INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

Trabajo Terminal I.

Autentificación Mediante Chaffing And Winnowing En El Protocolo HTTP

2018-B003.

Integrantes:
Carrillo Fernández Jerry
Blancas Pérez Bryan Israel
Morales González Diego Arturo
Paredes Hernández Pedro Antonio

Directores:
Moreno Cervantes Axel Ernesto
Díaz Santiago Sandra

Índice

Índice de figuras.

Índice de cuadros.

A. Introducción.

A.1. Planteamiento del problema.

En la actualidad todos los usuarios de internet necesitan guardar contraseñas para sus distintas cuentas en las diferentes páginas web en las que ingresa ya que recordarlas es un problema que avanza constantemente. El uso de estas contraseñas son utilizadas principalmente en correos electrónicos y redes sociales por lo que el robo de las mismas puede poner en riesgo la seguridad del usuario, así como también, existe la tediosa tarea de ingresar usuario y contraseña en cada sesion. Las contraseñas son comúnmente utilizadas para el inicio de sesion y existen diferentes métodos de autentificación para dicho inicio como lo son biométricos. En nuestro proyecto nos enfocaremos más en el uso de text password en donde se autentificará el usuario por medio de una extensión de Google Chrome. Con ayuda de esta extensión resolveremos los problemas comentados anteriormente, dando así comodidad y seguridad al usuario que habilite la extensión.

A.2. Justificación.

Los usuarios deben de guardar las contraseñas en medios fisicos o digitales y perderlos presenta un grave problema de seguridad. La gran mayoria de servicios web han implementado una solución la cual es recordar tu usuario y contraseña para que se pueda automaticamente acceder al servicio. Dicha solucion presenta cierta vulnerabilidad ya que los archivos donde se guarda la información se puede copiar y con ello replicarlo a otra computadora. En el cuadro No.1, se muestra una tabla donde se comparan los diferentes métodos de autentificación basándose en la simplicidad de su aplicación para el usuario (extraída del artículo Comparison of Authentication Methods on Web Resources). Donde: 1 – Bajo desempeño, 2 - Medio desempeño y 3 – Alto desempeño.

	Recordar	Otros dispositivos	Acciones	Facilidad	Tiempo	Errores	Recuperación
Contraseñas	1	3	2	3	3	2	3
Otros recursos	2	3	3	3	3	3	2
Contraseñas gráficas	1	1	2	3	3	2	3
Contraseñas dinamicas	1	3	2	2	3	2	2
Tokens	3	1	1	2	2	3	1
Multivariación	1	1	1	3	2	2	1
Cryptografía	3	1	1	1	1	2	1
Biométricos	3	3	2	3	2	2	1

Cuadro 1: Comparación de la aplicación en los distintos métodos de autentificación

La tabla anterior concentra las siguientes caracteristicas:

- Recordar: Hace referencia a que tan complicado es que un usuario se acuerde de los datos necesarios para la autentificación.
- Otros dispositivos: El usuario usa una entidad externa para facilitar su autentificación.
- Acciones: Hace referencia a que tantas acciones adicionales se deben de realizar para autentificarse.
- Facilidad: Simplicidad de tecnología.
- Tiempo: Cantidad de recursos temporales que consume el método de autentificación.
- Errores: Posibles errores durante la autentificación.

- Recuperación: Denota la dificultad de recuperar la clave de acceso en caso de pérdida.

En el cuadro No.2 se muestra una tabla comparativa del nivel de seguridad en los distintos métodos de autentificación, donde 1 - baja seguridad, 2 – media seguridad y 3 – alta seguridad.

	Ataque por fuerza bruta	Observación	Hackeo indirecto	Phishing
Contraseñas	1	1	1	1
Otros recursos	2	2	3	3
Contraseñas gráficas	1	1	2	2
Contraseñas dinamicas	2	3	2	2
Tokens	3	3	3	3
Multivariación	1	1	3	3
Cryptografía	3	3	3	3
Biométricos	3	3	1	1

Cuadro 2: Comparación de la seguridad en los dstintos métodos de autentificación

La tabla se enfoca principalmente en los siguientes problemas de seguridad:

- Ataque por fuerza bruta: Se descifra el método de autentificación con una gran cantidad de intentos, usualmente generados por un programa.
- Observación: Cuando se intenta ver directamente los datos necesarios para la autentificación desde una distancia cercana hasta incluso usando binoculares, cámaras o algún otro dispositivo.
- Hackeo indirecto: El usuario confía sus datos del método de autentificación a terceros quienes pueden ser atacados.
- Phishing: Hace referencia a programas que se hacen pasar por entidades confiables para interceptar los datos que desean.

A.3. Objetivos.

A.4. Metodología.

A.5. Estado del Arte.

B. Marco Teórico.

B.1. Formato a decidir.

C. Análisis.

C.1. Prototipo I.

C.1.1. Descripción.

En este prototipo se busca la creación de una extensión de Google Chrome, que sea capaz de interceptar una petición HTTP hecha por el navegador.

C.1.2. Herramientas a usar.

Para el desarrollo de software de este prototipo, ocuparemos las siguientes tecnologías debido a que nos facilitan el desarrollo y nos proporcionan lo necesario para lograr nuestro objetivo para este prototipo:

JavaScript.

JavaScript es considerado como el lenguaje de programación de HTML y de la web. Es un lenguaje de programación fácil de usar y muy versátil para el hámbito de la comunicación en redes.

En el ambito del hardware utilizaremos los equipos de cómputo con los cuales contamos actualmente los integrantes, donde se especificaran a continuación:

Equipo de hardware utilizado.	
Nombre	Morales González Diego Arturo
Marca	Asus
Modelo	X550VC
Procesador	Intel Core i5
Tarjeta de video	NVidia GForce 720
Memoria RAM	12 GB
Disco duro	1TB

Equipo de hardware utilizado.	
Nombre	Carrillo Fernández Gerardo
Marca	HP
Modelo	Pavilion g4
Procesador	Intel Core i3
Tarjeta de video	Intel Sandybridge Mobile
Memoria RAM	6 GB
Disco duro	500GB

C.1.3. Estudio de requerimientos.

Requerimientos Funcionales.

PI_RF1. Interceptar petición HTTP. La extensión deberá interceptar la petición HTTP del navegador, en cuanto el usuario realice alguna a través del navegador.

PI_RF2. Deshabilitar extensión. El usuario podrá deshabilitar la extensión, para que ésta no vigile su actividad en el navegador.

PI_RF3. Habilitar extensión. El usuario podrá habilitar la extensión, para que ésta vigile constantemente cuando éste realice una petición HTTP.

Requerimientos no Funcionales.

PI_RNF1. Plataforma de implementación. La extensión será implementada en el navegador Google Chrome.

PI_RNF2. Versión del navegador La extensión funcionará a partir de la versión 65.0.3325.181.

PI_RNF3. Tecnologías para la interfaz de usuario Para el sistema se hará uso de HTML, JavaScript, CSS, JSON.
[1]

[1]Checar si es necesario especificar que debe estar habilitado JavaScript y si sería Funcional o No funcional

C.1.4. Reglas del negocio.

PI_RN1. Confidencialidad de la actividad web. En cuando el cliente lo indique por medio de la IU, la extensión deberá dejar de vigilar la actividad que el usuario realice en el navegador.

D. Desarrollo.

D.1. Prototipo I.

D.1.1. Diagrama de casos de uso.

Diagrama de casos de uso general para el prototipo I.

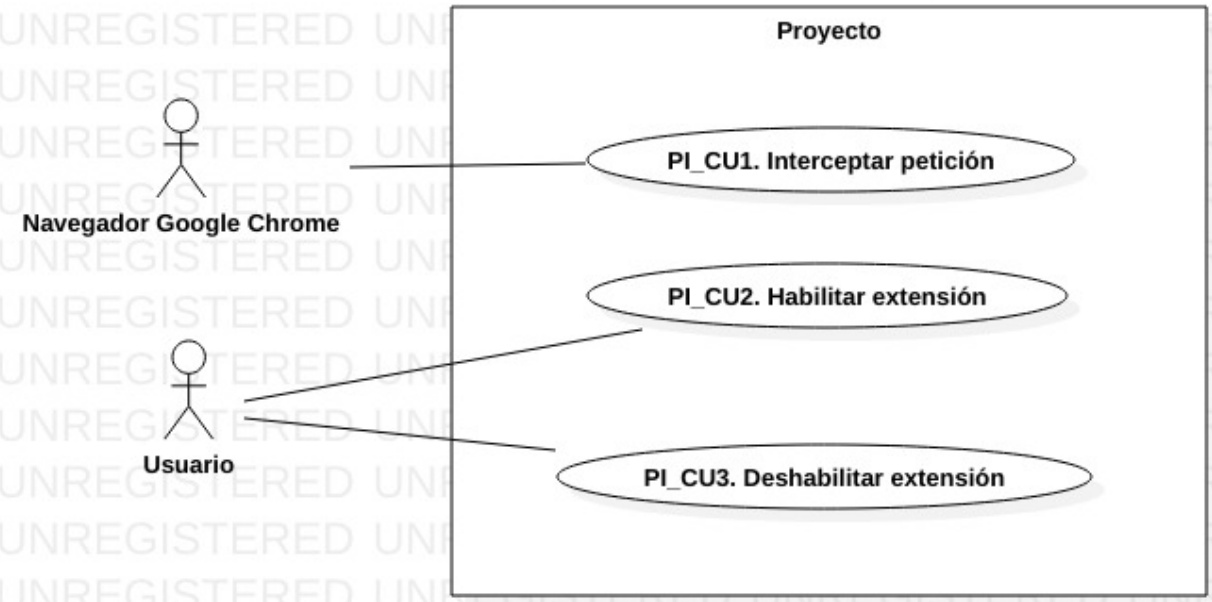

Figura 1: Diagrama de casos de uso.

D.1.2. Descripción de casos de uso.

[2]

Caso de uso: PI_CU1. Interceptar petición.	
Concepto	Descripción
Actor	Navegador de Google Chrome
Propósito	Este caso de uso permite al
Entradas	AL
Salidas	AL
Pre-condiciones	AL
Post-condiciones	AL
Reglas del negocio	AL
Errores	AL

Cuadro 3: Descripción CU: PI_CU1

Trayectoria Principal.

Fin de la Trayectoria Principal.

[2]EL actor es el navegador?

Caso de uso: PI_CU2. Habilitar extensión.	
Concepto	Descripción
Actor	AF
Propósito	AF
Entradas	AL
Salidas	AL
Pre-condiciones	AL
Post-condiciones	AL
Reglas del negocio	AL
Errores	AL

Cuadro 4: Descripción CU: PI_CU2

Trayectoria Principal.

Fin de la Trayectoria Principal.

D.1.3. Diagrama de flujo.

D.1.4. Flujo de datos.

D.1.5. Diagrama de clases.

D.1.6. Diagrama de secuencia.

D.1.7. Interfaz de usuario.

D.1.8. Requisitos de diseño.