ESPECIFICACIÓN DE REQUERIMIENTOS

para

Protocolo GuarDog

Versión 1.0

Autores:

Fernando Sebastián Silva Miramontes
Alejandro Pérez González
Sebastián Morales Martín
Jan Brandon Rivera Medina

Campus Estado de México

Abril 28, 2019

Tabla de versiones	
Versión 1.0	• **Lanzamiento del prototipo**

Introducción

1.1 Proposito

En este documento se informa sobre el protocolo de seguridad “Protocolo GuarDog”, o PGD por sus siglas, la información proporcionada deberá aclarar dudas del funcionamiento del protocolo y su ciclo de vida actual

1.2 Alcance

GuarDog es un protocolo de seguridad cuyo propósito es disminuir o erradicar las extorsiones por internet conocidas como phishing. El protocolo registrará las paginas en listas blancas y negras, en las cuales se encontrarán los dominios que tienen autorizado el uso del nombre de la empresa y los que no respectivamente; en cuanto se encuentra un nuevo dominio que usa el nombre de la empresa y no está autorizado para ello, se muestra un mensaje de advertencia que sugiere no compartir tus datos bancarios y personales en el sitio de ser pedidos por el mismo. El protocolo solo requiere una primera instalación como plug-in y estará habilitado en Google Chrome.

1.3 Definiciones y Acrónimos

Phishing	Técnica de ingeniería social utilizada por los delincuentes para obtener información confidencial como nombres de usuario, contraseñas y detalles de tarjetas de crédito haciéndose pasar por una comunicación confiable y legítima.
Plug-in	Es aquella aplicación que, en un programa informático, añade una funcionalidad adicional o una nueva característica al software.

Dominio	El nombre único que identifica a un sitio web en internet.
Requerimientos funcionales	Enunciado sobre los servicios que el sistema debe proveer, las entradas que acepta y sus salidas.
Requerimientos no funcionales	Limitaciones sobre el comportamiento del sistema, ej: tiempo de respuesta, espacio de memoria.
Modelo de desarrollo	Modelo en el que se basa el desarrollo de software, define cómo debe actuar los desarrolladores durante la creación de un software.

1.5 Visión general del producto.

1.5.1 Introducción: Se explican el propósito del documento
1.5.2 Descripción general
1.5.3 Requerimientos Específicos
1.5.4 Requerimientos de interfaces externas
1.5.5 Otros requerimientos no funcionales
1.5.6 Esquemas

Descripción general

2.1 Perspectiva del producto

GuardDog es un plug-in el cual actúa de manera independiente dentro del navegador de internet Google Chrome. En el momento que es instalado comienza su funcionamiento para defender tus datos personales por medio del análisis de los dominios que contengan “banorte.com”, dentro de ellas serán analizadas para determinar si pertenecen a la empresa, de no ser de ella, se muestra una ventana advirtiendo que la página es phishing y procede a ser agregada a una lista negra la cual se reportará para su cierre.

2.2 Funciones del producto

El usuario instalará el plug-in como extensión de Google Chrome, en ese momento termina la interacción con el usuario y comienza la funcionalidad principal de el protocolo, este analiza las páginas diferenciando las falsas de las reales, advirtiendo al usuario con las malignas. El protocolo cuenta con una lista blanca con las páginas permitidas, y una lista negra con las páginas maliciosas, la cual crecerá conforme se van agregando dominios falsos.

2.3 Características del usuario

PGD tiene como objetivo el mercado de personas de edad avanzada (55 en adelante) las cuales son más vulnerables a la ingeniería social y por ende mas propensas a caer en este tipo de estafas.

2.4 Restricciones

PGD debe cumplir con la ley de protección de datos de México y los lugares donde opera. El protocolo es de aplicación voluntaria.

2.5 Suposiciones y dependencias

Puesto a que PGD es un protocolo de seguridad, es de esperar que se intente penetrar para sobrepasarlo o inutilizarlo, por ello se revisará la integridad del protocolo con frecuencia para asegurar que se mantenga firme contra los posibles ataques que pueda sufrir. Se espera algún falso positivo o viceversa dentro del protocolo, es imperativo corregir estas discrepancias en el momento que sean detectadas para el uso óptimo del plug-in.

2.6 Modelo de desarrollo y fundamentación

El protocolo será desarrollado con un método de Programación Extrema, debido que el tiempo disponible hasta la entrega es mínimo y la retroalimentación con el cliente es después de acabado el protocolo. Creemos firmemente que la opción mas efectiva es que el protocolo tenga tan poca interacción con el usuario como sea posible, para así evitar que se comprometa su seguridad por alguna manipulación de los parámetros de manera accidental.

3. Requerimientos específicos

3.1 Requerimientos Funcionales

3.1.1: El usuario debe tener una interacción mínima con el protocolo.
3.1.2: El protocolo debe actualizar su lista negra una vez que se encuentra una página maliciosa.
3.1.3: El protocolo debe verificar primero si el dominio accesado no está en su lista negra, en caso de estar en esta, se alerta al usuario sobre la estafa.
3.1.4: El protocolo debe verificar el dominio accesado contra la lista blanca. En caso de estar en ella, se permite el acceso, de lo contrario, si no estaba previamente en la lista negra, se agrega a ella y se alerta sobre una posible página phishing.
3.1.5: Una vez que el protocolo agrega un dominio a la lista negra este no debe ser retirado de ella hasta que el dominio anteriormente mencionado sea cerrado.

3.2 Requerimientos NO Funcionales

3.2.1: El protocolo debe actualizar su lista negra de manera mensual, reportando los dominios que se encuentran en ella con una semana de anticipo de actualizar y borrándose completamente con la actualización.
3.2.2: El protocolo debe ser capaz de ampliar sus bases de datos de lista blanca para alojar más dominios validados por el protocolo mismo.
3.2.3: El protocolo debe ser actualizado con regularidad para poder reparar cualquier error que pueda aparecer, generalmente esto ocurre de manera mensual.

4. Requerimientos de Interfaces Externas

4.1: El protocolo debe de funcionar principalmente en segundo plano.

5. Otros requerimientos no funcionales

5.1 Seguridad

5.1.1: El protocolo no debe ser invasivo a las páginas ajenas a las maliciosas.
5.1.2: El protocolo no registrará la información personal del usuario, solamente los dominios maliciosos.

5.2 Requerimientos de Desempeño

5.2.1: El plug-in tendrá un espacio de almacenamiento dinámico no menor a 1 Mb y sin límite máximo.
5.2.2: El protocolo se debe ejecutar cada vez que detecta una página maliciosa.

6. Pruebas

Caso de Prueba	Req. Funcional	Caso de Uso	Descripción	Entradas	Salidas esperadas	Aceptado
TA01	**3.1.2**	**CU01**	Se encuentra una página maliciosa después de analizar la lista blanca.	URL de la página phishing.	Advertencia sobre la estafa. Se agrega el URL a la lista negra de páginas, para ser reportada posteriormente.	N/A
TA02	**3.1.3**	**CU01**	Se encuentra una página maliciosa al leer la lista negra.	URL de la página phishing.	Advertencia sobre la posible estafa.	N/A
TA03	**3.1.4**	**CU01**	Se detecta una página genuina con la lista blanca.	URL de la página verificada.	Acceso sin advertencias a la página.	N/A
TA04	**3.1.5**	**CU01**	Está a una semana de la fecha de borrado de lista negra.	Lista negra.	Reporte de los elementos en la lista negra para su eliminación de los dominios.	N/A
TA08	**3.2.2**	**CU01**	El cliente decide ampliar su cobertura de servicios.	Datos verificados sobre el nuevo dominio.	N/A	N/A