU-D41 Implementación de casos de prueba de seguridad [v1.0][Iter 1][27-08-2007] [Alvarez, et al., 2007]

DEFINR LAS PRUEBAS DE SEGURIDAD

ETERMINAR LOS NIVELES DE CALIDAD DE SEGURIDAD

Criterios de selección del estándar

- Formalmente definido
- Estándar acepado internacionalmente
- Presenta resultados que reflejan claramente una medida de la calidad de la seguridad
- Congruente con el proceso de pruebas de seguridad definido
- Aplicable al SI-SAAB



Selección del estándar

El diseño de las métricas de seguridad se realiza con base a la propuesta hecha por el instituto nacional de estándares y tecnología de Estados Unidos (NIST[1]) en el estándar **NIST SP 800-26** [Swanson, et al., 2003] donde se sugieren unas medidas que pueden ser implementadas basadas en 17 tópicos sobre tecnología de información que afectan la seguridad a tener en cuenta por una organización.

NIST: National Institute of Standards and Technology

DEFINR LAS PRUEBAS DE SEGURIDAD

ETERMINAR LOS NIVELES DE CALIDAD DE SEGURIDAD

Definición de criterios a medir

Debido al gran universo de posibles métricas que se pueden tener para medir la seguridad del software y de la organización, se debe seleccionar un conjunto que se adecue a las necesidades del SI-SAAB por lo cual de los 17 tópicos planteados en NIST SP 800-26 se han seleccionado los siguientes con sus respectivas métricas

Tópicos	Número de métricas
Administración de riesgos	2
Controles de seguridad	2
Plan de seguridad del sistema	1
Personal de seguridad	2
Protección física	3
Controles de entrada y salida en producción	1
Planes de contingencia	2
Integridad de datos	2
Identificación y autorización	1
Control de acceso lógico	1
Seguimientos de auditoría	1

NIST: National Institute of Standards and Technology

DEFINR LAS PRUEBAS DE SEGURIDAD

ETERMINAR LOS NIVELES DE CALIDAD DE SEGURIDAD

Formato de definición de métricas

Para definir las métricas de seguridad se utiliza el formato definido en NIST SP 800-26 el cual estable los siguientes criterios a tener en cuenta por cada métrica.

Elemento crítico	Estado de los resultados deseados de la implementación de uno o varios sistemas y técnicas de control de seguridad que van a ser medidos por la métrica.
Pregunta subordinada	Establecer las acciones que son requeridas para obtener el objetivo presentado. Este ítem puede presentar una o más preguntas subordinadas.
Métrica	Define la métrica por la descripción de mediciones cuantitativas brindadas por la métrica. Se acostumbra a iniciar con palabras como "porcentaje", "número", "frecuencia", etc.
Propósito	Describe la funcionalidad total obteniendo la métrica. Incluye cuando una métrica deberá ser usada para una medición del rendimiento interno o un reporte externo.
Evidencia de implementación	Enumera la existencia de pruebas de los controles de seguridad que validan la implementación. Esta evidencia es utilizada para calcular la métrica, también se presentan indicadores indirectos que validan que la actividad es realizada, y factores que puedan causar resultados insatisfactorios.
Frecuencia	Propone periodos de tiempo para la recolección de la información que es usada para medir los cambios a través del tiempo.
Formula	Describe el cálculo que se debe realizar para obtener una expresión numérica de la métrica.
Fuente de datos	Lista la ubicación de la información que se usa para calcular la métrica.
Indicadores	Proporciona información acerca del significado de la métrica y su tendencia de funcionamiento. Propone posibles causas de las tendencias identificadas a través de la medición y puntualiza posibles soluciones para los resultados obtenidos.

DEFINR LAS PRUEBAS DE SEGURIDAD

DETERMINAR LOS NIVELES DE CALIDAD DE SEGURIDAD

Formato de definición de métricas

Todo el conjunto de métricas de seguridad definidas se encuentran en el documento SISAAB-PRU-D35 Métricas de calidad de seguridad [V1.0][Iter1][2007-11-06] [Alvarez, et al., 2007]

Elemento crítico	Se han identificado las operaciones más críticas y sensibles y los recursos de soporte computacional
Pregunta subordinada	Se han identificado los archivos de datos y operaciones críticos y la frecuencia de generación de "backups" está documentada
Métrica	Porcentaje de archivos de datos y operaciones críticos con una frecuencia de "backups" establecida
Propósito	Cuantificar el riesgo debido a "backups" insuficientes.
Evidencia de implementación	1. Se han identificado los datos y operaciones críticos ___ Si ___ No ____ No hay datos ni operaciones críticos 2. Si la respuesta a la pregunta 1 es no, por qué no ___ No se conocía el requerimiento ___ Falta de recursos ___ Otro 3. Número de archivos de datos y operaciones críticos identificados que requieren "backups" ____ 4. Número de archivos de datos y operaciones críticos identificados que requieren "backups" para los cuales la frecuencia de los "backups" está establecida y documentada. ____ 5. Los "backups" están documentados ____ Si ____ No 6. Los "backups" se realizan regularmente (de acuerdo a los requerimientos) ____ Si ____ No 7. Los archivos de "backup" son probados cada vez que son generados para asegurar la total transferencia y copia de datos ____ Si ____ No
Frecuencia	Anual
Formula	Número de archivos críticos con una frecuencia de "backups" establecida (pregunta 4) / Número de archivos críticos que requieren "backup" (pregunta 3)
Fuente de datos	
Indicadores	Los resultados de esta métrica se deben acercar al 100% indicando que todos los archivos que requieren "backups" están siendo respaldados de acuerdo al proceso de "backups" establecido.

EJECUCIÓN DE PRUEBAS DE SEGURIDAD

JECUCIÓN DE CASOS DE PRUEBA DE ATAQUE

Después de implementar los casos de prueba de ataque definidos se procede a ejecutarlos por cada subsistema, teniendo en cuenta las definiciones de cada caso de prueba.

Una vez ejecutado cada caso se genera un reporte con los resultados obtenidos y se reportan los defectos encontrados por medio de la herramienta testopia a los encargados de la seguridad del sistema. El resumen de este reporte se encuentra en el documento SISAAB-PRU-D37 Resultados de pruebas de seguridad [V1.0][Iter1][2007-11-15] [Alvarez, et al., 2007]

EJECUCIÓN DE PRUEBAS DE SEGURIDAD

ESULTADOS DE LA EJECUCIÓN DE LOS CASOS DE PRUEBA DE SEGURIDAD

Subsistema				Casos de pruebas de seguridad				
				No ejecutados	Cancelados	Fallidos	Existosos	Total
Operación informática	Gestión de seguridad	Administración de usurios	Inicio de sesión	0	1	10	10	21
			Registro de usuarios	2	0	9	1	12
			Modificación de usuarios	2	0	8	1	11
			Eliminación de usuarios	2	0	5	2	9
			Control de acceso	2	0	5	1	8
		Gestión de seguridad, control y seguimiento	Generación de logs	0	0	2	0	2
	Gestión de auditoría y control	Gestión de acceso al sistema	Consulta de logs	0	0	3	1	4
	Gestión de recuperación		Administración de bases de datos	0	0	1	2	3
			Administración de aplicaciones	0	0	1	0	1
Gestión de oferta y demanda	Gestión de negociación	Registro de oferentes y demandandantes	Registro de oferentes	9	0	0	0	9
			Registro de demandantes	9	0	0	0	9
		Registro de ofertas y demandas	Registro de ofertas	0	0	0	7	7
			Registro de demandas	0	0	0	7	7
Total				26	1	44	32	103

EJECUCIÓN DE PRUEBAS DE SEGURIDAD

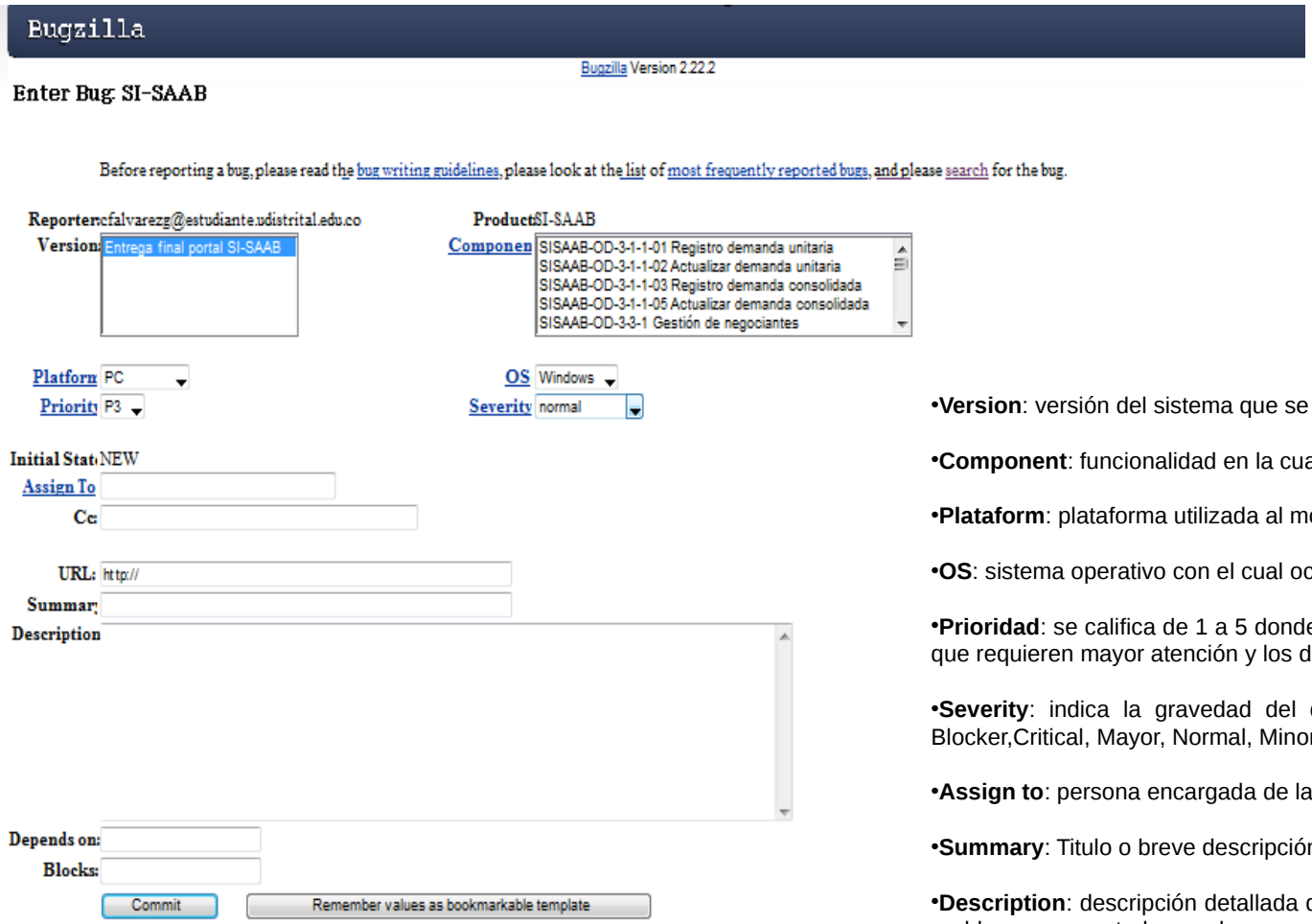
EPORTE DE DEFECTOS

Para realizar el reporte de los defectos se utiliza la metodología utilizada por el equipo de pruebas del SI-SAAB, donde el procedimiento básico en el de registrar y gestionar los defectos por medio de la herramienta Testopia, el cual está explicado más detalladamente en el documento SISAAB-PRU-D05 ADMINISTRACIÓN DE DEFECTOS CON QATRAQ Y BUGZILLA [v1.0][Iter 1][25-09-2006] (Alvarez, et al., 2006).

EJECUCIÓN DE PRUEBAS DE SEGURIDAD

EPORTE DE DEFECTOS

Formato de reporte



•**Version**: versión del sistema que se está trabajando

•**Component**: funcionalidad en la cual se encontró el defecto

•**Plataform**: plataforma utilizada al momento de encontrar el defecto

•**OS**: sistema operativo con el cual ocurrió el defecto

•**Prioridad**: se califica de 1 a 5 donde los defectos con prioridad 1 son los que requieren mayor atención y los de prioridad 5 los de menor atención

•**Severity**: indica la gravedad del defecto don se tienen los estados Blocker,Critical, Mayor, Normal, Minor, Trivial y Enhacement.

•**Assign to**: persona encargada de la solución del defecto

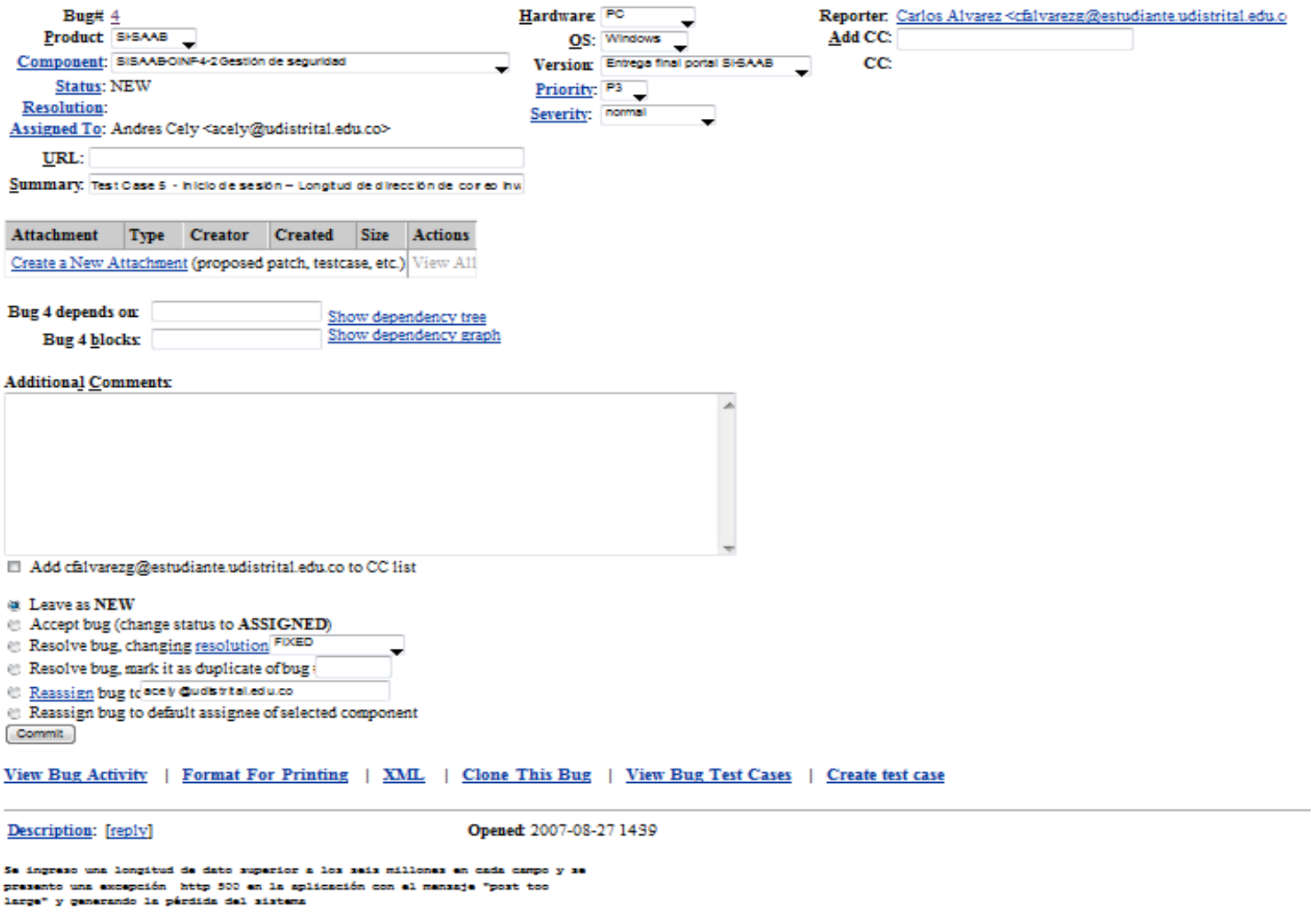
•**Summary**: Titulo o breve descripción del defecto

•**Description**: descripción detallada del defecto, explicando claramente el problema presentados y los pasos llevados a cabo para que se presentara el defecto (pasos de la prueba)

EJECUCIÓN DE PRUEBAS DE SEGURIDAD

EPORTE DE DEFECTOS

Ejemplo de reporte



Bug# 4
Product: SIHSAAB
Component: SIHSAAB:INF4-2 Gestión de seguridad
Status: NEW
Resolution:
Assigned To: Andres Cely <acely@udistrital.edu.co>
URL:
Summary: Test Case 5 - Inicio de sesión – Longitud de dirección de correo inv

Hardware: PC
OS: Windows
Version: Entrega final portal SIHSAAB
Priority: P3
Severity: normal

Reporter: Carlos Alvarez <cfalvarezg@estudiante.udistrital.edu.c
Add CC:
CC:

Attachment	Type	Creator	Created	Size	Actions
Create a New Attachment (proposed patch, testcase, etc.)					View All

Bug 4 depends on:
Bug 4 blocks:
Show dependency tree
Show dependency graph

Additional Comments:

Add cfalvarezg@estudiante.udistrital.edu.co to CC list

Leave as NEW
Accept bug (change status to ASSIGNED)
Resolve bug, changing resolution FIXED
Resolve bug, mark it as duplicate of bug
Reassign bug to acely@udistrital.edu.co
Reassign bug to default assignee of selected component
Commit

View Bug Activity | Format For Printing | XML | Clone This Bug | View Bug Test Cases | Create test case

Description: [reply]
Opened: 2007-08-27 1439

Se ingreso una longitud de dato superior a los seis millones en cada campo y se presento una excepción http 500 en la aplicación con el mensaje "post too large" y generando la pérdida del sistema

MEJORAR LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD

FINAR LA ESTRATEGIA DE PRUEBAS DE SEGURIDAD

Para la siguiente iteración de las pruebas de seguridad se sugiere que cada uno de los cuatro integrantes de pruebas sea asignado a funcionalidades específicas y que realicen las labores correspondientes al diseñador de pruebas de seguridad y al probador de seguridad para esa funcionalidad, además que el líder de pruebas asuma también el rol de líder de pruebas de seguridad y se asigne una persona que cumpla con la función de analista de pruebas de seguridad.

MEJORAR LOS ARTEFACTOS DE PRUEBAS DE SEGURIDAD

EFINIR NUEVOS ELEMENTOS DE SEGURIDAD A PROBAR

Para la siguiente iteración de las pruebas de seguridad se sugiere al líder y al analista de las pruebas de seguridad incluir los siguientes subsistemas dentro de la superficie de ataque y realizar todo el proceso correspondiente para la ejecución de esas pruebas de seguridad.

- Gestión de oferta y demanda
  - Gestión de procesos de solicitud
    - Gestionar selección de ofertas
    - Gestionar de carro de compras
    - Gestionar de documentos de soporte
- Operación informática
  - Portlet de transcripción

Así mismo se sugiere revisar nuevos cambios que se hayan dado en los subsistemas tenidos en cuenta durante esta iteración de pruebas de seguridad

CONCLUSIONES

CONCLUSIONES

NCLUSIONES SOBRE LA DEFINICIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD DE SOFTWARE

- Durante la realización de este proyecto se encontró que aunque existen ideas y tendencias para la generación de las pruebas de seguridad, no hay ninguna metodología o proceso formalmente definido que se pueda acoplar fácilmente al desarrollo de software llevado a cabo por medio de RUP.

- El proceso de pruebas de seguridad de software generado en este proyecto tiene las características de ser un proceso con un nivel de definición formal semejante al estilo manejado por RUP y que puede ser integrado fácilmente al proceso de pruebas de software definido en este.

- El proceso definido permite la generación y ejecución de pruebas de seguridad de software de una manera formal por parte de una persona sin grandes conocimientos en seguridad, el cual generalmente es el perfil de un “tester” de software.

- De acuerdo a la conclusión anterior se observa que por lo tanto el proceso general de pruebas de software (pruebas funcionales y pruebas de seguridad) puede ser llevado a cabo por el mismo equipo lo cual mantendrá una uniformidad en el proceso de pruebas.

- El flujo definido en el proceso de pruebas de seguridad de software permite llevar un orden sistémico, iterativo e incremental que permite la maduración del las pruebas de seguridad de software a medida de se avanza en el desarrollo del proceso dentro de un proyecto de software.

- Aunque el proceso definido fue generado para realizar las de pruebas de seguridad en el SI-SAAB, el grado de abstracción que se obtuvo en su definición permite que, con una definición más granular en los distintos pasos de las actividades definidas y con la definición de formatos estándar para la definición de artefactos como el plan de pruebas de seguridad, los casos de prueba de seguridad, las métricas de seguridad y los reportes de defectos, se pueda utilizar este proceso en el desarrollo de proyectos similares que sean realizados bajo la metodología definida en RUP.

CONCLUSIONES

NCLUSIONES SOBRE LA IMPLEMENTACIÓN DEL PROCESO DE PRUEBAS DE SEGURIDAD DE SOFTWARE

- Tras integrar el proceso de pruebas de seguridad de software en el SI-SAAB, se obtiene un proceso que se acopló fácilmente al proceso que ya se venía trabajando de pruebas de software.

- Debido a que el inicio de este proyecto no comenzó simultáneamente con el desarrollo del SI-SAAB, sino en etapas donde ya se encontraban desarrollos con funcionalidades implementadas y políticas y planes de seguridad totalmente definidos, las pruebas de seguridad llevadas a cabo generan cierto retroceso en el proceso ya que obligan a generar cambios de diseño en estos artefactos que afectan la seguridad y que pudieron ser abordados desde las etapas de diseño de los mismos si el proceso de pruebas de seguridad hubiera comenzado simultánea y paralelamente a las demás etapas.

- Al momento de integrar y presentar el proceso a los demás integrantes del equipo de pruebas este fue recibido sin mayores impactos siendo tomado como una iteración más del proceso de pruebas y no como un proceso totalmente aparte y ajeno al proceso que ya se venía llevando.

- Al utilizar las herramientas y formatos utilizados en las pruebas de software se obtuvo un ahorro de tiempo referente al conocimiento y familiarización con los mismos, ya que el equipo ya tenía este conocimiento de tiempo atrás.

- Debido a que para la elaboración de este proyecto solo se llevó a cabo una iteración como prueba piloto del proceso definido, se encontró una gran cantidad de defectos de seguridad, y se consiguieron casos de prueba de seguridad que no pudieron ser ejecutados debido a que se requería de cierta mejora en los otros defectos encontrados antes de poder ejecutarlos.

- No se pudo llevar a cabo las métricas de seguridad de software debido a que se requería cierta madurez en la seguridad del sistema que se obtiene con el desarrollo iterativo del proceso de pruebas de seguridad.

- Aunque el proceso fue implementado en los puntos más críticos de seguridad del sistema, en futuras iteraciones se debe ampliar el cubrimiento de este proceso a todos los subsistemas del SI-SAAB como una labor conjunta de todo el equipo de pruebas.

REFERENCIAS

1. **Alvarez, Carlos and Diosa, Henry. 2006.** *Administración de defectos con QaTraq y Bugzilla.* Bogotá : s.n., 2006. Documento técnico. SISAAB-PRU-D05.
2. **—. 2007.** *Implementación de casos de prueba de seguridad.* Bogotá : s.n., 2007. Documento técnico. SISAAB-PRU-D41.
3. **—. 2007.** *Métricas de calidad de seguridad.* Bogotá : s.n., 2007. Documento técnico. SISAAB-PRU-D35.
4. **—. 2007.** *Plan de pruebas de seguridad.* 2007. Documento técnico. SISAAB-PRU-D24.
5. **—. 2007.** *Priorización de ideas de pruebas de seguridad.* Bogotá : s.n., 2007. Documento técnico. SISAAB-PRU-D36.
6. **—. 2007.** *Resultados de pruebas de seguridad.* Bogotá : s.n., 2007. Documento técnico. SISAAB-PRU-D37.
7. **Diaz, Oscar and Diosa, Henry. 2006.** *Plan de pruebas.* s.l. : Grupo de investigación Arquisoft, 2006. Documento técnico. SISAAB-PRU-D01.
8. **Grady, Robert. 1992.** *Practical Software Metric for Project Management and Process Improvement.* s.l. : Prentice Hall, 1992. p. 50.
9. **Guerrero, Diego and Diosa, Henry. 2006.** *Diseño de despliegue del SI-SAAB.* s.l. : Convenio No 27 entre la Universidad Distrital y la UESP, 2006. Documento técnico. SISAAB-PTF-D61.
10. **Howard, Michael.** A look inside the security develpment lifecycle at Microsoft. [Online] http://msdn.microsoft.com/msdnmag/issues/05/11/SDL/default.aspx.
11. **IBM. 2003.** *Rational Unified Process.* 2003.
12. **IEEE. 1990.** *International Engineer and Electronic Standards.* 1990.

REFERENCIAS

13. **Isaza, Carlos and Diosa, Henry. 2006.** *Concepción del proyecto de desarrollo SI-SAAB.* s.l. : Grupo de investigación Arquisoft, 2006. Documento técnico. SISAAB-PLA-D01.
14. **—. 2006.** *Modelo funcional y línea base arquitectural del sistema de información para el sistema de abastecimiento de alimentos de Bogotá.* 2006. Documento técnico.
15. **—. 2006.** *Sistema de abastecimientos para Bogotá. Localidad Ciudad Bolivar.* Bogotá : Sección de publicaciones Universidad Distrital Francisco José de Caldas, 2006. Documento interno.
16. **Jürjens, Jan. 2004.** *Secure Systems Development with UML.* s.l. : Springer, 2004.
17. **Kruchten, Philippe. 2003.** *The Rational Unified Process: an introduction.* Tercera. s.l. : Addison - Wesley, 2003.
18. **Meier, J.D., Mackman, Alex, Wastell, Blaine. 2005.** *Threat modeling web applications.* s.l. : Microsoft corporation, 2005.
19. **Myers, Glenford. 2004.** *The art of software testing.* Segunda edición. s.l. : John Wiley & Sons, 2004.
20. **Ness, Pete. 2005.** *The Rational Unified Process for testers.* s.l. : IBM, 2005.
21. **Perry, William. 2000.** *Efective methods for software testing.* Segunda. s.l. : Wiley, 2000.
22. **Piattini, Mario. 2004.** *Análisis y diseño de aplicaciones informáticas de gestión.* s.l. : Alfaomega, 2004.
23. **Pressman, Roger. 2004.** *Ingeniería de software. Un enfoque práctico.* Sexta edición. s.l. : Prentice - Hall, 2004.
24. **Schumacher, Markus. 2003.** *Security Engineering with Patterns: Origins, Theoretical Models, and New Applications (Lecture Notes in Computer Science).* Primera edición. s.l. : Springer, 2003.
25. Security in the software devlopment lifecycle. [Online] http://www-128.ibm.com7developerworks/rational/library/content/RationalEdge/oct04/viega/index.html.

REFERENCIAS

26. **Shirey, R. 2000.** *RFC 2828.* [ed.] GTE / BBN Technologies. 2000.
27. **Stallings, William. 2003.** *Cryptography and Network Security.* Tercera edición. s.l. : Prentice Hall, 2003.
28. **Steel, Christopher. 2006.** *Core security patterns: best practices and strategies for J2EE, web services and identity management.* Primera edición. s.l. : Pearson, 2006.
29. **Swanson, Mariane, et al. 2003.** *Security metrics guide for information technology systems.* s.l. : NIST, 2003.
30. Testopia. [Online] http://sisaab-pru.udistrital.edu.co/testopia.
31. The trustworthy computing security development lifecycle. [Online] http://msdn.microsoft.com/security/default.aspx?pull=/library/en-us/dnsecure/html/sdl.asp.
32. **Weitzenfeld, Alfredo.** *Ingeniería de software orientada a objetos con UML, Java e Internet.* s.l. : Thomson.
33. **Wysopal, Chris, et al. 2006.** *The Art of Software Security Testing.* Primera edición. s.l. : Addison-Wesley, 2006.

GRACIAS