FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS

IMPLEMENTACIÓN DE UNA METODOLOGÍA DE PRUEBAS DE PENETRACIÓN EN EL ÁREA DE ASEGURAMIENTO DE LA CALIDAD DE AEP ENERGY

PRESENTADO POR

LAURA JANETT PAJUELO GRÁNDEZ

INFORME POR EXPERIENCIA PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE COMPUTACIÓN Y SISTEMAS

LIMA – PERÚ

2015

USMP
UNIVERSIDAD DE SAN MARTIN DE PORRES

FACULTAD DE INGENIERÍA Y ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS

IMPLEMENTACIÓN DE UNA METODOLOGÍA DE PRUEBAS DE PENETRACIÓN EN EL ÁREA DE ASEGURAMIENTO DE LA CALIDAD DE AEP ENERGY

INFORME POR EXPERIENCIA

PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE COMPUTACIÓN Y SISTEMAS

PRESENTADO POR

PAJUELO GRÁNDEZ, LAURA JANETT

LIMA – PERÚ

2015

Dedicatoria

A mis padres Jaime y Ofelia por su esfuerzo, su apoyo incondicional y confiar siempre en mí. A Roberto por ser el agente principal en el desarrollo de este trabajo.

Agradecimiento

Mi agradecimiento a la Universidad “San Martín de Porres” por brindarme las mejores herramientas, y a mis profesores por compartir sus conocimientos y experiencias profesionales. A la empresa AEP Energy por darme la oportunidad de ser parte de su equipo profesional y permitirme desarrollar este proyecto.

A mis padres y familiares ya que con su apoyo y ejemplo, me ayudan a cumplir mis metas y superarme siempre. A Roberto, mi compañero de vida y socio profesional gracias por la paciencia, por compartir sus conocimientos y crecer juntos.

ÍNDICE

Lista de tablas

Lista de figuras

Lista de anexos

RESUMEN

A inicio del año 2013, se inició en AEP Energy la implementación de controles de seguridad orientados a cumplir con la normativa SOX. Como parte de esta implementación se desarrolló el módulo de Seguridad para la administración de roles y perfiles de usuarios, y la creación de nuevas tablas de auditoría para el seguimiento y registro de transacciones en la base de datos (creación, actualización, eliminación).

Debido a esto y como parte de la propuesta de objetivos de desempeño para el año 2013, en el área de aseguramiento de la calidad se propone el proyecto de implementación de pruebas de penetración a la aplicación NextStar para agregar un nivel de calidad adicional a las actividades de pruebas que se realizan en el área. El presente documento detalla la implementación de esta metodología.

Las herramientas libres elegidas para la implementación fueron Tamper Data y Hackbar, ambos extensiones libres del navegador Firefox, que permiten simular la obtención de información oculta para un usuario corriente y la posterior elevación de privilegios a usuarios no autorizados.

La ejecución de este proyecto nos permitió detectar vulnerabilidades en la aplicación utilizada por AEP Energy para el soporte de sus operaciones de negocio. Esto ayudó a establecer mejores controles y pruebas de seguridad que permitieron evitar futuras observaciones de auditoría y cumplir con las políticas establecidas por la normativa SOX.

El proyecto hizo posible que los integrantes del área de aseguramiento de la calidad elevaran sus conocimientos y mejoren sus habilidades para el trabajo que realizan de manera cotidiana.

Palabras claves: Seguridad de información, pruebas de penetración, suplantación de identidad, calidad de software, ataque informático

ABSTRACT

At the beginning of 2013, the implementation of security controls designed to comply with SOX regulations began in AEP Energy. As part of this implementation a module for managing security roles and user profiles was developed, and were created new audit tables for monitoring and recording all transactions in the database (create, update, delete).

Because of this and as part of the performance proposed targets for the year 2013 , in the Quality Assurance area was proposed the deployment project penetration testing for NextStar application, to add an additional level of quality to the testing activities executed in the area. This document details the implementation of this methodology.

Free tools chosen for implementation were Tamper Data and Hackbar, both are Firefox browser extensions, and which simulates obtaining hidden information for a regular user and the subsequent elevation of privileges to unauthorized users.

The execution of this project allowed us to detect vulnerabilities in the application used by AEP Energy to support their business operations. This helped to establish better controls and safety tests which allowed avoid future audit observations and comply with the policies established by SOX.

The project made possible that members of quality assurance area increase their knowledge and improve their skills for the work they do on a daily basis.

Keywords: information security, penetration testing, phishing, quality software, hacking.

INTRODUCCIÓN

Esta investigación evalúa el impacto que tienen las pruebas de penetración en la aplicación NextStar de AEP Energy, cuyas actividades de negocio se desarrollan en Chicago – EEUU, y el soporte de sistemas que se brinda desde Lima – Perú, con servidores en Chicago.

En las empresas del país los profesionales del área no conocen de manera amplia lo que son las pruebas de penetración y no cuentan con la preparación necesaria para realizarlas y es por eso que el trabajo de esta investigación busca centrar los aspectos específicos para que sirvan en la implementación de una metodología adecuada.

A través de investigaciones, en toda América se ha encontrado que aún es muy poco probable que las empresas ejecuten pruebas de penetración para evaluar las fallas de seguridad, estas solo recurren a las mencionadas técnicas cuando se encuentran en un estado crítico, cuando la seguridad ya ha sido violada.

Esta investigación tiene como objetivo general: evaluar las vulnerabilidades que se presentan en la infraestructura tecnológica en la aplicación NextStar de AEP Energy y como objetivos específicos:

- Analizar los conceptos y fundamentos de seguridad en aplicaciones web.
- Determinar los tipos de vulnerabilidades en la aplicaciones web
- Determinar las herramientas existentes actualmente para detección de vulnerabilidades de penetración en aplicaciones web.
- Diseño y ejecución de pruebas de penetración en la aplicación NextStar.
- Desarrollar la propuesta metodológica para determinar la seguridad en las aplicaciones web de AEP Energy.
- Capacitación y transferencia de conocimientos a todo el equipo de aseguramiento de la calidad.

Las metodologías utilizadas en esta investigación son las siguientes: **Bibliográfica**, ya que se consulta material de libros, artículos de Internet, revistas y otro tipo de fuentes para obtener información relevante. **De campo**, porque se hace uso de herramientas y técnicas para determinar las vulnerabilidades de seguridad dentro de la infraestructura de TI. **Descriptiva**, porque se evalúan las fallas de seguridad de la infraestructura de TI de la empresa respondiendo a preguntas ¿Cómo estaba la seguridad antes de evaluar el área? ¿Cuántas áreas fueron evaluadas? y **Prospectiva**, porque se toma en cuenta un período determinado dentro del cual se evalúa la empresa.

Para la implementación y desarrollo se tomará en cuenta como universo el módulo de B2B para la evaluación del funcionamiento de envío de métodos y data para acciones de creación, actualización y eliminación de registros en la base de datos.

Las pruebas se harán utilizando la metodología de caja blanca, la cual consiste en evaluar la seguridad de la infraestructura tecnológica de la empresa con conocimiento general del entorno, y de las funcionalidades de la aplicación y del negocio. Este hace más referencia a lo que podría suponer un ataque interno.

El primer capítulo muestra la trayectoria profesional del autor del informe, revisando cronológicamente su experiencia y logros obtenidos durante este tiempo.

En el segundo capítulo se presenta una breve descripción de la empresa AEP Energy, donde se implementa el proyecto y cómo es su arquitectura.

El tercer capítulo detalla el desarrollo e implementación del proyecto y la ejecución de las pruebas de penetración en el módulo B2B de la aplicación NextStar, que son realmente las pruebas, la investigación requerida, el diseño de los casos de prueba y escenarios, las técnicas y herramientas

que se utilizan para realizar las pruebas de penetración, los pasos que hay que realizar con cada una de ellas para encontrar vulnerabilidades en la aplicación y la obtención de resultados.

Finalmente, en el cuarto capítulo se da a conocer la relevancia de la experiencia adquirida con el desarrollo de este proyecto, así como el impacto que tuvo sobre la organización la presentación de los resultados obtenidos.

CAPÍTULO I

TRAYECTORIA PROFESIONAL

Agosto 2010 – Febrero 2012
Empresa Editora El Comercio
Analista de Calidad de Software

Laura Pajuelo inició su experiencia como Analista de Calidad de Software en la Empresa Editora El Comercio donde se desempeñó como Analista Junior en el área de Plataforma Digital en los proyectos de implementación de las primeras versiones de Portales Verticales y Clasificados:

- Clasificados:
  - Urbania - www.urbania.pe
  - Neoauto – www.neoauto.pe
  - Kotear – www.kotear.pe
  - Nuestro Mercado – www.nuestromercado.pe
  - Aptitus. – www.aptitus.pe
- Tienda virtuales:
  - iQuiero – www.iquiero.pe (ahora: www.estilomio.pe)
  - Club Suscriptores – www.clubsuscriptores.pe
- Pasarela de pagos:
  - Pago efectivo: www.pagoefectivo.pe

Las principales actividades del rol, estaban orientadas a la preparación de los planes de prueba, el diseño y elaboración de los casos y escenarios de prueba, ejecución de las pruebas diseñadas a nivel funcional y de base de datos.

Cuando ella empezó sus actividades, el área de aseguramiento de la calidad aún estaba en sus inicios, por lo que fue una buena oportunidad para que propusiera y asumiera el liderazgo del proyecto

de “Metodología: Implementación del ciclo de vida de desarrollo de Software”. Las actividades del proyecto fueron:

- Definición de flujos, roles, actividades del ciclo de vida de desarrollo.
- Implementación y preparación de la herramienta Redmine para el despliegue de la metodología.
- Ejecución de plan piloto.
- Preparación de reportes de resultados.
- Capacitación a todos los involucrados.
- Puesta en producción.

El proyecto fue puesto en producción en diciembre 2011 y quedó institucionalizado por la gerencia de plataforma digital para su uso en todos los proyectos.

Marzo 2012 – A la fecha
Bluestar Energy
Quality Assurance Engineer II

El ingreso a AEP Energy (Bluestar Energy en Perú) - como **Quality Assurance I** - se produce en el momento en que la empresa se encontraba en proceso del proyecto de Refactoring, dado por la fusión entre las empresas Norteamericanas: Bluestar Energy y AEP Energy. Esto requirió su participación en el proyecto de documentación del módulo Billing, así como de la validación funcional de nuevos requerimientos y de la migración de data de clientes.

Posteriormente, le fue asignado el módulo B2B, en el que se propuso el proyecto de regularización de la documentación, mapeo de transacciones y automatización de las pruebas de funcionalidades web. Este proyecto requirió un proceso de auto-aprendizaje de la herramienta Selenium.

La experiencia en el módulo B2B le permitió conocer la relación que establecen todos los módulos entre sí, y conocer a mejor detalle cómo se establecen las comunicaciones y transacciones a nivel de background. Una vez finalizado el proyecto de documentación, se le asignó en paralelo una

participación en el módulo de provisioning para el proyecto eSignature en octubre 2012, llegando a ser un recurso importante en la implementación de un nuevo flujo de negocio participando activamente en la definición de requerimientos y comunicación con los usuarios – ubicados en Chicago – para esta definición y las pruebas de aceptación de usuario. El proyecto eSignature fue puesto en producción en febrero 2013.

Los resultados exitosos obtenidos en sus proyectos 2012, dieron lugar a que obtuviera un ascenso significativo en abril del 2013 a **Quality Assurance II.** En esta nueva etapa su asignación es cambiada al módulo Billing, uno de los módulos más complejos de la aplicación NextStar y que requiere un mayor nivel de abstracción dado el tipo de transacciones que en este se ejecutan (facturaciones, pagos).

Además, en abril 2013 se implementa una nueva forma de evaluación anual de desempeño para todos los integrantes de la organización, siendo necesario ahora que cada persona defina sus objetivos anuales y los exponga ante la gerencia para su pre-aprobación. El cumplimiento de estos objetivos será la base para la calificación del desempeño 2014.

Es en este contexto, que Laura Pajuelo propuso los proyectos:

- Implementación de una metodología de pruebas de penetración en el área de aseguramiento de la calidad de AEP Energy.
- Implementación de una metodología de pruebas Cross Browser en el área de aseguramiento de la calidad de AEP Energy.

Ambos proyectos fueron desarrollados durante el año 2013 e incluían actividades de investigación, desarrollo, implementación y capacitación. Las capacitaciones al equipo fueron realizadas en los meses de enero y febrero, a través de reuniones presenciales, grabación de material audiovisual y la realización de evaluaciones prácticas.

La experiencia en Bluestar Energy durante el año 2013 ha sido la más significativa a lo largo de su carrera, los proyectos realizados han demandado una gran inversión de tiempo de auto-aprendizaje que le han permitido ganar mayor conocimiento empírico que ha aumentado su nivel y calidad como profesional. Sus proyectos han permitido a la organización el descubrimiento de vulnerabilidades en sus sistemas, que al ser solucionados permitieron asegurar en un mayor porcentaje de la seguridad, confidencialidad e integridad de su información y transacciones.

Es por esta razón que sus proyectos han sido bien calificados por la gerencia y la institucionalización de la metodología se encuentra en proceso para ser ejecutado por los ingenieros de calidad de todos los módulos.

CAPÍTULO II

CONTEXTO EN EL QUE SE DESARROLLÓ LA EXPERIENCIA

AEP Energy, con sede en Chicago - EEUU, proporciona suministro eléctrico competitivo para los clientes minoristas en Ohio, Illinois, Pensilvania, Nueva Jersey, Maryland, Delaware y el Distrito de Columbia. La compañía también ofrece soluciones de energía, incluyendo la respuesta de la demanda y servicios de eficiencia energética, en todos los Estados Unidos. AEP Energía ha estado en operación desde 2002 y actualmente sirve a más de 150,000 clientes.

El año 2002 AEP y AEP Retail Energy compra BlueStar Energy Holdings Inc. y sus proveedores independientes de electricidad minorista BlueStar Energy Solutions, con sede en Chicago. Logrando fusionar así a dos importantes empresa en el campo del retail de energía. En ese momento, BlueStar Energy contaba con la aplicación propia NextStar, desarrollada in house. Parte del proceso de compra incluyó el proyecto de Refactoring que entre sus principales actividades tenía la de la migración de la información de la aplicación que usaba antes AEP a través de un proveedor. El proyecto duró 8 meses.

La visión de AEP es mantenerse en el top de empresas de retail de energía y trabajan constantemente en brindar el mejor servicio a sus clientes. Además, recientemente la empresa comenzó a cotizar en la bolsa de valores, lo cual incrementa su posición en el mercado.

El equipo de tecnologías de información de AEP Energy se encuentra distribuido de la siguiente forma:

AEP Energy cuenta con un equipo de TI experto en desarrollo e implementación de sistemas web, conformado por las áreas: arquitectura, soporte & infraestructura, desarrollo y aseguramiento de la calidad.

Las áreas de Arquitectura, Soporte & Infraestructura trabajan conjuntamente para brindar las mejores soluciones de hardware para el soporte de la aplicación.

Las responsabilidades del área de aseguramiento de la calidad en la empresa AEP Energy, incluyen actividades de diseño y ejecución de pruebas funcionales y de base de datos de la aplicación que soporta todos los procesos y flujos de negocio.

CAPÍTULO III

ACTIVIDADES DESARROLLADAS

3.1. Formulación del problema

Durante años las aplicaciones web han sido vulneradas, permitiendo que agentes no autorizados (externos e internos a las organizaciones) tengan accesos a información confidencial y a su manipulación y consiguiente pérdida o utilizada para fines para los que no fue creado.

Con el auge del Cloud Computing y de la implementación de granjas de servidores alojados a miles de kilómetros de los equipos usuarios, la información de las organizaciones se encuentra cada vez más expuesta y vulnerable. De manera que, contar con una aplicación web con altos estándares de seguridad, es vital para las operaciones del negocio.

El problema es que, tal y como ocurría hace unos pocos años con las pruebas de control de calidad de software, empresas como AEP no cuentan aún con una estrategia formalmente implementada y documentada sobre la seguridad informática. Leyes como SOX obligan a empresas como AEP a cumplir con procesos de auditoría, al ser una empresa que mueve millones de dólares en el negocio, AEP está obligada a asegurar que sus procesos son transparentes y conservan la integridad de la información a través de la implementación de, por ejemplo, tablas de auditoría en sus Bases de Datos, que aseguren que toda transacción realizada en la aplicación sea ejecutada por roles debidamente definidos y autorizados y que todo movimiento sea perfectamente rastreado.

Los niveles de incidentes que afectan la seguridad de las empresas en los últimos tres años han aumentado. Solamente en el caso del Acceso Indebido a la Información se había notado un decrecimiento leve durante el 2011. Pero para el 2012, el porcentaje de víctimas de este tipo de incidentes se incrementó nuevamente.

Los atacantes pueden, potencialmente, usar muchas diferentes rutas a través de su aplicación para causar daño a un negocio u organización. Cada una de estas rutas representa un riesgo que puede, o no, ser lo suficientemente serio como para merecer atención.

A veces, estas rutas son triviales de encontrar y explotar y a veces son extremadamente difíciles. De manera similar, el daño causado puede ir de ninguno hasta comprometer toda la organización.

Tomando la base expuesta se plantea el siguiente problema: "¿Se puede desarrollar una propuesta metodológica para determinar el nivel de seguridad informática en AEP Energy, que sirva como herramienta para adoptar mecanismos de protección como parte de una estrategia de seguridad?"

3.2. Proyecto de solución

El Proyecto busca como resultado:

- Evaluar el impacto de las pruebas de penetración (Pentest) referidas a la "Referencia insegura a objetos" en la aplicación NextStar de AEP Energy.
- Obtener una metodología de pruebas de penetración referidas a la "Referencia insegura a objetos" para el área de aseguramiento de calidad.
- Capacitar al equipo de aseguramiento de la calidad en la aplicación de la metodología de pruebas de penetración.

3.3. Objetivo general y específicos

3.3.1. Objetivo general

Desarrollar una propuesta metodológica de ejecución de pruebas de penetración para determinar la seguridad en la aplicación NextStar de AEP Energy.

3.3.2. Objetivos específicos

- Analizar los conceptos y fundamentos de seguridad en aplicaciones web.
- Determinar los tipos de vulnerabilidades en las aplicaciones web.
- Determinar las herramientas existentes actualmente para detección de vulnerabilidades de penetración en aplicaciones web.
- Diseño y ejecución de pruebas de penetración en la aplicación NextStar.
- Desarrollar la propuesta metodológica para determinar la seguridad en las aplicaciones web de AEP Energy.
- Capacitación y transferencia de conocimientos a todo el equipo de aseguramiento de la calidad.

3.4. Etapas del proyecto

El proyecto fue desarrollado en 3 etapas: análisis, diseño, ejecución y resultados. A continuación se presentan las actividades desarrolladas durante cada etapa.

3.4.1. Análisis

3.4.1.1. Base teórica

a) Seguridad informática

Es posible enunciar que seguridad es el conjunto de recursos (metodologías, documentos, programas y dispositivos físicos) encaminados a lograr que los recursos de cómputo disponibles en un ambiente dado, sean accedidos única y exclusivamente por quienes tienen la autorización para hacerlo.

Eugene Spafford afirmó que un sistema es seguro si se comporta como los usuarios esperan que lo haga.

b) Propiedades de la información que protegen la seguridad informática.

Según la Norma Técnica Peruana ISO/IEC 27001 (2013), la Seguridad Informática debe vigilar principalmente por las siguientes propiedades:

- **Confidencialidad:** La información debe ser vista y manipulada únicamente por quienes tienen el derecho o la autoridad de hacerlo. Un ejemplo de ataque a la Privacidad es la divulgación de Información Confidencial.

- **Integridad:** La información debe ser consistente, fiable y no propensa a alteraciones no deseadas. Un ejemplo de ataque a la Integridad es la modificación no autorizada de saldos en un sistema bancario o de calificaciones en un sistema escolar.

- **Disponibilidad:** La información debe estar en el momento que el usuario requiera de ella. Un ataque a la disponibilidad es la negación de servicio (En Inglés Denial of Service o DoS).

- **Autenticación o autentificación:** Es la propiedad que permite identificar el generador de la información. Por ejemplo al recibir un mensaje de alguien, estar seguro que es de ese alguien el que lo ha enviado, y no una tercera persona haciéndose pasar por la otra (suplantación de identidad).

c) Seguridad en aplicaciones web – OWASP

El Proyecto Abierto de Seguridad de Aplicaciones Web (OWASP), es una comunidad de nivel mundial abierto y libre, enfocado en mejorar la seguridad en las aplicaciones de software.

La misión de OWASP es hacer la seguridad en aplicaciones "visible", de manera que las organizaciones pueden hacer decisiones informadas sobre los riesgos en la seguridad de aplicaciones.

d) Vulnerabilidades en la Web – Proyecto TOP TEN 2010

Dentro de los proyectos de OWASP se encuentra el TOP 10 de vulnerabilidades que proporciona un documento de sensibilización de gran alcance para la seguridad de aplicaciones web. La OWASP Top Ten representa un amplio consenso acerca de cuáles son los más importantes fallos de seguridad de aplicaciones web. Los miembros del proyecto incluyen una variedad de expertos en seguridad de todo el mundo que han compartido su experiencia para producir esta lista. Actualmente se encuentra en la versión 2010 que fueron traducidos al inglés, francés, idiomas español, japonés, coreano y turco, entre otros.

OWASP Top 10 (2010) se enfoca en las siguientes vulnerabilidades:

- **Inyección SQL:** Las fallas de inyección, tales como SQL, OS, y LDAP, ocurren cuando datos no confiables son enviados a un intérprete como parte de un comando o consulta. Los datos hostiles del atacante pueden engañar al intérprete en ejecutar comandos no intencionados o acceder datos no autorizados.

- **Secuencia de comandos en sitios cruzados (XSS):** Las fallas XSS ocurren cada vez que una aplicación toma datos no confiables y los envía al navegador web sin una validación y codificación apropiada. XSS permite a los atacantes ejecutar secuencia de comandos en el navegador de la víctima los cuales pueden secuestrar las sesiones de usuario, destruir sitios web, o dirigir al usuario hacia un sitio malicioso.

- **Pérdida de Autenticación y Gestión de Sesiones:** Las funciones de la aplicación relacionadas a autenticación y gestión de sesiones son frecuentemente implementadas incorrectamente, permitiendo a los atacantes comprometer contraseñas, llaves, token de sesiones, o explotar otras fallas de implementación para asumir la identidad de otros usuarios.

- **Referencia Directa Insegura a Objetos:** Una referencia directa a objetos ocurre cuando un desarrollador expone una referencia a un objeto de implementación interno, tal como un fichero, directorio, o base de datos. Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder a datos no autorizados.

- **Falsificación de Peticiones en Sitios Cruzados (CSRF):** Un ataque CSRF obliga al navegador de una víctima autenticada a enviar una petición HTTP falsificado, incluyendo la sesión del usuario y cualquier otra información de autenticación incluida automáticamente, a una aplicación web vulnerable. Esto permite al atacante forzar al navegador de la víctima para generar pedidos que la aplicación vulnerable piensa son peticiones legítimas provenientes de la víctima.

- **Defectuosa configuración de seguridad:** Una buena seguridad requiere tener definida e implementada una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor web, base de datos, y plataforma.

  Todas estas configuraciones deben ser definidas, implementadas, y mantenidas ya que por lo general no son seguras por defecto. Esto incluye mantener todo el software actualizado, incluidas las librerías de código utilizadas por la aplicación.

- **Almacenamiento Criptográfico Inseguro:** Muchas aplicaciones web no protegen adecuadamente los datos sensibles, tales como tarjetas de crédito, y credenciales de autenticación con mecanismos de cifrado o hashing. Atacantes pueden modificar o robar tales datos protegidos inadecuadamente para conducir robos de identidad, fraudes de tarjeta de crédito u otros crímenes.

- **Falla de Restricción de Acceso a URL:** Muchas aplicaciones web verifican los privilegios de acceso a URLs antes de generar enlaces o botones protegidos. Sin embargo, las aplicaciones necesitan realizar controles similares cada vez que estas páginas son accedidas, o los atacantes podrán falsificar URLs para acceder a estas páginas igualmente.

- **Protección:** Las aplicaciones frecuentemente fallan al autenticar, cifrar, y proteger la confidencialidad e Insuficiente en la integridad de tráfico de red sensible. Cuando esto ocurre, es debido a la utilización de algoritmos capa de transporte débiles, certificados expirados, inválidos, o sencillamente no utilizados correctamente.

- **Redirecciones y reenvíos no validados:** Las aplicaciones web frecuentemente redirigen y reenvían a los usuarios hacia otras páginas o sitios web, y utilizan datos no confiables para determinar la página de destino. Sin una validación apropiada, los atacantes pueden redirigir a las víctimas hacia sitios de phishing o malware, o utilizar reenvíos para acceder páginas no autorizadas.

e) Pruebas de penetración

Con los grandes fallos de seguridad que se presentan constantemente, las empresas han tenido que recurrir a diferentes métodos y pruebas para proteger su infraestructura tecnológica de atacantes que buscan tener acceso a los sistemas, ya sea para afectar el funcionamiento de estos, robar o modificar información de importancia para la empresa. Estas pruebas llevan a las empresas a disminuir los riesgos de seguridad y tener un entorno más seguro dentro de la infraestructura informática.

Las pruebas de penetración se definen como un procedimiento que involucra métodos y técnicas para inspeccionar la infraestructura tecnológica de una institución en busca de fallas de seguridad que afectan la integridad, confidencialidad y disponibilidad de los datos.

Las pruebas de penetración no son técnicas ilegales, ya que antes de realizar estas pruebas se solicita una autorización a la gerencia de la institución donde se realizan, haciendo que todo se conlleve dentro de un ámbito profesional.

Las pruebas de penetración son muy distintas de las pruebas funcionales tradicionales; no sólo carecen los evaluadores de penetración de la documentación apropiada sino que, además, éstos deben pensar como usuarios que tienen la intención de hacer daño.

Cuando se realizan pruebas de penetración el principal objetivo es encontrar vulnerabilidades en la infraestructura tecnológica de la organización, dentro de las cuales podemos destacar las más comunes:

- **Identificación de vulnerabilidades de alto riesgo:** este tipo de vulnerabilidad ponen en riesgo la infraestructura tecnológica de la institución.

- **Identificación de vulnerabilidades de bajo riesgo:** este tipo de vulnerabilidad contienen informaciones de menor riesgo, pero aun así son importantes.

- **Determinar los puntos clave donde se pueden realizar diferentes tipos de ataques:** esto verifica todas las áreas dentro de la infraestructura de red de la institución donde pueda haber un acceso, así como los routers y puntos de acceso que sirven como medio para la conexión de dispositivos.

- **Evaluar el impacto de los ataques exitosos:** también se debe evaluar que tan exitoso fue un ataque determinado en los diferentes puntos clave para solucionar las fallas de seguridad.

f) Tipos de pruebas de penetración

Los procedimientos que se emplean en estos dos tipos de metodologías dotan de información necesaria a quien los requiere sobre el estado de la infraestructura, también cabe destacar que cada uno de ellos tiene sus ventajas y desventajas en cuanto a su uso, así como también que están destinados a detectar vulnerabilidades específicas y depende de lo que se quiera lograr. Los hackers éticos que utilizan estas metodologías con propósitos benignos tardan años en aprender y adquirir toda la experiencia necesaria para brindar sus servicios así como redactar documentos que sirven de fuentes para futuras investigaciones y orientaciones.

Prueba de caja negra: Se conoce como Caja Negra o Black Box a la metodología de prueba que se realiza desde el exterior de la institución sin tener ningún tipo de conocimiento de la infraestructura que se va a evaluar, haciendo una simulación de un ataque real hacia el sistema, antes de comenzar las pruebas, las personas que practican esta metodología primero deben detectar la localización y la extensión del sistema a analizar, esta es importante porque determina el impacto que tendría un ataque desde el exterior. Las pruebas de penetración de caja negra son actividades intensas que requieren de experiencia para reducir al mínimo el riesgo que podrían correr los sistemas de que se analizan.

Prueba de caja blanca: Se conoce como Caja Blanca o White Box a la metodología de prueba que se realiza cuando se tiene conocimiento total de la infraestructura tecnológica con la que se está trabajando a diferencia de la metodología de caja negra. El objetivo de esta es comprobar errores de código y verificar configuraciones de software y hardware, centrándose más en los procesos principales del sistema.

3.4.2. Diseño de la metodología de pruebas de penetración

A continuación se describe la etapa de diseño de la metodología de pruebas de penetración para AEP Energy.

Basados en nuestra investigación teórica – presentada en la etapa anterior – basado en las pautas y recomendaciones que nos da la guía OWASP, en el

siguiente punto se describirán las etapas a seguir en la metodología, las herramientas a utilizar, así como cuáles y como serán los documentos de salida generados por la metodología: Casos de Prueba, Informe de resultados.

3.4.2.1. Definición de etapas en la metodología de pruebas de penetración

Es preciso que al momento de hacer las pruebas, el especialista realice una lista de los requerimientos para hacer las evaluaciones de seguridad en la infraestructura de red de la empresa.

Dentro de los pasos o procedimientos que hay que realizar para obtener los resultados esperados de las pruebas, se pueden destacar: Recolección de información, detección de vulnerabilidades y escalamiento de privilegios. Cada uno de estos procedimientos se realiza con un objetivo común, y con herramientas específicas para cada etapa.

Cada uno de estos niveles posee vital importancia para lograr los objetivos que se plantean al inicio de las pruebas, trabajan de forma secuencial, por consiguiente el nivel de experiencia y profesionalismo que se requiere en el área debe ser alto. Cuando una prueba usando la técnica de Caja Negra se ejecuta hay que tener en cuenta, primero que la empresa a la cual se le realiza corre riesgos con sus equipos de hardware, ya que esta trabaja de forma paralela, es decir, mientras se realiza la evaluación la empresa opera y realiza sus funciones normales bajo este ámbito de riesgos.

a) Recolección de Información

Siendo la primera etapa del proceso de evaluación de seguridad, básicamente consiste en recolectar la mayor cantidad de información sobre el objetivo a ser evaluado, esta etapa busca contar con todos los detalles necesarios e información que ayuden a facilitar al hacker ético conocer el entorno en el que trabajará, sin revelar la presencia de este ni las intenciones, analizando así la forma en que la institución opera y determinando la mejor

ruta para hacerlo. La recolección de información requiere de paciencia, mucha investigación y sobre todo pensar como el atacante.
La mayoría de los profesionales sabe que los detalles de la investigación o recolección pueden significar la diferencia entre el éxito y el fracaso de la prueba de penetración. La recolección de información se conoce como la etapa de mayor importancia ya que prevé todas las bases para la continuidad de los demás niveles. La inteligencia abierta es una forma utilizada para recabar, seleccionar información disponible, también existe la forma pasiva que permite identificar los límites de una red, los sistemas operativos y los servidores web en dicho objetivo.

b) Detección de vulnerabilidades

Determinar la vulnerabilidad que existe en los sistemas es el siguiente paso luego de reunir toda la información relevante sobre el objetivo. Con este propósito un hacker ético debe tener a su disposición un conjunto de vulnerabilidades, el conocimiento del profesional es parte vital también en este proceso ya que pone a prueba toda su experiencia, se realiza un análisis de toda la información recolectada para determinar la vulnerabilidad lo cual se le llama escaneo manual de vulnerabilidad mientras que la detección de vulnerabilidad se hace manualmente.

Cuando se finaliza la detección de vulnerabilidades se produce una lista definitiva de brechas sobre el objetivo para investigar a profundidad, esta lista se utiliza en la siguiente etapa. Luego se realiza un intento de intrusión en estas brechas las cuales ya fueron definidas previamente.

En este punto se identificarán

b.1.- Los riesgos potenciales:

- Posibles agentes atacantes:
    - Humano sin intención (accidente, descuido, etc.)
    - Humano intencionado (persona interna, externa, outsourcing, etc.)

- Tipo de ataque utilizado:
    - Predicción / Deducción de credenciales / sesión
    - Intentos de acceso no autorizado
    - Intersección de la información
- Impacto en el negocio de un ataque exitoso.

b.2.- Factores determinantes de la probabilidad de ocurrencia: Se requiere determinar la probabilidad de que el riesgo potencial detectado sea utilizado por un atacante.

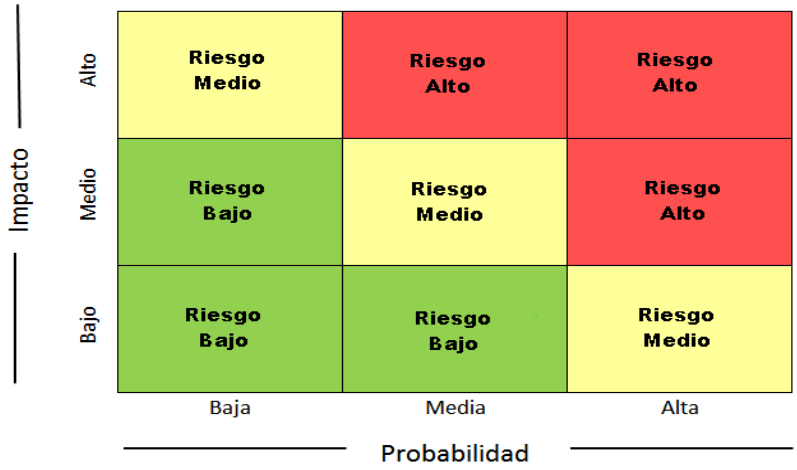
- Factores de los agentes atacantes
    - Nivel técnico
    - Motivación
    - Oportunidad de encontrar y explotar la vulnerabilidad.
    - Tamaño/Tipo de atacantes
- Factores de vulnerabilidad/riesgo
    - Facilidad de descubrimiento
    - Facilidad de explotación
    - Conocimiento previo de la vulnerabilidad/riesgo
    - Detección de la intrusión/explotación

b.3.- Factores determinantes del impacto: Un riesgo puede tener un impacto técnico (sistemas, datos almacenados, etc…) y un impacto sobre el negocio (operativa, lógica de la aplicación).

- Factores de impacto técnico
    - Pérdida de confidencialidad
    - Pérdida de integridad
    - Pérdida de disponibilidad
    - Pérdida de rastro
- Factores de impacto en el negocio
    - Repercusión financiera
    - Repercusión reputacional
    - Violación de la privacidad (cantidad de personas afectada)

b.4.- Determinar la severidad del riesgo: a partir de la valoración de los puntos anteriores se obtiene la severidad del riesgo, utilizando la siguiente matriz:

Tabla 1: Matriz de obtención del riesgo



Fuente: Propia

c) Escalamiento de privilegios

Esta etapa se enfoca en hacer pruebas de seguridad con la información previamente recolectada para la explotación de las vulnerabilidades encontradas en la infraestructura de red. Se utilizan todos los conocimientos que se tenga sobre los sistemas para probar todas las alternativas necesarias que se puedan, para introducirse al sistema. En esta es donde se refleja la profesionalidad y el conocimiento que posee el especialista que realiza las pruebas, la organización donde se realizan estas pruebas debe de estar informada de cualquier actividad que pueda ocasionar problemas al funcionamiento del sistema, de modo que pueda planificarse en una fecha donde se realicen sin inconvenientes.

Cuando se trata de acceder al sistema esto nunca se logra a la primera, primero hay que hacer explotar varias vulnerabilidades del sistema para conseguir un acceso de diferentes niveles en la plataforma que se esté trabajando, no suele pasar de inmediato, hay que hacer uso de varias técnicas y herramientas para conseguir un acceso total y hacerse del control

del sistemas. Esta tiene como objetivo comprobar si un atacante puede tener privilegios en el sistema para obtener datos que estén restringidos y ocasionar daños a la infraestructura del sistema. En esta etapa es donde se requiere el uso de técnicas de programación o de explotación de los niveles de seguridad que se encuentren en el sistema.

3.4.2.2. Definición de herramientas para pruebas de penetración

a) Tamper Data

Tamper data es una extensión de Firefox que permite visualizar, grabar e incluso modificar las solicitudes HTTP salientes. Esto es muy útil cuando se trata de responder a preguntas como:

- ¿Se envían cookies al navegador?
- ¿Las cookies están marcadas como "seguras"?
- ¿Qué tipo de autentificación HTTP está ocurriendo?

Tamper Data puede ayudar a responder a cada una de estas y otras preguntas desconcertantes del comportamiento del sitio web.

Primeros pasos

1. Ya que es una extensión de Firefox, primero deberá descargar e instalar Firefox.
2. A continuación, visite la página del proyecto Tamper Data (https://addons.mozilla.org/firefox/966/) y haga clic en el enlace que dice "Instalar ahora".
3. Por último, reinicie Firefox y abra herramientas → Tamper Data. Con ello se abre el "Tamper Data" de la ventana.

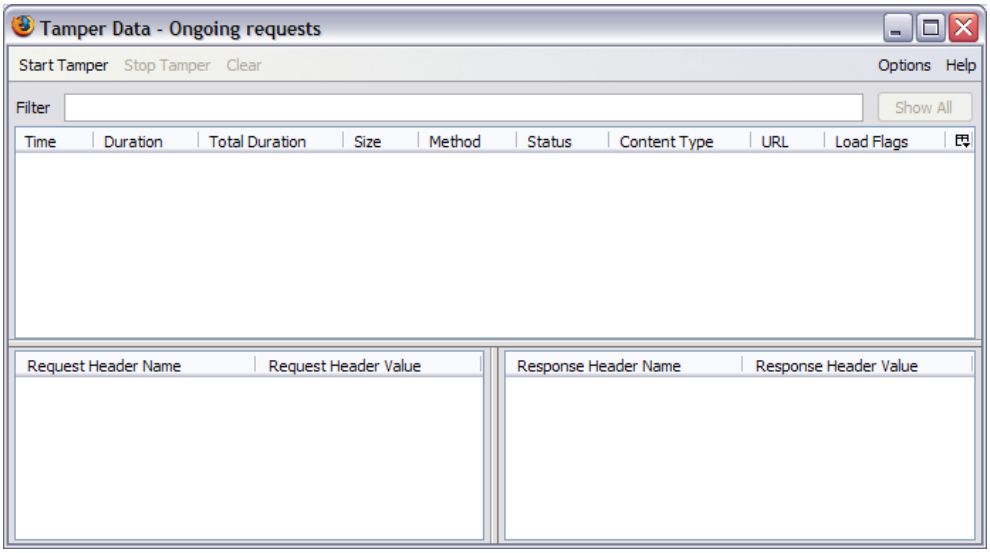


Figura 1. Interfaz de Tamper Data **Fuente:** Tamper Data

4. Grabación de Transacciones: Tan pronto como la ventana de solicitudes en curso termine, Tamper Data empezará a grabar las peticiones HTTP. Este es el aspecto de la ventana después de solicitar blogger.com.

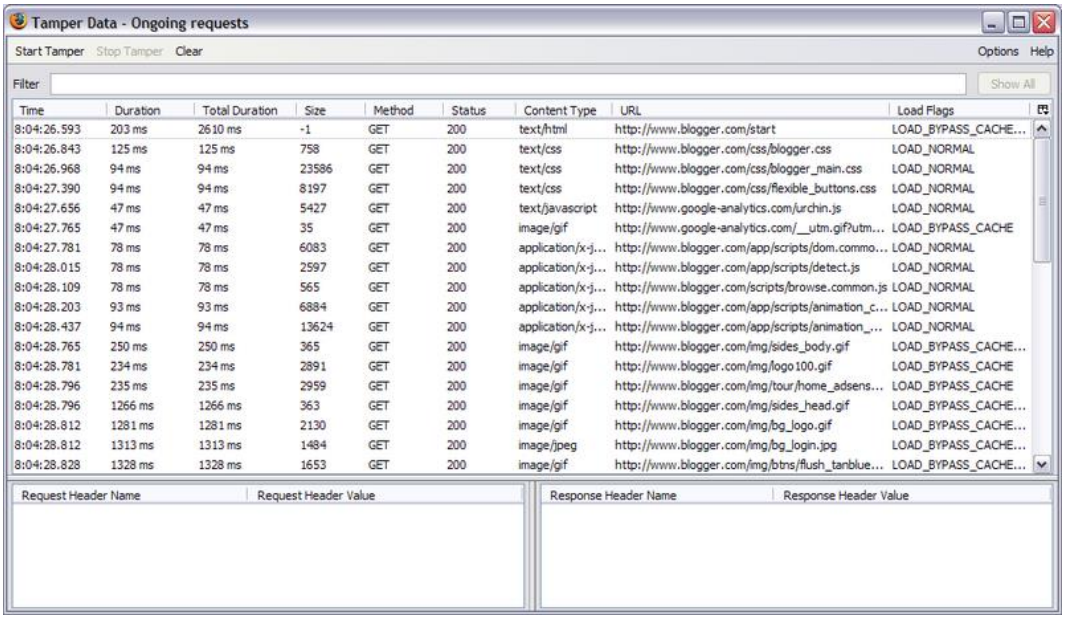


Figura 2. Grabación de transacciones (solicitudes en curso) **Fuente:** Tamper Data

Las columnas en el panel de la ventana principal son:

- Tiempo - Cuando ocurrió la solicitud.
- Duración - Cuánto tiempo se tardó en responder.
- Duración Total - Cuánto tiempo le tomó en responder (incluye la respuesta el tiempo de descarga del tema y todos los subtemas).
- Tamaño - Tamaño de contenido recibido (-1 indica que el elemento se ha cargado desde la memoria caché)
- Method - Método HTTP emitida (GET o POST)
- Estado - Código de estado HTTP recibida o " Cargado de caché "
- Tipo de Contenido - Tipo de datos recibidos (también conocido como Mime - Type)
- URL - URL completa de la solicitud.
- Cargar Banderas - Información HTTP adicional utilizado en la recuperación o de la respuesta de contenido.

5. Al seleccionar un elemento HTTP se muestra la información de respuesta en los dos paneles inferiores izquierdo y derecho respectivamente.

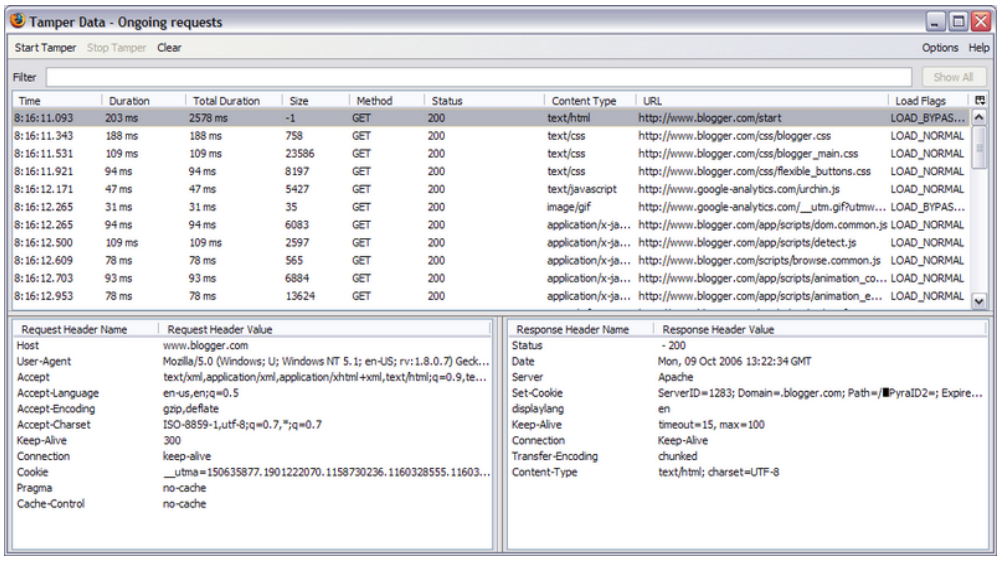
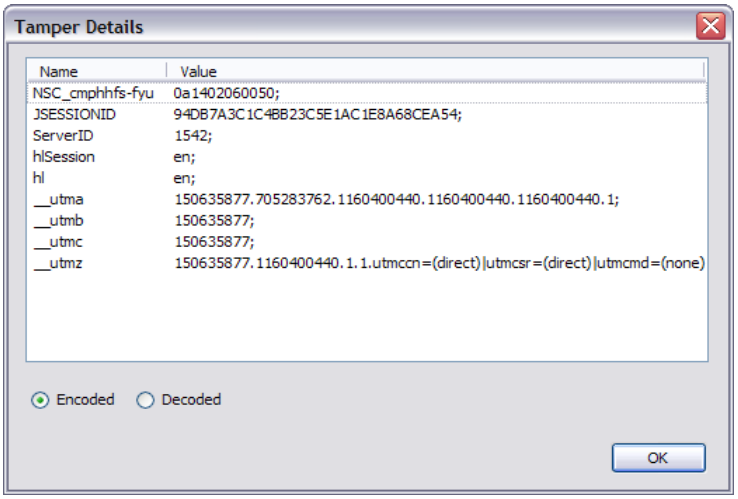


Figura 3. Información de una respuesta en Tamper Data **Fuente:** Tamper Data

Esto le da una visión más detallada de lo que está haciendo la solicitud. Si la solicitud que se ha seleccionado contiene la información de

cookies, se verá una línea cookie en el panel de la izquierda o una línea de Set-Cookie en el panel de la derecha o ambas.

Hacer doble clic en una entrada hará que aparezca el detalle de la ventana, lo que facilita el acceso a los datos de ese elemento. Aquí, se ingresó a los detalles para la cabecera Cookie de la solicitud inicial de la página principal blogger.com.



F**igura 4.** Información de un elemento **Fuente:** Tamper Data

Usando el proceso descrito anteriormente, es fácil de inspeccionar lo que está pasando durante una sesión de navegación.

Aunque son bastantes los datos dentro de Tamper Data, a menudo es conveniente mover esos datos en un archivo externo para su visualización. Para ello, vuelva a la ventana de solicitudes en curso, haga clic derecho y elegir la opción " Copiar todo".

Esto hará que toda la información de la solicitud en el portapapeles para que pueda pegarlo en su editor de texto favorito.

6. Resultados gráficos

Para representar gráficamente los resultados registrados, en la ventana de solicitudes en curso, seleccione los resultados deseados, haga clic derecho y elegir la opción "Graph seleccionado" o "Gráfico de todo".

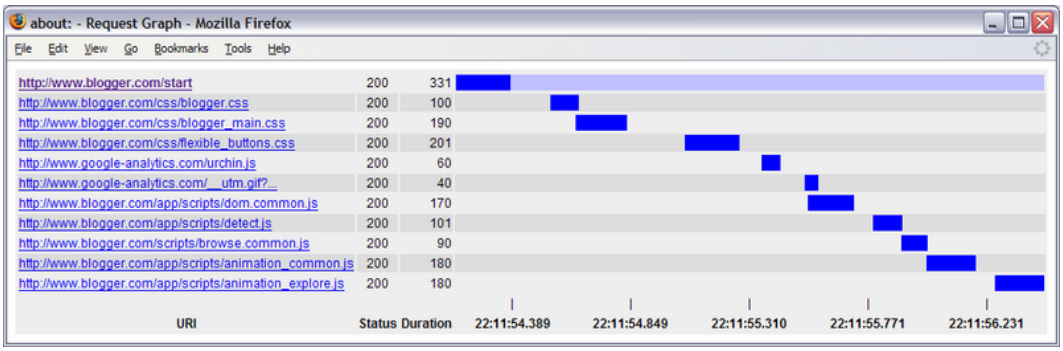


Figura 5. Resultados gráficos en Tamper Data **Fuente:** Tamper Data

Las columnas de la gráfica son:

- URL - URL completo del artículo
- Estado - Estadísticas del código HTTP
- Duración - Cuánto tiempo se tardó en descargar
- Tiempo - Un diagrama de Gantt de las solicitudes

Las barras azules más oscuras representan la duración, mientras que el azul más claro representa la duración de todos los componentes incluidos. Por ejemplo, una página HTML tendrá una barra de color azul claro que abarca la totalidad de su CSS, JavaScript y las inclusiones de la imagen.

Pasar el mouse sobre una URL revela más información sobre ese componente.

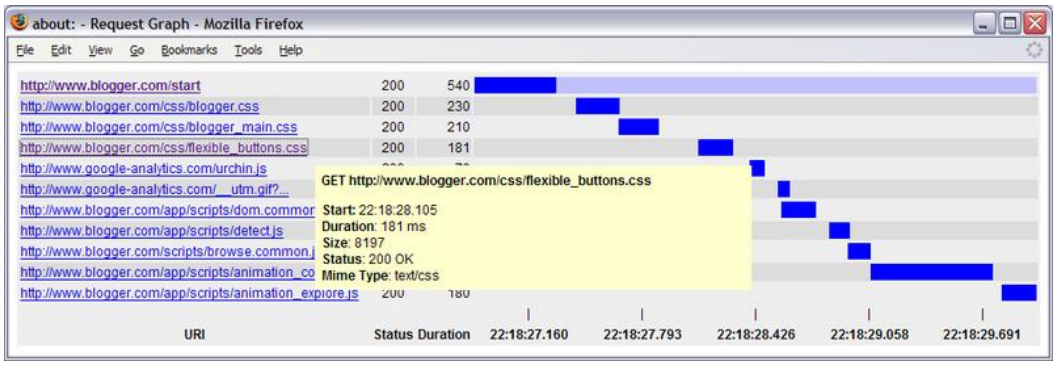


Figura 6. Información por URL en Tamper Data **Fuente:** Tamper Data

Al hacer clic en el enlace URL abre una pestaña con el contenido de ese elemento.

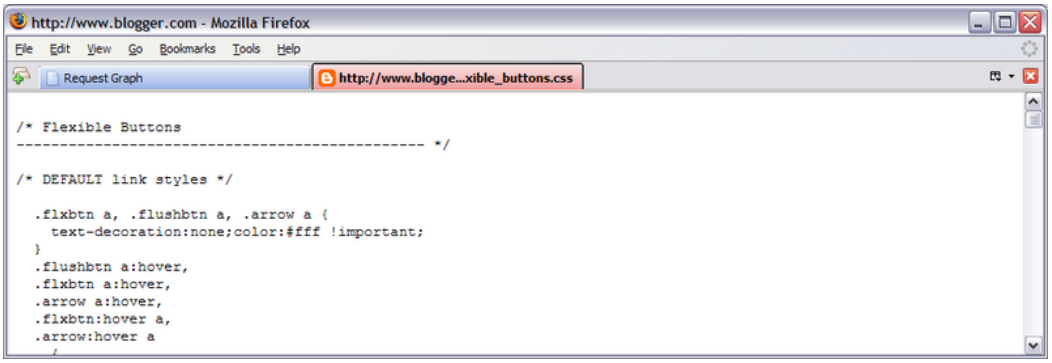


Figura 7. Contenido de una URL en Tamper Data. **Fuente:** Tamper Data

Manipulación (Tampering):

La manipulación es el acto de modificar parámetros de la petición antes de solicitarla al servidor. Para empezar la manipulación, en la ventana de solicitudes en curso, haga clic en el botón "Start tamper" en la esquina superior izquierda.

De aquí en adelante, cada vez que se emite una solicitud de nivel superior, se le pedirá que altere la solicitud. Al seleccionar el botón Tamper lanzará el Tamper emergente.

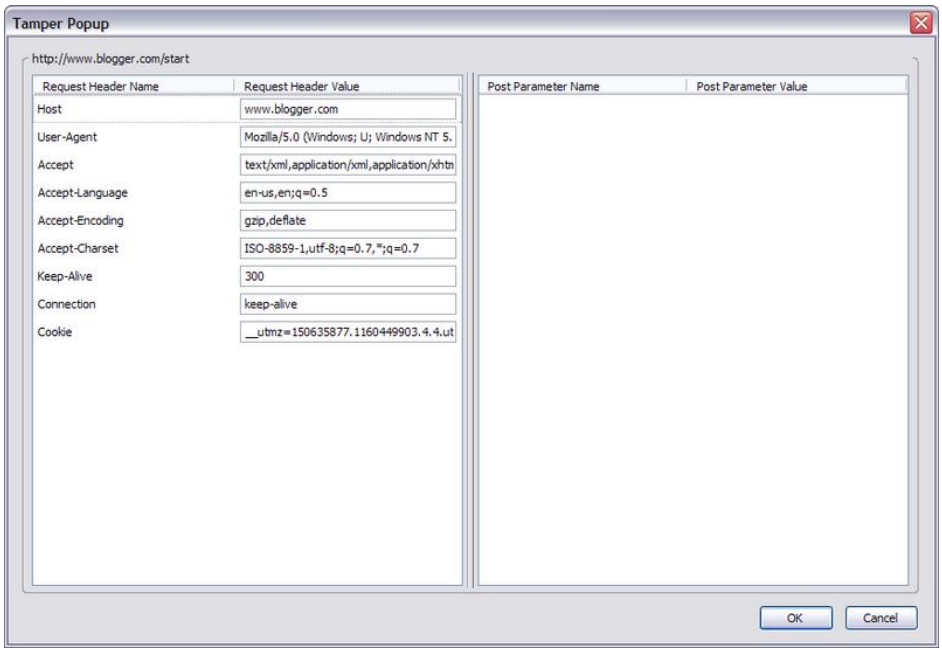


Figura 8. Tamper emergente **Fuente:** Tamper Data

Los campos de cabecera HTTP tradicionales están a la izquierda, mientras que los datos POST están hacia la derecha. Si la solicitud utiliza el método GET, entonces el lado derecho del cuadro de diálogo estará vacío.

Una vez cambiados los parámetros de la petición, al hacer clic en **Aceptar** se ejecutará la solicitud. En la ventana emergente Tamper, hacer clic en un campo que revela métodos de acceso directo para visualizar por ejemplo la codificación URL/decodificación, codificación Base64/decodificación y la eliminación de caracteres HTML.

Tamper Data es una excelente extensión para Firefox que coincide con IBM Page Detailer en características y utilidad. Cuando Firefox es un navegador permisible, Tamper Data es la clara elección entre los dos. Sin embargo, hay casos en que se requiere de un navegador no basado en Mozilla (léase: IE). En los casos raros, IBM Page Detailer es el camino a seguir.

Licencia

A diferencia de IBM Page Detailer que sólo es gratuito durante la prueba de 90 días, Tamper Data es software de código abierto. El código fuente está disponible en la página de descarga de código fuente del proyecto.

b) Hackbar

Hackbar es una barra de herramientas que proporciona una consola para diferentes tipos de peticiones web. Un cuadro de texto de tamaño variable brinda un adecuado espacio para la edición de los URI, con lo que se puede también enviar peticiones GET o POST.

Se añade como barra de herramientas debajo de la barra de direcciones principal de Firefox que se puede activar o desactivar con la tecla F9.

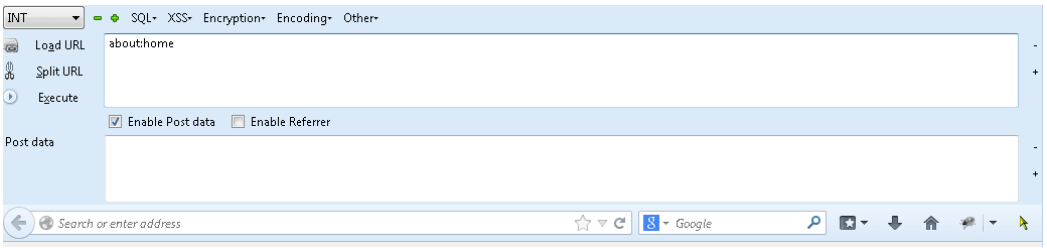


Figura 9. Interfaz de Hackbar **Fuente:** Hackbar

Los menús en la parte superior de la barra proporcionan funciones comunes para trabajar con diferentes tipos de datos, tales como los algoritmos hash o codificación y decodificación en Base64, formato URI, e incluso hexadecimal.

Figura 10. Opciones de encriptación en Hackbar **Fuente:** Hackbar

Esta barra de herramientas permite ejecutar pruebas de inyecciones SQL, XSS y de seguridad del sitio en general como validación de privilegios. Su objetivo principal es permitir hacer auditorías de seguridad.

Para utilizar la herramienta se debe conocer la petición, esta petición es posible obtenerla mediante la herramienta Tamper Data, según sea GET o POST se debe ingresar el parámetro según corresponda, como se indica en la siguiente imagen.

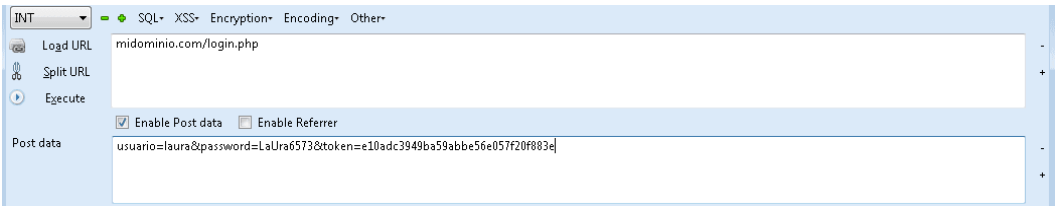


Figura 11. Ingreso de parámetros en Hackbar **Fuente:** Hackbar

Luego de ingresar los parámetros de prueba es posible realizar la petición presionando el botón “Execute”.

Con lo cual se puede realizar reproducir solicitudes al servidor para ingresar parámetros maliciosos o elevar privilegios según corresponda.

3.4.2.3. Definición de roles en las pruebas de penetración

Analista de pruebas

El rol de analista de pruebas es el responsable de realizar un análisis del sistema y en base a ello desarrollar el set de casos de prueba, elaboración del documento de casos de prueba.

El analista de pruebas es el responsable de realizar la ejecución de pruebas.

3.4.2.4. Definición de plantillas para el diseño de Casos de Prueba

En este entregable se plasman los casos de prueba que se han obtenido en base al análisis realizado a los documentos funcionales y técnicos proporcionados por el equipo de desarrollo.

La plantilla propuesta para el diseño de los casos de prueba está basada en la plantilla utilizada en el área para el diseño de los casos de prueba funcionales y de bases de datos que se realizan actualmente por el equipo de aseguramiento de la calidad.

La estructura del documento a utilizar en las pruebas es la siguiente:

1. Nombre del caso de prueba: nombre para identificar el escenario a probar. La nomenclatura a utilizar dependerá de la ya usada por cada módulo para numerar sus casos de prueba funcionales.
2. Tipo de prueba: Seguridad – Penetración.
3. Número de caso de prueba: número para identificar el escenario a probar. La numeración a utilizar dependerá de la ya usada por cada módulo para numerar sus casos de prueba funcionales.
4. Descripción de la prueba: breve descripción para definir la prueba a ejecutar, en que consiste, que se espera obtener con su ejecución.
5. Pre-condiciones: condiciones previas que se requieren para poder iniciar y ejecutar las pruebas.
6. Post-condiciones: resultado final que se espera luego de ejecutar las pruebas; como se espera que se encuentre la aplicación luego de la ejecución.
7. Paso de prueba: descripción de cada paso a ejecutar por el analista de pruebas.
8. Resultado esperado: que es lo que se espera obtener luego de ejecutar cada paso descrito. En este paso se pueden incluir imágenes.

En el **Anexo 1** se muestra el Documento de Casos de Prueba a utilizar en la metodología.

3.4.2.5. Definición de plantilla para el informe de resultados de pruebas de penetración

En este entregable se registran las observaciones encontradas durante el proceso de pruebas. Se clasifican por tipo (Critico, No Critico, Aporte, No es error) y estado (Pendiente, Aceptado, Regularizado, Conforme, Reincidente, Postergado, Rechazado).

3.4.3. Ejecución: Implementación de la metodología de pruebas de penetración

A continuación se describe cómo fue la etapa de ejecución de la metodología diseñada, el módulo seleccionado y la descripción del paso a paso de las tareas relacionadas, los documentos generados y cuáles fueron los resultados obtenidos.

3.4.3.1. Recolección de información

a) Documentos de entrada

Para conocer y evaluar los flujos que pueden ser vulnerables se utilizaron los siguientes documentos entregados por el área de análisis funcional, desarrollo y calidad.

- Análisis Funcional.
- Diseño Técnico.
- Manual de Usuario.
- Casos de prueba funcional

b) Descripción de las pruebas

El procedimiento describe los pasos a ejecutar para las pruebas de penetración en la aplicación NextStar.

Para esto se ejecutará cualquier flujo de la aplicación que involucre acciones de creación, actualización o eliminación.

El objetivo de las pruebas es verificar que estas acciones no sean permitidas para un usuario no autorizado.

Las pruebas a ejecutar son del tipo “Caja blanca” por lo que es necesario conocer la funcionalidad de la aplicación y contar con un usuario que si esté autorizado para ejecutar estas acciones. A través de la ejecución previa de alguna de estas acciones, con este usuario autorizado, se obtendrá las peticiones enviadas al servidor para la ejecución de estas acciones, para luego ser ejecutadas y enviadas con el usuario no autorizado elevando sus privilegios.

c) Pre-Condiciones

1. La aplicación web debe estar disponible y funcionando correctamente.
2. La persona que realiza las pruebas debe tener un usuario con acceso permitido a las opciones que se probarán.
3. La persona que realiza las pruebas debe tener un usuario sin acceso permitido a las opciones que se probarán.
4. Tener instalado el navegador Firefox.
5. Tener instalada la extensión Tamper Data de Firefox
6. Tener instalada la aplicación Hackbar de Firefox.
7. Se debe tener acceso de consulta a la Base de Datos de la aplicación.

d) Flujo básico de prueba

1. Iniciar sesión en el explorador Firefox
2. Abrir la extensión Tamper Data
3. Ingresar a la aplicación con el usuario que cuente con permisos en la opción donde se realizarán las pruebas.
4. Iniciar el flujo seleccionado sobre la aplicación, ejecutando cualquier acción de creación, actualización o eliminación.
5. Al finalizar la acción ejecutada ir a la ventana de Tamper Data e identificar la petición correspondiente a la acción ejecutada. Esto se puede ver en la URL.
6. Colocarse sobre esta petición e identificar y capturar la URL y el Posdata

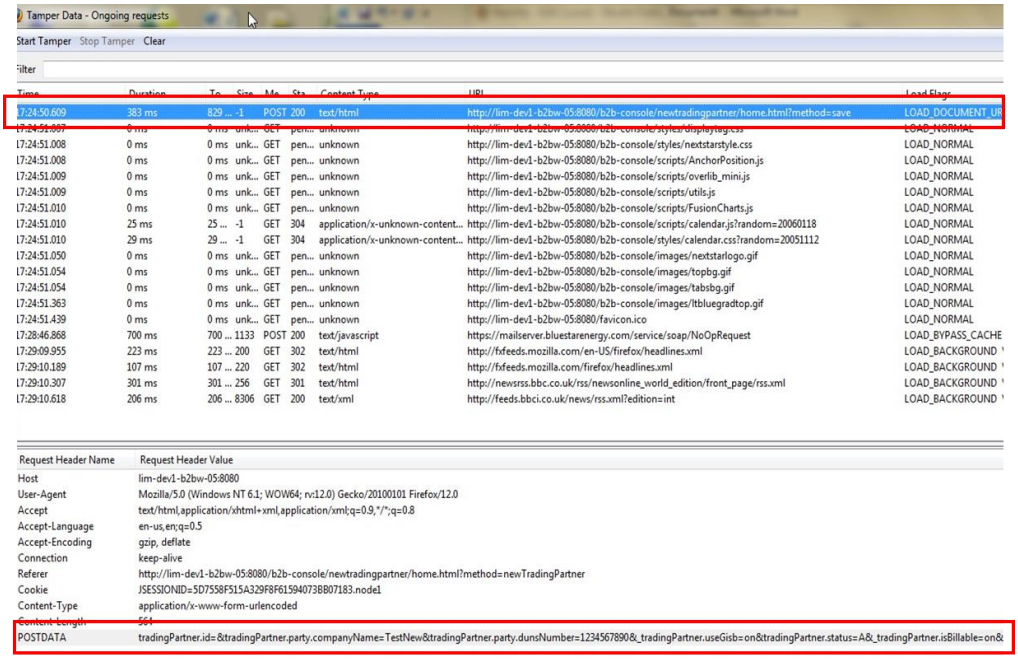


Figura 12. Ejecución de pruebas: Captura de URL y Posdata **Fuente:** Tamper Data

7. Cerrar la sesión de usuario en la aplicación.
8. Iniciar nuevamente sesión en la aplicación, pero esta vez con el usuario no autorizado para la ejecución de acciones de creación, actualización y eliminación.
9. Abrir la extensión Hackbar e inserte la información capturada en el paso 6:

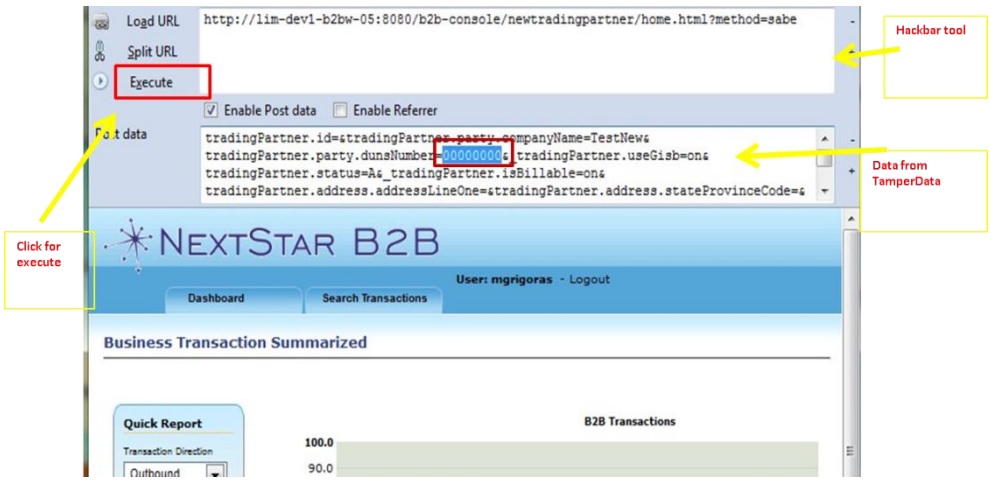


Figura 13. Ejecución de pruebas: Ingreso de parámetros en Hackbar

Fuente: Tamper Data

10. Click en "Execute“
11. Ingresar a la Base de Datos y verificar si la información fue insertada, actualizada o eliminada, según corresponda. Si esto fue realizado, este es un fallo (Issue) en la seguridad de la aplicación

e) Selección de módulo y opciones para la ejecución de las pruebas

Las comunicaciones entre todos los módulos de la aplicación se establecen a través del módulo B2B, la información se envía y se recibe a través del uso de archivos con extensión xml y edi que son recibidos y procesados por este módulo que es el responsable de validar que los archivos se encuentren bien armados, identificando a qué cliente pertenece la transacción a través de códigos de cliente y códigos de transacciones previamente definidos. Algunas de las transacciones más importantes que pasan a través de este módulo son las establecidas con el módulo Billing, son las transacciones de facturación y pagos de clientes.

Todas estas configuraciones son administradas a través de la web del módulo B2B, cualquier error en la administración o cambio no autorizado puede ocasionar grandes pérdidas de dinero para la organización ya que podrían dejar de facturarse grandes cantidades de dinero, lo cual no solo ocasionaría pérdidas económicas, sino que también supondría la pérdida de prestigio y confianza de parte de los miles de clientes que tiene la empresa.

Es por esta razón, que se decidió iniciar el proyecto ejecutando las pruebas de penetración en todas las opciones del módulo B2B.

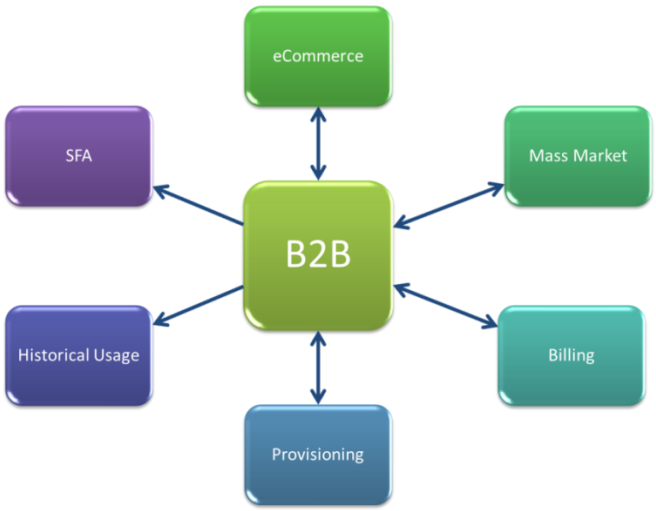


Figura 14. Relación entre B2B y otros módulos de la aplicación NextStar

Fuente: Propia

e.1.- Opciones disponibles del módulo:

La web de B2B cuenta con las siguientes opciones:

1. Dashboard
2. Search Transactions
3. Trading Partner
   - 3.1 Trading Partner
   - 3.2 Trading Partner Control
   - 3.3 Trading Partner Information
   - 3.4 Trading Partner Association
   - 3.5 Trading Partner Network
   - 3.6 Trading Partner Transaction Control
   - 3.7 Trading Partner Transaction
4. Queue Connectivity

e.2.- Opciones seleccionadas

La opción **Trading Partner** es donde se administran todas las configuraciones, por esto fue la opción seleccionada para la ejecución de las pruebas de penetración

1. Trading Partner
    - Trading Partner
    - Trading Partner Control
    - Trading Partner Information
    - Trading Partner Association
    - Trading Partner Network
    - Trading Partner Transaction Control
    - Trading Partner Transaction

e.3.- Roles del módulo

- **Guest:** usuario para consulta, solo puede acceder a las opciones: Dashboard y Search Transactions

- **Administrator:** usuario con acceso total a todas las opciones del módulo

e.4.- Gestión de usuarios y roles en B2B

La gestión de usuarios, roles y perfiles es administrada por el módulo Security, que a través de su interfaz web permite la asignación de los permisos a cada rol. Para esto se cuenta con un mapeo total de todas las opciones de cada módulo a través del uso de identificadores por cada URL, menú, submenú, botón, caja de texto, método.

3.4.3.2. Detección de vulnerabilidades

De la ejecución de la fase anterior se obtuvo una lista de todas las funcionalidades del módulo en estudio. En base a la documentación técnica y los documentos funcionales, se identificaron los flujos que podrían ser susceptibles a ataques. Además, se identificaron y diseñaron los pasos y resultados esperados en la ejecución de los casos de prueba.

a) Identificación de vulnerabilidades e impacto

El módulo B2B administra configuraciones para la comunicación entre los demás módulos de la aplicación NextStar, por lo que cuenta con un gran número de escenarios de creación, edición y eliminación de información sensible para el negocio. Este tipo de flujos son susceptibles a ataques de ethical hacking del tipo “Referencia Insegura a Objetos” a través de las cuales, el atacante puede actuar sobre la información, aún sin tener acceso a la aplicación.

Vulnerabilidad 1: Referencia insegura a objetos – Creación sin privilegios

Esta vulnerabilidad permite al atacante crear nuevas configuraciones en la aplicación. Su impacto sobre la aplicación y sobre el negocio es Bajo. El riesgo de ocurrencia es Alto debido a que tiene un nivel de dificultad de ejecución bajo.

Vulnerabilidad 2: Referencia insegura a objetos – Edición sin privilegios

Esta vulnerabilidad permite al atacante editar las configuraciones existentes en la aplicación. Su ejecución podría alterar la información existente y bloquear el flujo correcto de actividades importantes del negocio, tales como “Facturaciones”; por lo tanto, su impacto sobre la aplicación y sobre el negocio es Alto.

Vulnerabilidad 3: Referencia insegura a objetos – Eliminación sin privilegios

Esta vulnerabilidad permite al atacante eliminar las configuraciones existentes en la aplicación. Su ejecución podría eliminar configuraciones

existentes y bloquear el flujo correcto de actividades importantes del negocio, tales como “Facturaciones”; por lo tanto, su impacto sobre la aplicación y sobre el negocio es Alto.

b) Diseño de casos de prueba

Ya que el único usuario con permisos para administrar las configuraciones de B2B es el usuario administrador, las pruebas estuvieron orientadas a probar que el usuario Guest no pueda ejecutar acciones de creación, edición y eliminación de registros en estas.

Por lo tanto, el éxito de las pruebas será dado en la medida que la aplicación no permita la ejecución de estas acciones con el usuario Guest.

Los casos de prueba definidos fueron:

1. Security_Guest_Crear_Sin_Privilegios
2. Security_Guest_Actualizar_Sin_Privilegios
3. Security_Guest_Eliminar_Sin_Privilegios

Cada caso de prueba fue ejecutado para cada una de las opciones del menú Trading Partner, siendo en total 21 escenarios de prueba.

1. Security_Guest_Crear_Sin_Privilegios_Trading_Partner
2. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner
3. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner
4. Security_Guest_Crear_Sin_Privilegios_Trading_Partner_Control
5. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner_Control
6. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner_Control
7. Security_Guest_Crear_Sin_Privilegios_Trading_Partner_Information
8. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner_Information
9. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner_Information
10. Security_Guest_Crear_Sin_Privilegios_Trading_Partner_Association
11. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner_Association
12. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner_Association
13. Security_Guest_Crear_Sin_Privilegios_Trading_Partner_Network

14. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner_Network
15. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner_Network
16. Security_Guest_Crear_Sin_Privilegios_Trading_Partner_Transaction_ Control
17. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner_Transaction_ Control
18. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner_Transaction_ Control
19. Security_Guest_Crear_Sin_Privilegios_Trading_Partner_Transaction
20. Security_Guest_Actualizar_Sin_Privilegios_Trading_Partner_Transaction
21. Security_Guest_Eliminar_Sin_Privilegios_Trading_Partner_Transaction

3.4.3.3. Escalamiento de privilegios – Ejecución y resultados

En esta etapa se presenta la ejecución y resultados de las pruebas, según lo diseñado y propuesto en las etapas anteriores.

Para la ejecución del caso de prueba, el tester debe conocer previamente el flujo de las funcionalidades: crear, editar y eliminar.

a) Ejecución de casos de prueba

Pre-condiciones:

- La aplicación web está disponible.
- La base de datos está disponible.
- El analista de pruebas requiere tener 2 usuarios: administrador (con accesos de creación) e invitado (sin accesos de creación, edición y eliminación).
- La herramienta Hackbar está instalada y configurada en el equipo de prueba.
- La herramienta Tamper Data está instalada y configurada en el equipo de prueba.
- Las pruebas deben ser ejecutadas usando el navegador Firefox.

Flujos de pruebas – Vulnerabilidad: crear sin privilegios

A) Flujo: Capturar (interceptar) información al crear un nuevo registro

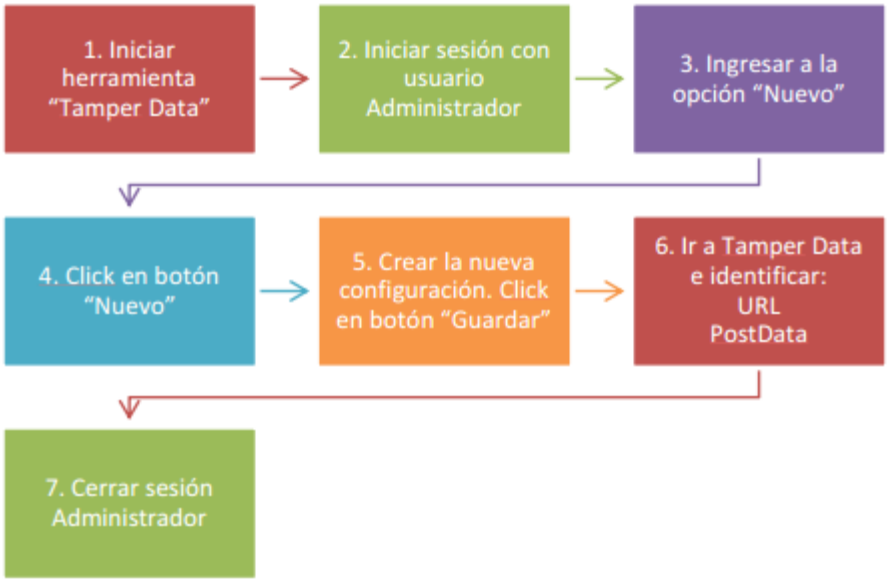


Figura 15. Flujo: Capturar información al crear un nuevo registro **Fuente:** Propia

B) Flujo: Crear sin privilegios

Figura 16. Flujo: creación sin privilegios **Fuente:** Propia

Flujos de pruebas – Vulnerabilidad: editar sin privilegios

A) Flujo: capturar (interceptar) información al editar un registro existente

Figura 17. Flujo: Capturar información al editar un registro **Fuente:** Propia

B) Flujo: editar sin privilegios

Figura 18. Flujo: editar sin privilegios **Fuente:** Propia

Flujos de pruebas – Vulnerabilidad: Eliminar sin privilegios

A) Flujo: capturar (interceptar) información al eliminar un registro existente

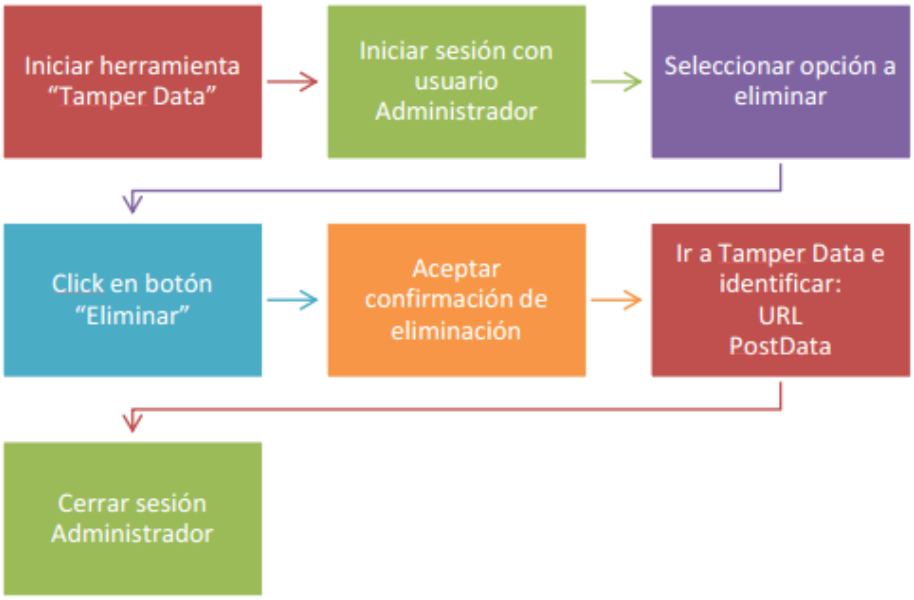


Figura 19. Flujo: Capturar información al eliminar un registro **Fuente:** Propia

B) Flujo: eliminar sin privilegios

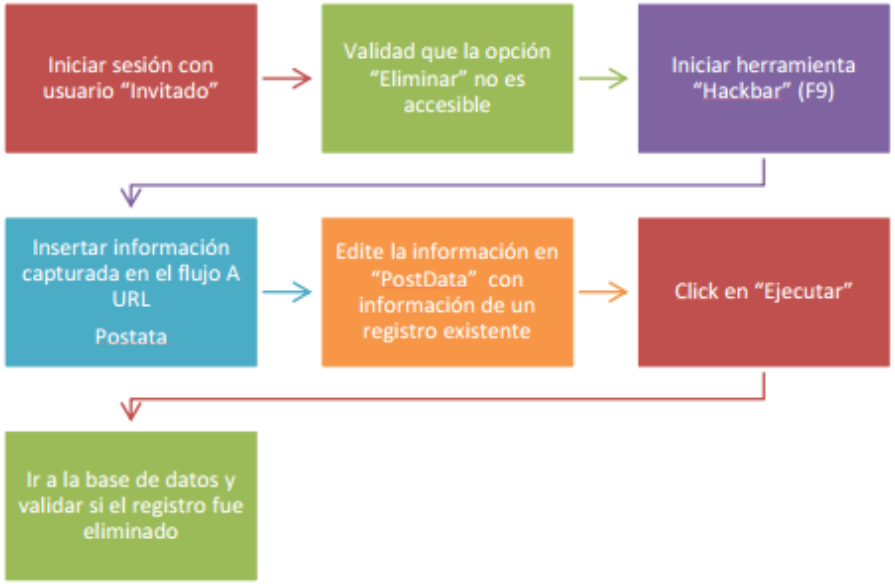


Figura 20. Eliminar sin privilegios **Fuente:** Propia

b) Resultados de ejecución

Con el fin de mantener y respetar la confidencialidad de la información de la empresa, no se adjuntan en el presente informe, todas las imágenes correspondientes a los flujos de creación, edición y eliminación. Todos los pasos completados, así como los resultados, se encuentran registrados y reportados en las herramientas utilizadas en la organización (Jira, HPQC).

A) Flujo: capturar (interceptar) información al crear un nuevo registro

Captura de información con Tamper Data:

- URL
- PostData

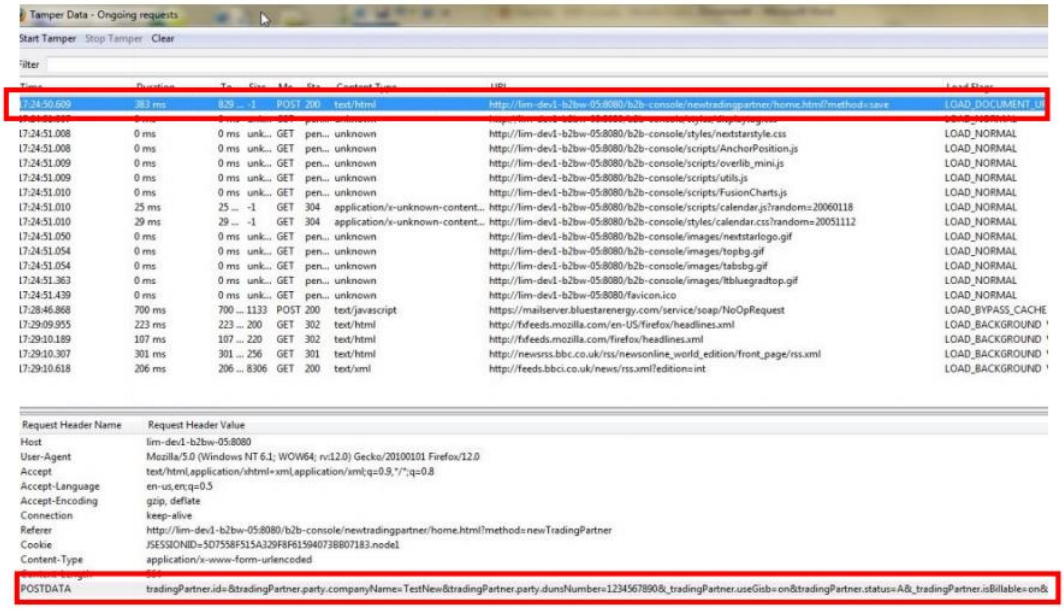


Figura 21. Resultados de pruebas captura de información con Tamper Data **Fuente:** Propia

B) Flujo: crear sin privilegios

1. La opción “Nuevo” no se encuentra disponible para el usuario “Invitado”

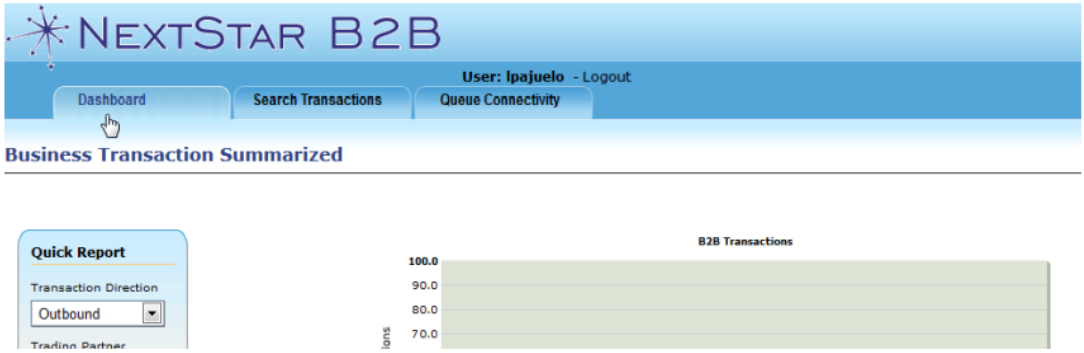


Figura 22. Resultados de pruebas: validación de privilegios limitados del usuario **Guest**
Fuente: NextStar

2. Pegar información capturada en el flujo A, usando Hackbar:

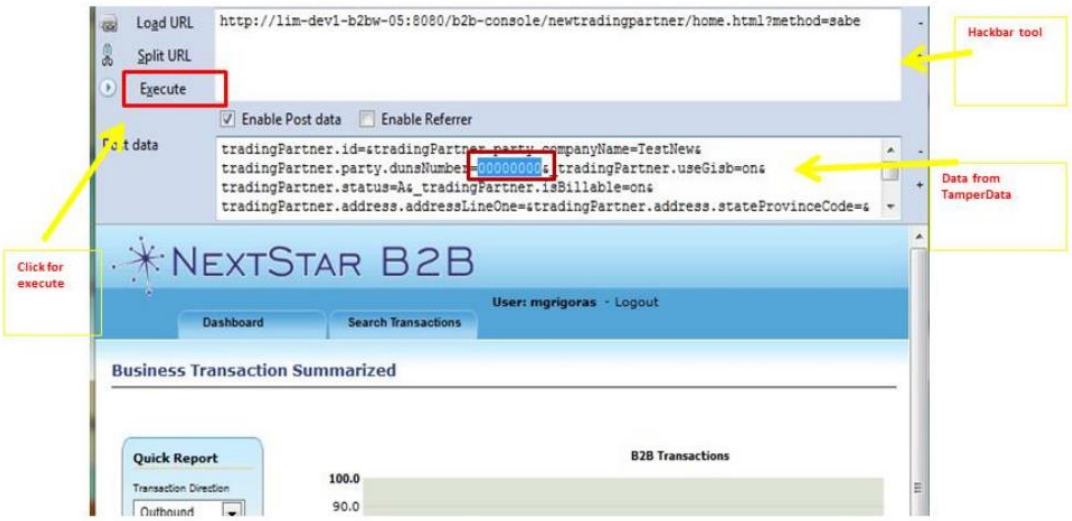


Figura 23. Resultados de pruebas: ejecución de parámetros en Hackbar con usuario **Guest** **Fuente:** NextStar

3. Se validó en la Base de Datos que el nuevo registro fue creado.
4. Conclusión: Funcionalidad es vulnerable a ataques de “Referencia insegura a objetos”

c) Casos de prueba ejecutados

La ejecución de las pruebas se realizó con los siguientes casos de prueba para cada opción.

Con el fin de mantener y respetar la confidencialidad de la información de la empresa, no se adjuntan en el presente informe, todos los casos de prueba diseñados para los 21 escenarios ejecutados, sino que se adjuntan de manera referencial: un caso de prueba de creación sin privilegios, uno de edición sin privilegios y un caso de prueba de eliminación sin privilegios. Todos los casos de pruebas completos, así como los resultados se encuentran registrados y reportados en las herramientas utilizadas en la organización (Jira, HP-QC).

1. **B2B Security-Guest – Crear sin privilegios:** El caso de prueba válida la posibilidad de crear registros en opciones que no deben ser visibles para el usuario **Guest**, a través del uso de herramientas de pruebas de penetración. **Anexo 2.**

2. **B2B Security-Guest – Actualizar sin privilegios:** El caso de prueba válida la posibilidad de actualizar registros en opciones que no deben ser visibles para el usuario **Guest**, a través del uso de herramientas de pruebas de penetración. **Anexo 3.**

3. **B2B Security-Guest – Eliminar sin privilegios:** El test case valida la posibilidad de eliminar registros en opciones que no deben ser visibles para el usuario **Guest**, a través del uso de herramientas de pruebas de penetración. **Anexo 4.**

d) Informe de resultados

La ejecución de las pruebas de penetración en el módulo B2B de NextStar arrojó como resultado que es posible la ejecución de acciones de creación, eliminación y eliminación sobre la aplicación. Por lo tanto, la aplicación se encuentra bajo riesgo de sufrir ataques internos.

En base a la ejecución de las pruebas se detectó que todos los flujos de creación, edición y eliminación son vulnerables a “Referencia insegura a objetos” pudiendo ser ejecutados sin privilegios.

A continuación se presenta gráficamente la distribución de vulnerabilidades por impacto sobre la aplicación:

Tabla 2:

Distribución de vulnerabilidades detectadas

	Nombre	Impacto	Dificultad	Riesgo
VUL01	Referencia insegura a objetos – Crear sin privilegios	Bajo	Media	**Alto**
VUL02	Referencia insegura a objetos – Actualizar sin privilegios	Alto	Media	**Alto**
VUL03	Referencia insegura a objetos – Eliminar sin privilegios	Alto	Media	**Alto**

Fuente: Propia

A continuación se muestra la cantidad de incidencias detectadas por nivel de impacto sobre la aplicación:

Tabla 3:

Cantidad de vulnerabilidades detectadas por nivel de impacto

Vulnerabilidad	Impacto bajo	Impacto medio	Impacto alto
Creación	7		
Edición			7
Eliminación			7

Fuente: Propia

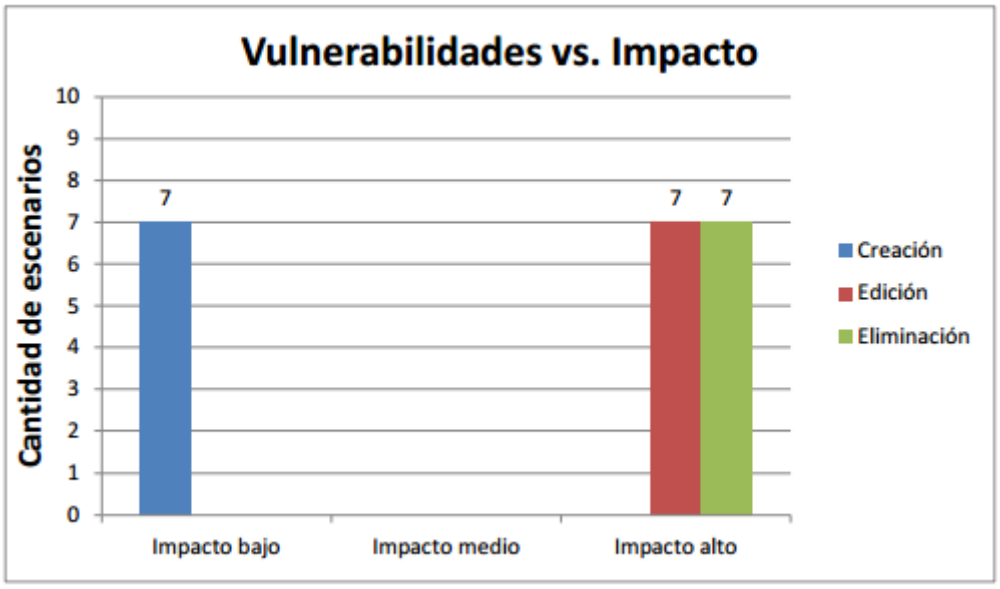


Figura 24. Distribución gráfica de vulnerabilidades vs. Nivel de impacto

Fuente: Propia

La presentación de resultados a la Jefatura y Gerencia de la empresa se realizó a través del documento de resultados que se presenta en el **Anexo 5.**

3.4.3.4. Capacitación interna

Como parte final del proyecto, se programaron reuniones de capacitación, por grupos, para todos los integrantes del área de aseguramiento de la calidad. El objetivo de la capacitación fue transferir el conocimiento obtenido a todo el equipo, con la finalidad de que este tipo de pruebas sea replicado y ejecutado en todos los módulos y funcionalidades de la aplicación NextStar.

La capacitación constó de una presentación teórica y de procedimientos **Anexo 6**, la presentación de los documentos que son parte de este documento (Anexos 2, 3, 4, 5) y se realizaron pruebas en vivo sobre la aplicación replicando los pasos expuestos en el presente informe sobre el módulo B2B y Billing.

Las capacitaciones fueron realizadas el día 27 de enero del 2014 en las instalaciones de AEP Energy Lima. Como parte final del proceso de inducción, se solicitó a cada integrante seleccionar una funcionalidad de

cualquier módulo, diseñar y ejecutar las pruebas de penetración de acuerdo a los documentos presentados en este proyecto.

Los anexos correspondientes a la capacitación se presentan en el idioma original utilizado (Inglés) que es el utilizado en la empresa para toda la documentación utilizada.

Para mantener y respetar la confidencialidad de la información de la empresa, no se adjuntan en el presente informe, los resultados obtenidos de la evaluación al equipo.

CAPÍTULO IV

REFLEXIÓN CRÍTICA DE LA EXPERIENCIA

La implementación del proyecto significó el dar un nuevo paso en la organización, tanto para el equipo de aseguramiento de la calidad como para todo el equipo de Tecnologías de Información. Este ha sido un proyecto que demandó el aprendizaje de nuevos conceptos, de nuevos tipos de ejecución de pruebas de aplicaciones, además sus resultados tuvieron repercusión en las áreas de desarrollo y arquitectura, cuyos integrante tuvieron que investigar métodos y formas de desarrollo e implementación para poder solucionar las vulnerabilidades encontradas.

Para la organización, este tipo de proyecto ha representado un paso más en los trabajos de protección de los datos de los clientes y por lo tanto en el cumplimiento de la normativa SOX, con lo cual la organización aumenta su porcentaje de éxito ante auditorías internas y externas.

Debido al éxito de este proyecto, en el área de aseguramiento de la calidad se ha propuesto la continuación de la revisión de toda la guía Owasp para continuar implementando las demás vulnerabilidades Top 10, además de nuevas herramientas e identificar nuevas necesidades.

El cumplimiento exitoso de este proyecto, representó la obtención de un reconocimiento a nivel del área y de la organización, por el buen desempeño y adecuada transferencia de conocimientos a todo el equipo de aseguramiento de la calidad (20 integrantes) con lo que se asegura que todo el equipo tiene los conocimientos necesarios para la implementación total.
Además, la investigación en el tema, llevó a conocer que existen certificaciones en el ramo de la Seguridad de Información y el hacking ético, por lo tanto la obtención de estas certificaciones se convierte en un nuevo objetivo para este año 2014.

CONCLUSIONES

Primera: Se concluyó satisfactoriamente con la definición de una metodología de pruebas de penetración para AEP Energy, la cual fue presentada a la organización con los resultados de la ejecución en el módulo B2B. La metodología será utilizada por todo el equipo de aseguramiento de la calidad como parte de la extensión de este tipo de pruebas a todos los módulos de la aplicación NextStar.

Segunda: Las herramientas Tamper Data y Hackbar han cumplido con el propósito del proyecto y serán utilizadas por todo el equipo de aseguramiento de la calidad, con el fin de extender este tipo de pruebas a todos los módulos de la aplicación NextStar.

Tercera: Mediante la revisión realizada en el módulo B2B realizado se encontraron 21 incidentes de seguridad referidos a la vulnerabilidad “Referencia Insegura a Objetos” las cuales dejan expuesta la aplicación ante el riesgo de ataques de parte de personas no autorizadas a ingresar en la aplicación, pudiendo alterar los datos del servidor web de manera crucial.

Cuarta: Al ser administrados bajo el mismo módulo de seguridad, los resultados nos indican que existe una alta probabilidad de que todos los módulos puedan ser atacados por la misma vía. De tal manera que se hace necesaria la revisión de la arquitectura utilizada para el desarrollo e implementación de las funcionalidades de la aplicación NextStar.

Quinta: Es importante indicar que ninguna arquitectura es invencible, al igual que nosotros, los atacantes están continuamente buscando vías de ataque. Por lo que se debe estar siempre alertas y

monitoreando los sistemas para tomar planes de contingencia ante algún tipo de ataque.

Sexta: Con la presentación de este proyecto quedó demostrada la capacidad del equipo de aseguramiento de la calidad de abarcar nuevos tipos de pruebas que permitirán mejorar la calidad de su trabajo y de los resultados obtenidos a través de las pruebas realizadas, asegurando así una mayor calidad del producto final.

RECOMENDACIONES

Primera: Las vulnerabilidades continuarán cambiando. Incluso sin cambiar una línea de código en su aplicación, la misma puede ser vulnerable a algo que nadie haya pensado anteriormente. Por lo tanto, es importante que los equipos de calidad, desarrollo y arquitectura de AEP Energy trabajen constantemente en la identificación de nuevas vulnerabilidades, con el propósito de identificar e implementar las soluciones de protección necesarias.

Segunda: Cuando el equipo se encuentre preparado para dejar de buscar vulnerabilidades y focalizarse en establecer controles seguros de aplicaciones, OWASP ha producido el Application Security Verification Standard (ASVS) como una guía para organizaciones y revisores de aplicaciones que detalla los controles de seguridad a verificar en una aplicación. En este sentido, se debe propiciar la lectura e interpretación de este documento a los integrantes del equipo de arquitectura, de tal manera que puedan identificar y poner en marcha un plan estratégico enfocado en la disminución de vulnerabilidades de penetración y proteger a las aplicaciones contra posibles ataques.

Tercera: Las vulnerabilidades de seguridad pueden ser bastante complejas y encontrarse ocultas en montañas de código. El enfoque más eficiente y económico para encontrar y eliminar estas vulnerabilidades es asignar expertos con buenas herramientas para realizar esta tarea.

Se recomienda hacer énfasis en la necesidad del equipo de capacitarse constantemente en la utilización de nuevas herramientas que permitan identificar nuevas vulnerabilidades.

Cuarta: Aplicaciones web seguras son solo posibles cuando se utiliza un ciclo de vida de desarrollo (SDLC por sus siglas en inglés) seguro. Para orientación sobre cómo implementar un SDLC seguro, se recomienda leer el Open Software Assurance Maturity Model (SAMM), el cual es un marco abierto para ayudar a las organizaciones a formular e implementar una estrategia para la seguridad del software que se adapte a los riesgos específicos que enfrenta la organización.

Se recomienda al equipo de arquitectura la lectura e interpretación de este documento, de tal manera que puedan identificar y poner en marcha un plan estratégico enfocado en la disminución de vulnerabilidades de penetración y proteger a las aplicaciones contra posibles ataques.

FUENTES DE INFORMACIÓN

Ali, H. (2011). BackTrack: *Assuring Security by Penetration Testing*. United State Of América.

EC-Council (2010). *Penetration Testing: Procedures & Methodologies*. United State Of America.

Hackbar. Firefox addon. [acceso agosto 2013]. Disponible en: https://addons.mozilla.org/es/firefox/addon/hackbar/

Norma Técnica Peruana ISO/IEC 27001 (2010), "EDI. Tecnología de la Información. Técnicas de seguridad. Sistemas de gestión de seguridad". Disponible en: http://bvirtual.indecopi.gob.pe/normas/isoiec27001.pdf

OWASP Top 10 – 2010: The ten Most Critical Web Application Security Risk". Owasp.org [sede web]. 2010 [acceso julio 2013]. Disponible en: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

OWASP: Category Vulnerability (2012). Owasp.org [sede web]. [acceso julio 2013]. Disponible en: https://www.owasp.org/index.php/Category:Vulnerability

Tamper Data Tutorial. Jimbojw.com. [acceso agosto 2013]. Disponible en: http://www.jimbojw.com/wiki/index.php?title=Tamper_Data

Tamper Data. Firefox addon. [acceso adigosto 2013]. Disponible en: https://addons.mozilla.org/es/firefox/addon/tamper-data/?src=search

¿What is an attack? (2012). Owasp.org [sede web]. [acceso julio 2013]. Disponible en: https://www.owasp.org/index.php/Category:Attack.

ANEXOS

ANEXO N° 1

Plantilla: Casos de Prueba

MODULO

NOMBRE DEL CASO DE PRUEBA

HISTORIAL DE REVISIONES

Fecha	Version	Descripción	Autor	Dominio	Proyecto

Nombre del caso de prueba	
Tipo de prueba	Seguridad - Penetración
Número de caso de prueba	
Descripción de la prueba	-
Pre-condiciones	Ejemplo: 1. La web debe estar disponible. 2. La Base de Datos está disponible. 3. El probador/ejecutor debe tener dos usuarios: Administrator and Guest 4. El usuario Guest debe estar configurado para no tener acceso a las opciones de creación/edición. 5. HackBar está activa y habilitada en el ambiente de pruebas. 6. Tamper Data está activa y habilitada en el ambiente de pruebas. 7. Las pruebas se deben realizar utilizando el explorador Firefox.
Post-condiciones	Ejemplo: - Los registros no son creados/editados/eliminados. - La aplicación debe mostrar el mensaje “User cannot access"
Resultado {Fecha} (Exitoso/Fallido/Incompleto)	

	Paso de prueba	**Resultado esperado**	**P**	**F**
1.				
2.				
3.				
4.				
5.				
6.				

ANEXO N° 2

Caso de Prueba: Crear información sin privilegios

B2B

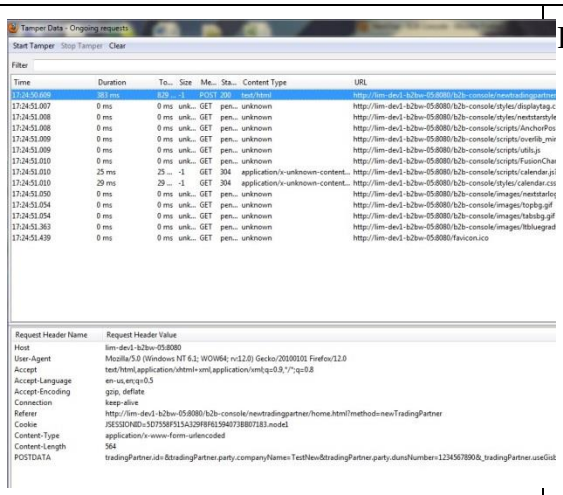
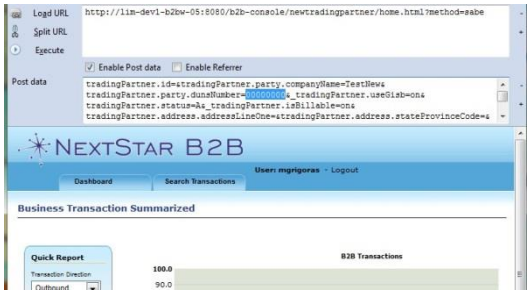
B2B SECURITY - GUEST – CREAR SIN PRIVILEGIOS

HISTORIAL DE REVISIONES

Fecha	Version	Descripción	Autor	Dominio	Proyecto
09/09/2013	1.0	Creación.	Laura Pajuelo	**B2B**	**Pruebas de penetración**

Nombre del caso de prueba	B2B SECURITY - GUEST – Creación sin privilegios
Tipo de prueba	Seguridad - Penetración
Número de caso de prueba	TCBTB09_02_B2B
Descripción de la prueba	- El caso de prueba valida la posibilidad de crear registros en opciones que no deben ser visibles para el usuario **Guest,** a través del uso de herramientas de pruebas de penetración. Las opciones que no son visibles para el usuario Guest, son: New Trading Partner * Trading Partner * Trading Partner Control * Trading Partner Information * Trading Partner Association * Trading Partner Network * Trading Partner Transaction Control * Trading Partner Transaction
Pre-condiciones	8. La web B2B debe estar disponible. 9. La Base de Datos Bse_eis está disponible. 10. El probador/ejecutor debe tener dos usuarios: Administrator and Guest 11. El usuario Guest debe estar configurado para no tener acceso a las opciones del menú "New trading partner" 12. HackBar está activa y habilitada en el ambiente de pruebas. 13. Tamper Data está active y habilitada en el ambiente de pruebas. 14. Las pruebas se deben realizar utilizando el explorador Firefox.
Post-condiciones	- Los registros en trading partner no son editados. - La aplicación debe mostrar el mensaje “User cannot access"
Resultado 09/12/2013 (Exitoso/Fallido/Incompleto)	Fallido

	Paso de prueba	**Resultado esperado**	**P**	**F**
1.	Con el usuario Administrator, ingresar a la web de B2B en el ambiente de prueba correspondiente.	La web se muestra correctamente con todas las opciones del menú disponibles.	P	
2.	Click on New button. Crear a new trading.	El Trading Partner es creado.	P	

3.	Al finalizar la acción ejecutada ir a la ventana de Tamper Data e identificar la petición correspondiente a la acción ejecutada. Esto se puede ver en la URL. Colocarse sobre esta petición e identificar y capturar la URL y el Posdata		P	
4.	Cerrar la sesión Administrator y volver a ingresar usando el usuario Guest.	La opción Trading Partner no debe mostrarse.	P	
5.	Use HackBar e inserte la información capturada en el paso anterior Editar el valor del trading information por un nuevo valor Click en "Execute"	Se ejecutará la acción enviada y ninguna data debe ser modificada. La web mostrará un mensaje de error de acceso. 	F	
6.	Acceder a la base de datos bse_eis en el correspondiente ambiente y ejecute el siguiente script para verificar que la data no ha sido insertada. **SELECT** tp.partyCompanyName, concat_ws (" ",tpt.txcode, tpt.txdescription), tptc.direction, tptc.active, tptc.generableQuantity **FROM** tradingpartnertransactioncontrol tptc **INNER JOIN** tradingpartner tp on tptc.tradingpartnerid=tp.id inner join tradingpartnertransaction tpt **ON** tpt.id=tptc.tradingpartnertransactionid order by tp.partyCompanyName **ASC**	Verificar que la nueva data no fue insertada.	F	

ANEXO N° 3

Caso de Prueba: Actualizar información sin privilegios

B2B

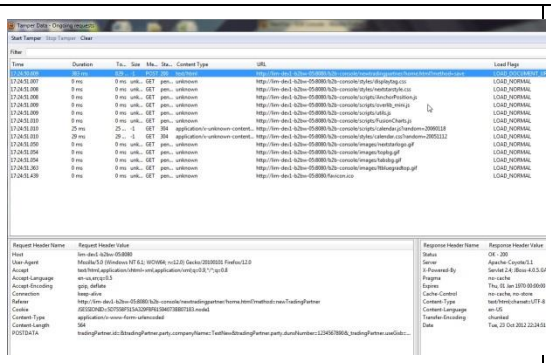
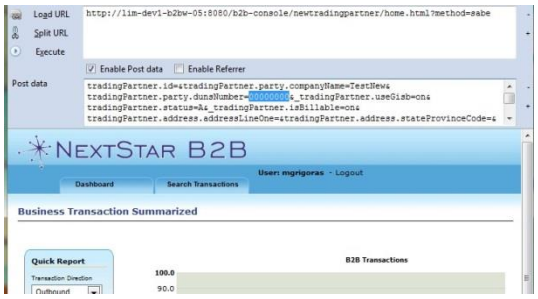
B2B SECURITY - GUEST – ACTUALIZAR SIN PRIVILEGIOS

HISTORIAL DE REVISIONES

Fecha	Version	Descripción	Autor	Dominio	Proyecto
09/09/2013	1.0	Creación.	Laura Pajuelo	**B2B**	**Pruebas de penetración**

Nombre del caso de prueba	B2B SECURITY - GUEST – Actualizar sin privilegios
Tipo de prueba	Seguridad - Penetración
Número de caso de prueba	TCBTB09_02_B2B
Descripción de la prueba	- El caso de prueba valida la posibilidad de actualizar registros en opciones que no deben ser visibles para el usuario **Guest,** a través del uso de herramientas de pruebas de penetración. Las opciones que no son visibles para el usuario Guest, son: New Trading Partner * Trading Partner * Trading Partner Control * Trading Partner Information * Trading Partner Association * Trading Partner Network * Trading Partner Transaction Control * Trading Partner Transaction
Pre-condiciones	1. La web B2B debe estar disponible. 2. La Base de Datos Bse_eis está disponible. 3. El probador/ejecutor debe tener dos usuarios: Administrator and Guest 4. El usuario Guest debe estar configurado para no tener acceso a las opciones del menú "New trading partner" 5. HackBar está activa y habilitada en el ambiente de pruebas. 6. Tamper Data está active y habilitada en el ambiente de pruebas. 7. Las pruebas se deben realizar utilizando el explorador Firefox.
Post-condiciones	- Las registros en trading partner no son editadas. - La aplicación debe mostrar el mensaje “User cannot access"
Resultado 09/27/2013 (Exitoso/Fallido/Incompleto)	Fallido

	Paso de prueba	**Resultado esperado**	**P**	**F**
1.	Con el usuario Administrator, ingresar a la web de B2B en el ambiente de prueba correspondiente.	La web se muestra correctamente con todas las opciones del menú disponibles.	P	
2.	Click en el botón Edit. Editar el registro.	El Trading Partner es editado.	P	

3.	Al finalizar la acción ejecutada ir a la ventana de Tamper Data e identificar la petición correspondiente a la acción ejecutada. Esto se puede ver en la URL. Colocarse sobre esta petición e identificar y capturar la URL y el Posdata		P	
4.	Cerrar la sesión Administrator y volver a ingresar usando el usuario Guest.	La opción Trading Partner no debe mostrarse.	P	
5.	Use HackBar e inserte la información capturada en el paso anterior Cambiar el tradingPartner.party.dunsNumber = 00000000 Click en "Execute"	Se ejecutará la acción enviada y ninguna data debe ser modificada. La web mostrará un mensaje de error de acceso. 	F	
6.	Acceder a la base de datos bse_eis en el correspondiente ambiente y ejecute el siguiente script para verificar que la data no ha sido insertada. **SELECT** tp.partyCompanyName, concat_ws (" ",tpt.txcode, tpt.txdescription), tptc.direction, tptc.active, tptc.generableQuantity **FROM** tradingpartnertransactioncontrol tptc **INNER JOIN** tradingpartner tp on tptc.tradingpartnerid=tp.id inner join tradingpartnertransaction tpt **ON** tpt.id=tptc.tradingpartnertransactionid order by tp.partyCompanyName **ASC**	Verificar que la nueva data no fue actualizada.	F	

ANEXO N° 4

Caso de prueba: Eliminar sin privilegios

B2B

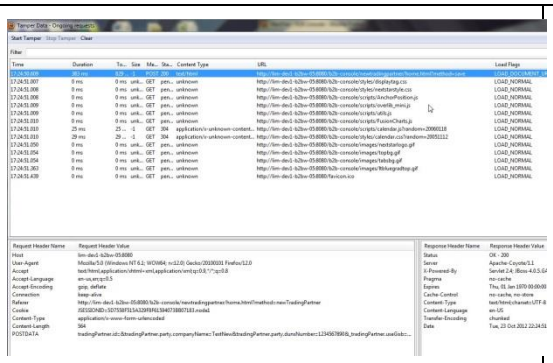
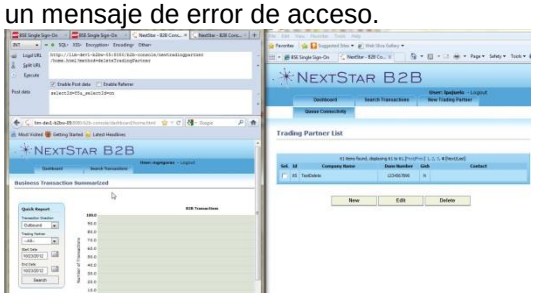
B2B SECURITY - GUEST – ELIMINAR SIN PRIVILEGIOS

HISTORIAL DE REVISIONES

Fecha	Version	Descripción	Autor	Dominio	Proyecto
09/09/2013	1.0	Creación.	Laura Pajuelo	**B2B**	**Pruebas de penetración**

Nombre del caso de prueba	B2B SECURITY - GUEST – Eliminar sin privilegios
Tipo de prueba	Seguridad - Penetración
Número de caso de prueba	TCBTB09_02_B2B
Descripción de la prueba	- Tel test case valida la posibilidad de eliminar registros en opciones que no deben ser visibles para el usuario **Guest,** a través del uso de herramientas de pruebas de penetración. Las opciones que no son visibles para el usuario Guest, son: New Trading Partner * Trading Partner * Trading Partner Control * Trading Partner Information * Trading Partner Association * Trading Partner Network * Trading Partner Transaction Control * Trading Partner Transaction
Pre-condiciones	1. La web B2B debe estar disponible. 2. La Base de Datos Bse_eis está disponible. 3. El probador/ejecutor debe tener dos usuarios: Administrator and Guest 4. El usuario Guest debe estar configurado para no tener acceso a las opciones del menú "New trading partner" 5. HackBar está activa y habilitada en el ambiente de pruebas. 6. Tamper Data está active y habilitada en el ambiente de pruebas. 7. Las pruebas se deben realizar utilizando el explorador Firefox.
Post-condiciones	- Los registros en trading partner no son eliminados. - La aplicación debe mostrar el mensaje “User cannot access"
Resultado 07/29/2013 (Exitoso/Fallido/Incompleto)	Fallido

	Paso de prueba	**Resultado esperado**	**P**	**F**
1.	Con el usuario Administrator, ingresar a la web de B2B en el ambiente de prueba correspondiente.	La web se muestra correctamente con todas las opciones del menú disponibles.	P	
2.	Click en el botón Eliminar. Eliminar un registro.	El Trading Partner es eliminado.	P	

3.	Al finalizar la acción ejecutada ir a la ventana de Tamper Data e identificar la petición correspondiente a la acción ejecutada. Esto se puede ver en la URL. Colocarse sobre esta petición e identificar y capturar la URL y el Posdata.		P	
4.	Cerrar la sesión Administrator y volver a ingresar usando el usuario Guest.	La opción Trading Partner no debe mostrarse.	P	
5.	Use HackBar e inserte la información capturada en el paso anterior Editar el id. Click en "Execute"	Se ejecutará la acción enviada y ninguna data debe ser modificada. La web mostrará un mensaje de error de acceso. 	F	
6.	Acceder a la base de datos bse_eis en el correspondiente ambiente y ejecute el siguiente script para verificar que la data no ha sido eliminada. **SELECT** tp.partyCompanyName, concat_ws (" ",tpt.txcode, tpt.txdescription), tptc.direction, tptc.active, tptc.generableQuantity **FROM** tradingpartnertransactioncontrol tptc **INNER JOIN** tradingpartner tp on tptc.tradingpartnerid=tp.id inner join tradingpartnertransaction tpt **ON** tpt.id=tptc.tradingpartnertransactionid order by tp.partyCompanyName **ASC**	Verificar que la nueva data no fue eliminada.	F	

ANEXO N° 5

Informe de resultados de pruebas de penetración

PRUEBAS DE PENETRACION
REPORTE DE RESULTADOS

HISTORIAL DE REVISIONES

Fecha	Version	Descripción	Autor
10/11/2013	1.0	Creación.	Laura Pajuelo

1. RESUMEN

Con el fin de establecer una adecuada gestión de información de seguridad de los módulos de aplicación NextStar, AEP ha desarrollado e implementado un módulo de seguridad para administrar roles y perfiles de usuarios que acceder a la aplicación en los diferentes entornos.

En este contexto, el equipo de control de calidad está considerando la implementación institucional de Hacking Ético para evaluar la seguridad del módulo de seguridad. Como primera etapa se define las pruebas piloto en la aplicación web B2B.

2. ALCANCE

El alcance de las pruebas incluye el análisis de las vulnerabilidades y los riesgos asociados con la siguiente aplicación y opciones:

Aplicación / Opción	*Usuario Administrator*	*Usuario Guest*
B2B / New Trading Partner / Trading Partner	✓	✓
B2B / New Trading Partner / Trading Partner Control	✓	✓
B2B / New Trading Partner / Trading Partner Information	✓	✓
B2B / New Trading Partner / Trading Partner Association	✓	✓
B2B / New Trading Partner / Trading Partner Network	✓	✓
B2B / New Trading Partner / Trading Partner Transaction Control	✓	✓
B2B / New Trading Partner / Trading Partner Transaction	✓	✓

3. ACTIVIDADES EJECUTADAS

Las pruebas se llevaron a cabo dentro de la red interna de AEP sin conocimiento previo de los controles de seguridad implementados.

Herramientas utilizadas

- Firefox navigator version 22.0
- Tamper data - Firefox extension
- Hackbar - Firefox extension
- SQL yog

Casos de prueba ejecutados

- B2B SECURITY - GUEST - Create Without Privileges
- B2B SECURITY - GUEST - Update Without Privileges
- B2B SECURITY - GUEST - Delete Without Privileges

4. RESULTADOS

El objetivo de las pruebas ejecutadas fue validar casos de prueba de permisos de ejecución de acciones del usuario Guest en la web con el fin de salvaguardar la seguridad de aplicaciones web bajo la responsabilidad del módulo de seguridad. Las actividades y pruebas ejecutadas ayudaron a identificar las vulnerabilidades que podrían ser utilizados por un atacante interno para obtener conocimientos acerca de los recursos de información internos y / o el acceso a información no autorizada.

En total, se identificaron tres vulnerabilidades, replicados en siete opciones de la web B2B.

Risk		
High	Medium	Low
14	0	7

Descripción de los riesgos

	Nombre	Impacto	Dificultad	Riesgo
VUL01	Referencia insegura a objetos – Crear sin privilegios	Bajo	Media	Bajo
VUL02	Referencia insegura a objetos – Actualizar sin privilegios	Alto	Media	Alto
VUL03	Referencia insegura a objetos – Eliminar sin privilegios	Alto	Media	Alto

Descripción de vulnerabilidades

VUL01	Referencia insegura a objetos – Crear sin privilegios		
Descripción	Es posible crear registros con el usuario invitado.		
Repercusión	El acceso a las acciones que manipulan información de parte de un grupo de usuarios no autorizados, puede ocasionar que se lleven acciones no autorizadas y que puedan alterar los resultados de las transacciones. Existe una alta probabilidad de que la misma vulnerabilidad está presente en otros módulos con mayor impacto.		
Riesgos	**Impacto**	**Probabilidad**	**Riesgo**
	Bajo	Medio	**Bajo**
Detalle	**Acción**	**Evidencia**	**Incidencias**
	Crear	01	7

VUL02	Referencia insegura a objetos – Editar sin privilegios
Descripción	Es posible editar registros con el usuario invitado.
Repercusión	El acceso a las acciones que manipulan información de parte de un grupo de usuarios no autorizados, puede ocasionar que se lleven acciones no autorizadas y que puedan alterar los resultados de las transacciones.

	Existe una alta probabilidad de que la misma vulnerabilidad está presente en otros módulos con mayor impacto.
Riesgos	**Impacto** **Probabilidad** **Riesgo** **Alto** Medio **Alto**
Detalle	**Acción** **Evidencia** **Incidencias** Editar 02 7

VUL03	**Referencia insegura a objetos – Eliminar sin privilegios**
Descripción	Es posible eliminar registros con el usuario invitado.
Repercusión	El acceso a las acciones que manipulan información de parte de un grupo de usuarios no autorizados, puede ocasionar que se lleven acciones no autorizadas y que puedan alterar los resultados de las transacciones. Existe una alta probabilidad de que la misma vulnerabilidad está presente en otros módulos con mayor impacto.
Riesgos	**Impacto** **Probabilidad** **Riesgo** **Alto** Medio **Alto**
Detalle	**Acción** **Evidencia** **Incidencias** Eliminar 03 7

5. COMENTARIOS

La aplicación que es parte del alcance de esta revisión contiene vulnerabilidades de alto riesgo de ocurrencia debido a que la arquitectura del módulo de seguridad no aplica validaciones y restricciones para la información de URL y la información enviada en el mensaje, a través del cual cualquier usuario sin accesos asignados a las aplicaciones para ejecutar ninguna acción (crear, editar, borrar).

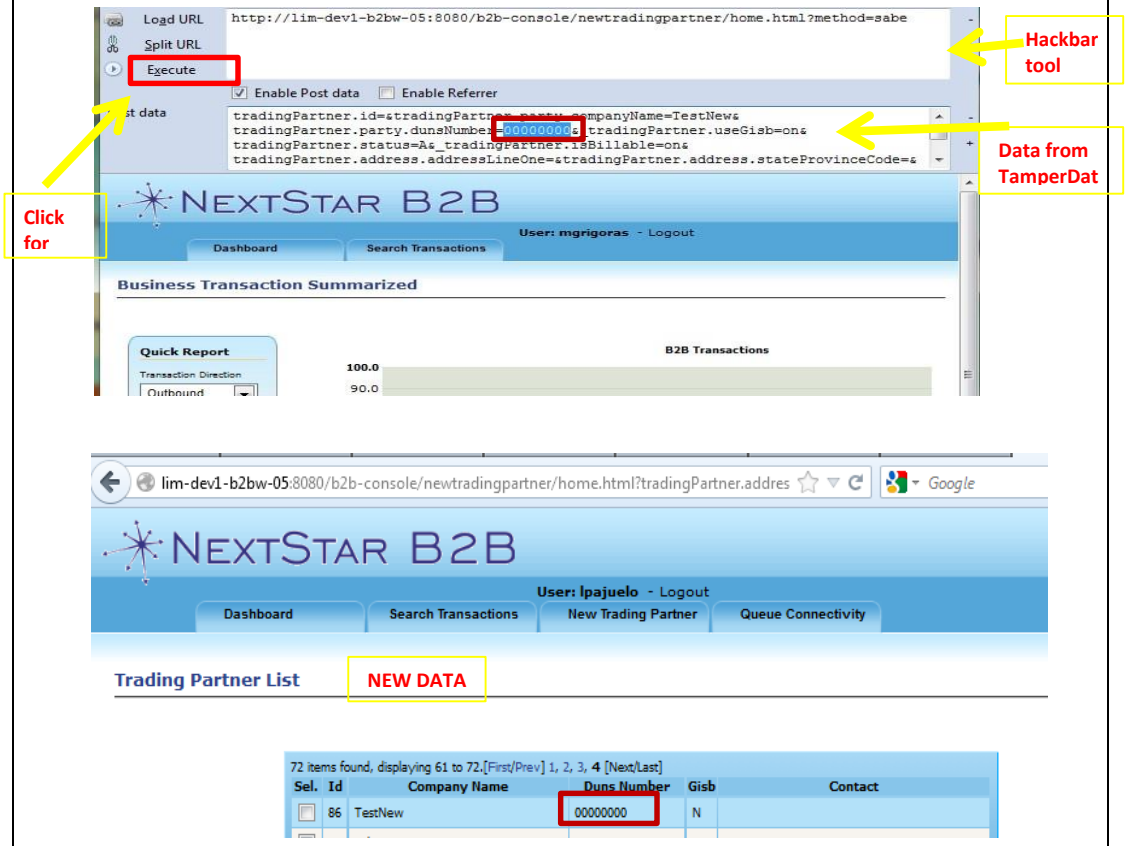
La revisión encontró que los métodos se envían a través de la URL siendo así posible capturarlos y posteriormente utilizarlos a través de herramientas gratuitas para la ejecución de los ataques.

EVIDENCIAS

EVI01

B2B WEB
TRADING PARTNER LIST
NEW TRADING

URL: http://lim-dev1-b2bw-05:8080/b2b-console/newtradingpartner/home.html?method=sabe
POST: tradingPartner.id=&tradingPartner.party.companyName=TestNew&tradingPartner.party.dunsNumber+1234567890&tradingPartner.useGisb=No



RESPONSE: Data creada

EVI02

B2B WEB

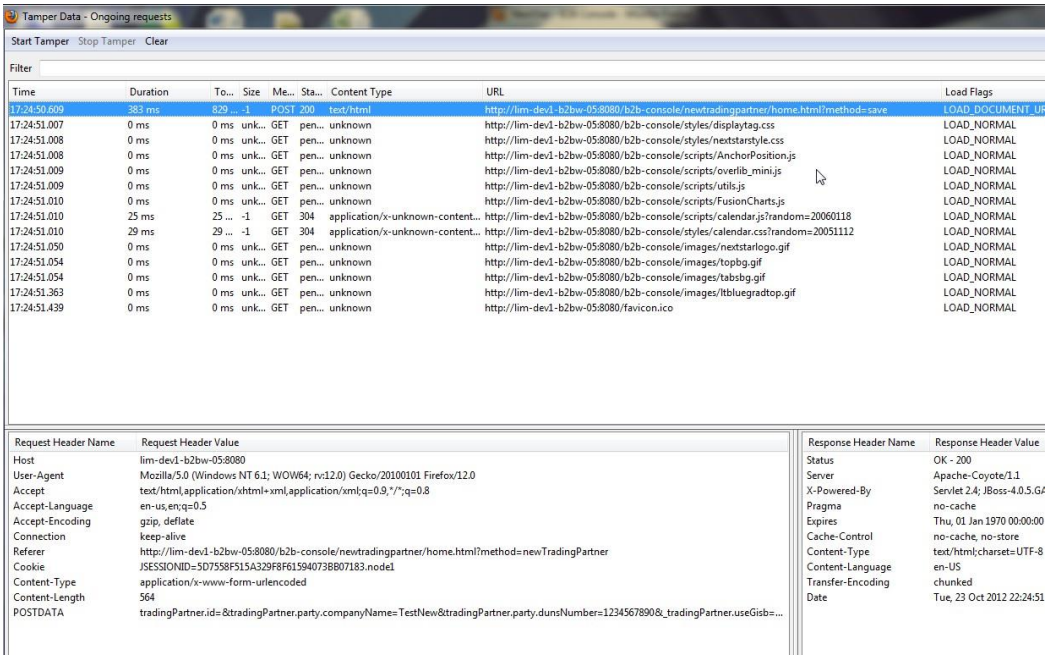
TRADING PARTNER LIST

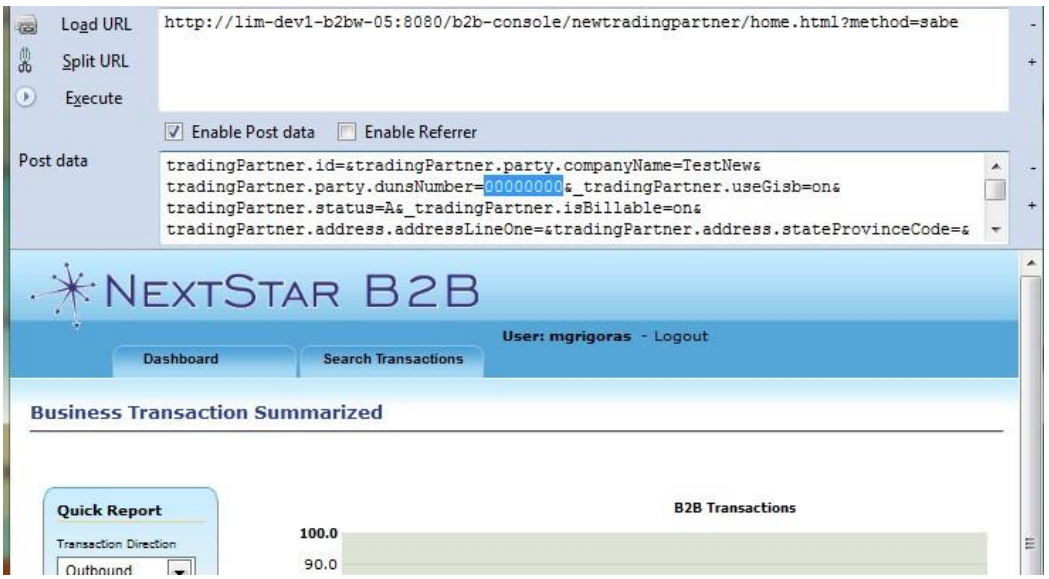
EDIT TRADING

URL: http://lim-dev1-b2bw-05:8080/b2b-console/newtradingpartner/home.html?method=save

POST:

tradingPartner.id=&tradingPartner.party.companyName=TestNew&tradingPartner.party.dunsNumber=00000000&tradingPartner.useGisb=on&tradingPartner.status=A&_tradingPartner.isBillable=on&tradingPartner.address.address.addressLineOne=&tradingPartner.address.stateProvinceCode=&





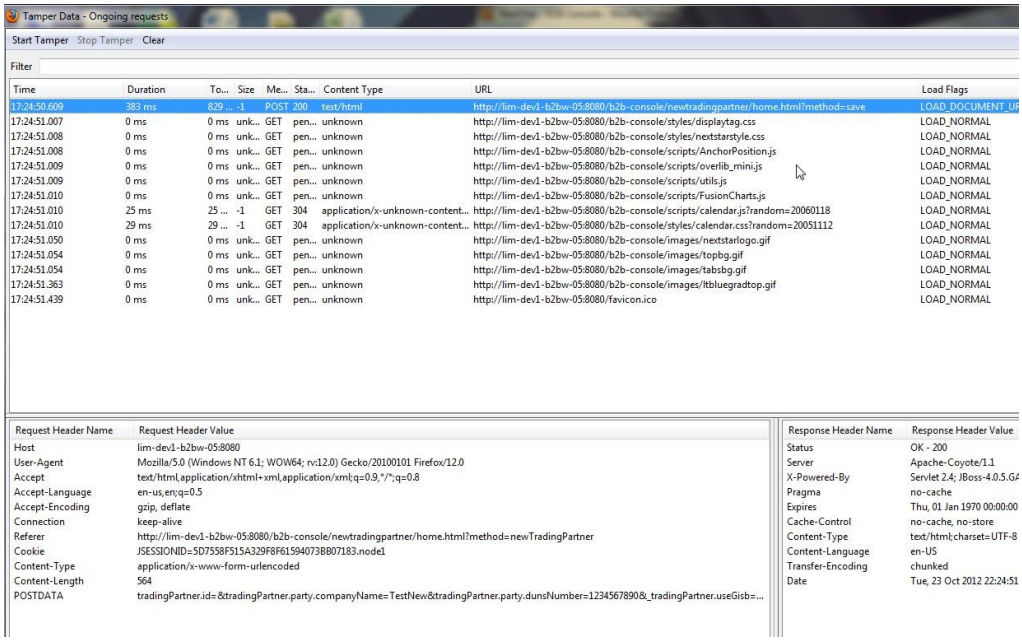
RESPONSE: Data editada

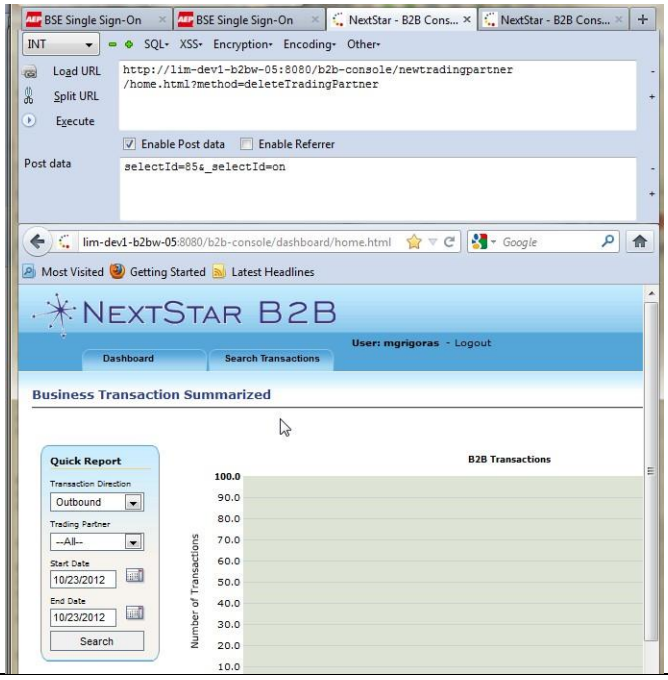
EVI03

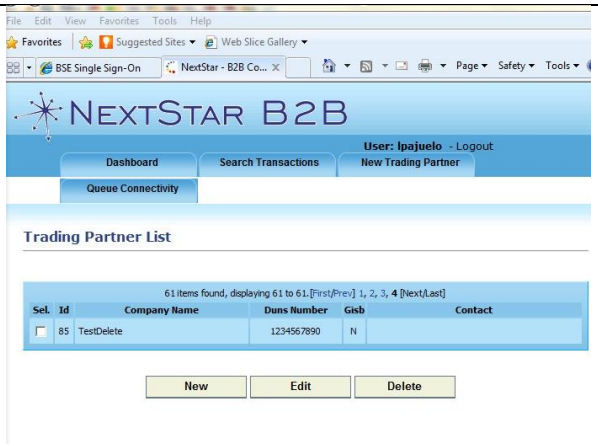
B2B WEB
TRADING PARTNER LIST
DELETE TRADING

URL: http://lim-dev1-b2bw-05:8080/b2b-console/newtradingpartner/home.html?method=deleteTradingPartner

POST: selectId=85&selectId=on







RESPONSE: Data eliminada

ANEXO N° 6

Capacitación interna – Material presentado

AEP ENERGY℠

Penetration Testing

AEP Energy is a competitive retail electric service provider affiliated with American Electric Power, Inc.
AEP Energy is not soliciting on behalf of and is not an agent for any AEP utility.

Agenda

- ¿What is Penetration Testing?
- ¿Why Penetration Testing?. Benefits
- ¿How we do penetration testing?
- Tools
- DEMO
- Evaluation

AEP ENERGY
2

¿What is Penetration Testing?

It is a vulnerabilities assessment method, executed pretending to be a malicious attacker.

¿Why Penetration Testing?. Benefits

Despite the meteoric rise of hackers in the public consciousness and the very real increase in the number of external attacks on company's websites and internet connections, reputable authorities such as the National Audit Office still maintain that internal attacks are much more common. Survey figures show that between 65% and 80% of security breaches come from employees or contractors of the victim organization.

¿Why Penetration Testing?. Benefits

Preserve corporate image and customer loyalty

Even a single incident of compromised customer data can be costly. Penetration testing helps prevent data incidents that put goodwill and reputation of your company is at stake.

Justifying security investments

Penetration testing can both evaluate the effectiveness of existing security products and build the case of the proposed investments.

¿Why Penetration Testing?. Benefits

Manage vulnerabilities Cleverly

Penetration testing provides detailed information on security threats, real exploitable. By performing a penetration test can identify which vulnerabilities are critical, which are negligible, and which are false positive. This allows you to intelligently apply patches and allocate security resources when and where they need it most.

Avoid cost of downtime network

Recovery of security breach can cost millions due to IT remediation efforts, loss of employee productivity and lost revenue. Penetration testing allows you to avoid this economic loss to identify and address risks before security breaches occur.

OWASP Top 10

The Open Web Application Security Project, focuses exclusively on penetration testing for web applications, providing an exhaustive catalog of 66 security controls to review in all web application.

OWASP Top 10 – 2010 (Previous)	OWASP Top 10 – 2013 (New)
A1 – Injection	A1 – Injection
A3 – Broken Authentication and Session Management	A2 – Broken Authentication and Session Management
A2 – Cross-Site Scripting (XSS)	A3 – Cross-Site Scripting (XSS)
A4 – Insecure Direct Object References	A4 – Insecure Direct Object References
A6 – Security Misconfiguration	A5 – Security Misconfiguration
A7 – Insecure Cryptographic Storage – Merged with A9 →	A6 – Sensitive Data Exposure
A8 – Failure to Restrict URL Access – Broadened into →	A7 – Missing Function Level Access Control
A5 – Cross-Site Request Forgery (CSRF)	A8 – Cross-Site Request Forgery (CSRF)
<buried in A6: Security Misconfiguration>	A9 – Using Known Vulnerable Components
A10 – Unvalidated Redirects and Forwards	A10 – Unvalidated Redirects and Forwards
A9 – Insufficient Transport Layer Protection	Merged with 2010-A7 into new 2013-A6

A4 - Insecure Direct Object References

Is when a web application exposes an internal implementation object to the user. Some examples of internal implementation objects are database records, URLs, or files.

An attacker can modify the internal implementation object in an attempt to abuse the access controls on this object. When the attacker does this they may have the ability to access functionality that the developer didn't intend to expose access to.

¿How we do penetration testing?

We will intercept the information using a super user. With it we will update relevant informacion that is only permissible for this user.

After that we will switch to an unprivileged user, and we will try to execute the same actions.

We are going to steal "hidden information"

Tools

Tools for Insecure reference object

Tamper Data:

Is a Firefox toolbar used to view and modify HTTP/HTTPS headers and post parameters.

Used for execution of web applications security testing by modifying POST parameters.

Tools for Insecure reference object

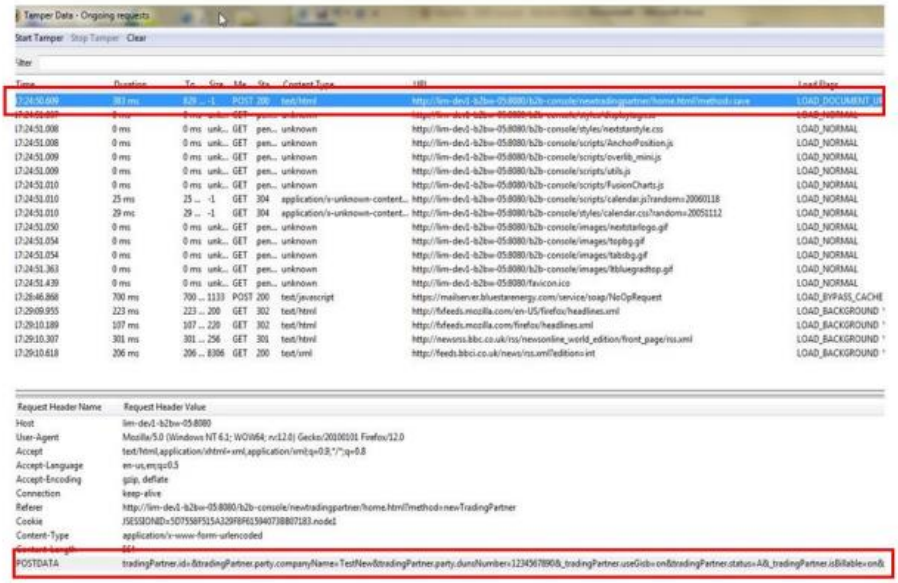
HackBar

I a Firefox toolbar that will help you test the SQLs injection, XSS holes and site security.

Basic Flow for PENTEST

- Open Tamper Data extension
- Open the application and initiate the flow to capture
- At the end of the flow go to the Tamper Data window and identify the following information: URL, POSTDATA

Basic Flow for PENTEST

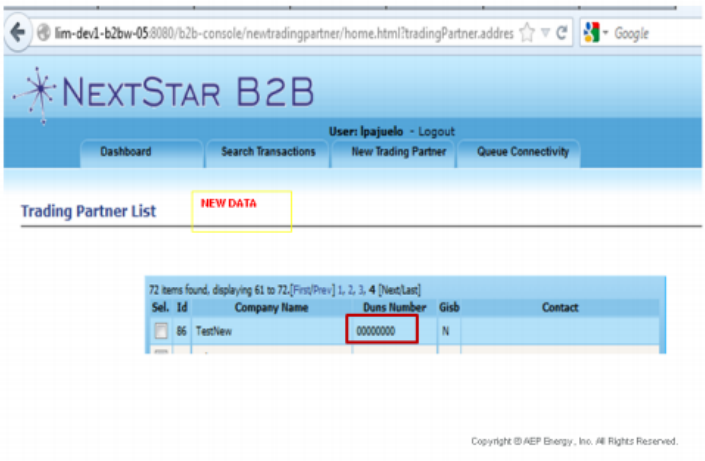


Basic Flow for PENTEST

- Press F9 and open Hackbar extension
- Use HackBar insert the information from tamper Data as the following picture:

Basic Flow for PENTEST

- Click en "Execute“
- Verify is the data inserted is save. If is save, this an Security issue.



DEMO

Billing Web
Add and approve Charges

Requirements:

- Billing Web is available
- Bse_core database is available.
- Tester requires have two test users: Administrator and Specialist.
- The Administrator user must be configured to set charges but not to Approve
- The Specialist user must be configured to approve charges.
- Hack Bar tool is enabled on test computer.
- Tamper Data is enabled on test computer.
- Testing must be done with Firefox browser.

Evaluation

Participants should select a flow on any module of NextStar:

- **Design test case**
- **Download and set penetration testing tools.**
- **Execute Penetration testing**
- **Presentation a report results.**