

Práctica de Evaluación de Cortafuegos personales

Objetivo

El objetivo de esta práctica es que el alumno aprenda a configurar y evaluar cuál es la mejor opción de producto en relación a los cortafuegos personales.

Descripción de la práctica

La práctica consistirá en instalar y configurar distintos productos disponibles en el mercado para implementar la función de Cortafuegos o firewall personal.

En la red existen diferentes peligros que pueden perjudicar a nuestro equipo cuando lo conectamos a la misma: los troyanos que pueden instalarse en él y robarnos información o administrarlo remotamente, los escaneos a los que nos vemos enfrentados cada vez que navegamos que buscan una respuesta o una puerta abierta en nuestra computadora, etc.

Los Cortafuegos son utilidades que nos ayudan en estos casos. El funcionamiento de la mayoría de estos productos consiste en avisarnos de que una aplicación intenta conectarse a Internet, y pone en nuestras manos la decisión de permitirlo o no. De esa forma, somos conscientes de lo que está pasando en nuestro PC cuando estamos conectados a la red. El funcionamiento es muy simple: controla los datos que fluyen de nuestro PC hacia la red y permite a los usuarios decidir qué aplicaciones pueden acceder el Internet. También puede controlar el sentido contrario del flujo de la información, es decir, las conexiones desde la red a nuestro PC, bloqueando así accesos no deseados a nuestro equipo.

Al estar dirigidos a usuarios que pueden no tener conocimientos técnicos, la configuración de este tipo de cortafuegos es mucho más simple que la de los cortafuegos corporativos.

1. Instalación y configuración de Cortafuegos.

Los productos seleccionados para realizar la práctica son:

- Sygate Personal Firewall
- ZoneAlarm
- Cortafuegos de Windows XP

Los pasos a seguir son los siguientes:

1. Descargar los productos.
 - Sygate: <http://www.tucows.com/preview/213160>
 - ZoneAlarm: <http://zonealarm.uptodown.com/>
2. Instalarlos separadamente (no instalar 2 productos simultáneamente, tener siempre uno sólo en ejecución). En nuestro caso, empezaremos con Sygate.
3. Una vez instalado realizar las siguientes configuraciones:
 - Permitir navegar por Internet (puerto 80) con Firefox pero hacer que pregunte cada vez que intente acceder con Internet Explorer.
 - Permitir usar el correo electrónico (puertos 25 y 110). Probarlo, por ejemplo, con Outlook.
 - No permitir que Microsoft Office acceda a Internet (por ejemplo, al hacer clic en un enlace en Word, no debe permitir que se abra la página)
 - Crear una regla para que no puedas hacer un ping a uno de tus compañeros. Para ello, pide la dirección IP a uno de ellos (ejecutar el comando "ipconfig" desde la ventana de MS-DOS) y prueba que al ejecutar el comando "ping <dirección IP>" desde la ventana de MS-DOS obtienes respuesta del PC de tu compañero. Luego, crea la regla y vuelve a probar. Cuando la regla esté bien configurada, deberás obtener el mensaje "Tiempo de espera agotado para esta solicitud" al ejecutar el ping.

Nota: Probar que todas las reglas funcionan ejecutando las aplicaciones pertinentes en cada caso. Es recomendable probar las aplicaciones antes de configurar la regla (comprobar que se puede acceder) y después (no se puede acceder).

4. Desinstalar Sygate, instalar ZoneAlarm y configurar las reglas de nuevo (paso 3).
5. Por último, desinstalar ZoneAlarm y configurar las reglas para el cortafuegos de XP.

2. Evaluación de Cortafuegos.

En el momento de realizar la adquisición de un producto, debemos asegurarnos de elegir la mejor opción. Esta mejor opción corresponde a una serie de criterios que los usuarios deben definir y que dependerán del producto a evaluar y del tipo de usuario que define dichos requerimientos. Además estos criterios deben poder medirse, ya que necesitamos de alguna manera poder comparar los diferentes productos y, para ello, vamos a medir ciertas características y a tomar una decisión según las mediciones obtenidas. Estas mediciones pueden ser cualitativas o cuantitativas. En Ingeniería del Software se define una medida cualitativa como aquella para la que no obtenemos un valor numérico. Por ejemplo, “*tiene interfaz gráfico*” es un criterio cualitativo ya que la respuesta al mismo será *sí* o *no*. Una medida cuantitativa es aquella para la que sí podemos obtener un valor numérico, por ejemplo, “*tiempo necesario para procesar una regla*”. Es importante destacar que no se pueden medir todas las características de un producto por razones de tiempo y rendimiento y también, porque es muy probable que ningún producto cumpla las expectativas de todos los usuarios y, por tanto, que una evaluación así no tenga éxito. Por ello, es necesario elegir cuáles son las características más importantes a evaluar y con qué grado de importancia en relación al total.

Vamos ahora a definir los criterios que, según la experiencia previa en la instalación y configuración de cortafuegos obtenida tras la realización de la primera parte de la práctica, creemos que son importantes a la hora de seleccionar un producto de cortafuegos. Luego aplicaremos dichos criterios a los productos anteriores para verificar cuál de ellos tiene una mayor calidad con respecto al resto. Para ello, realiza los siguientes pasos:

1. Define al menos 5 criterios que sirvan para evaluar Cortafuegos y que no aparezcan en este documento. Al final tienes un anexo en el que se definen las clases de criterios que suelen tenerse en cuenta en Ingeniería del Software para el software en general con ejemplos particulares aplicados a cortafuegos. Estas categorías pueden ayudarte a identificar dichos criterios.

2. Elabora una tabla con los criterios (los definidos por ti más los que te parezcan importantes de los que aparecen en este documento) y los productos que has probado como la que aparece a continuación:

| | Sygate | ZoneAlarm | WXP |
|-------------|--------|-----------|-----|
| Criterio 1 | | | |
| Criterio 2 | | | |
| Criterio 3 | | | |
| Criterio 4 | | | |
| Criterio 5 | | | |
| Criterio 6 | | | |
| Criterio 7 | | | |
| Criterio 8 | | | |
| Criterio 9 | | | |
| Criterio 10 | | | |
| Criterio 11 | | | |
| Criterio 12 | | | |
| Criterio 13 | | | |
| Criterio 14 | | | |

Mide cada uno de los criterios y pon los resultados en la tabla para poder comparar los productos.

3. ¿Con cuál de ellos te quedarías según los resultados de la tabla? ¿por qué?

Anexo I. Características, subcaracterísticas y atributos de calidad (ISO/IEC 9126)

Todo producto software debe cumplir, en mayor o menor medida dependiendo de una serie de factores como, por ejemplo, el tipo de producto o el entorno en el que se ejecutará, una serie de características. A continuación se explican brevemente las características de alto nivel comúnmente aceptadas en el área de calidad del software:

- **Funcionalidad.** Capacidad de satisfacer las necesidades del usuario.
- **Fiabilidad.** Capacidad para responder bajo las condiciones definidas durante un intervalo de tiempo dado.
- **Usabilidad.** Conjunto de características que influyen en el esfuerzo requerido para la comprensión, aprendizaje y uso por parte de un conjunto de usuarios dado.
- **Eficiencia.** Capacidad para proporcionar el rendimiento apropiado.
- **Mantenimiento.** Esfuerzo requerido para ser modificado en relación a correcciones, mejoras o adaptación del software.
- **Portabilidad.** Conjunto de características que determinan la capacidad del software para ser transferido de un entorno de operación a otro.

Cortafuegos personales

Las características citadas arriba se subdividen en subcaracterísticas que son propiedades más específicas de calidad. Para poder evaluar la calidad del un producto, tenemos que definir “atributos” relacionados con estas subcaracterísticas. Los atributos son propiedades que podemos medir, ya sea cualitativamente (por ejemplo, decir si un producto permite o no hacer una acción concreta) o ya sean cuantitativamente, es decir, dándoles un valor numérico (por ejemplo, tiempo que tarda un producto en procesar cierta función).

A continuación se exponen una serie de subcaracterísticas definidas para cada una de las categorías de anteriores. Dichas características y subcaracterísticas deben personalizarse para cada producto, es decir, no tienen por qué cumplirlas todas cualquier producto, ni tampoco tienen por qué aparecer todas las que habría que medir en cierto tipo de productos en esta lista sino que pueden añadirse otras que se consideren

apropiadas para el tipo de producto en cuestión. Además, se dan algunos ejemplos de atributos para el caso específico de Cortafuegos para cada subcaracterística con el fin de clarificar la definición de la misma.

- Funcionalidad:

- *Adecuación*: la capacidad para proporcionar un conjunto apropiado de funciones para tareas específicas de los usuarios.
 - Ejemplo: Permitir crear reglas avanzadas en las que puedan definirse todos los parámetros.
- *Exactitud*: la capacidad para proporcionar los resultados o efectos correctos y con el grado de precisión acordado.
 - Ejemplo: Bloquear un 100% de las aplicaciones configuradas.
- *Interoperabilidad*: la capacidad para interactuar con uno o más sistemas especificados.
 - Ejemplo: Compatible con los sistemas operativos W2K, WXP, W2003, Linux.
- *Seguridad*: referido a la capacidad para proteger la información y los datos.
 - Ejemplo:
 - Necesario permisos de administración en el sistema operativo para la modificación de las reglas configuradas en el cortafuegos.
 - No es posible terminar la aplicación sin privilegios administrativos (administrador o root).

- Fiabilidad:

- *Madurez*: indica la frecuencia con que ocurren los fallos (la capacidad para evitar fallos provocados por errores en el software).
 - Ejemplo: Número de errores en ejecución
- *Tolerancia a fallos*: grado en que el sistema mantiene un nivel de respuesta (aunque sea parcial) ante fallos del sistema o interfaces.

- Ejemplo: En caso de fallo de parte del software, seguir filtrando el tráfico según las reglas establecidas.
- *Capacidad de Recuperación*: la capacidad del software para restablecer su nivel de respuesta después de un fallo recuperando los procesos o datos afectados directamente en el momento del error.
 - Ejemplo:
 - En caso de fallo no perder la configuración (reglas).
 - Tiempo de recuperación de la protección (procesamiento de reglas).

- Usabilidad:

- *Comprensibilidad*: la capacidad para permitir al usuario que entienda si el software es adecuado, y como debe utilizarse para determinadas tareas y bajo ciertas condiciones de uso.
 - Ejemplos:
 - Consistencia en el orden de colocación de las opciones en la aplicación.
 - Ejecución de tarea sin ayuda
 - Explicación clara de parámetros de entrada (por ejemplo, explicación de los parámetros que hay que dar para la configuración de reglas)
 - Facilidad para entender la secuencia de actividades a realizar para completar una tarea.
 - Mensajes breves y lenguaje sencillo
 - El texto y las figuras facilitan la comprensión
- *Facilidad de aprendizaje*: características que influyen en el esfuerzo del usuario para aprender su aplicación
 - Ejemplos:

- Entorno familiar (menús en la parte superior en el orden esperado: archivo, herramientas, ..., ayuda, inmediatamente debajo botones, en la parte superior derecha guión para minimizar, rectángulos superpuestos para restaurar y aspas para cerrar, etc.)
 - Ayuda
 - Tutoriales
- *Operabilidad*: capacidad para permitir que el usuario ejecute y controle el software.
 - Ejemplos:
 - Auto-descripción de las opciones a las que el usuario puede acceder.
 - Opciones de personalización (por ejemplo, atajos para usuarios experimentados)
 - Uso de asistentes o ventana auto-explicativas para la configuración de reglas.
 - Configuración de reglas en línea: cuando una aplicación intenta acceder a la red, el cortafuegos pregunta si se quiere dejar su acceso sólo en esa ocasión y, por tanto, volverá a preguntar la próxima vez, siempre o nunca más.
 - Modificación de reglas a través de asistentes o ventanas auto-explicativas.
- *Atracción*: características del software que influyen en la satisfacción de los deseos y preferencias del usuario haciendo que el entorno del software le resulte.
 - Ejemplo:
 - Uso de interfaz gráfica (GUI)
 - Posibilidad de personalización de la interfaz de acuerdo a un perfil
 - Combinación de texto y gráficos

- Combinación de colores/fondos
- Estéticamente agradable

- Eficiencia:

- *Comportamiento temporal:* la capacidad para proporcionar tiempos de respuesta y de procesamiento apropiados cuando realiza sus funciones bajo condiciones determinadas.
 - Ejemplo: Eficiencia en el procesamiento de reglas
- *Utilización de recursos:* la capacidad para utilizar cantidades y tipos de recursos apropiados cuando el software realiza su función bajo determinadas condiciones.
 - Ejemplo: Bajo consumo de memoria y procesador.

- Mantenibilidad:

- *Actualización:* Facilidad para la instalación de parches, actualizaciones o mejoras proporcionadas por el proveedor de software.
 - Ejemplo: Dispone de facilidades para la detección e instalación de parches y mejoras del producto.
- *Estabilidad:* Capacidad de evitar los efectos inesperados de las modificaciones.
 - Ejemplo: Posibilidad de deshacer cambios y volver al estado anterior a la instalación de los mismos.

- Portabilidad:

- *Facilidad de instalación:* características del software que influyen en el esfuerzo requerido para instalar el software en un entorno especificado.
 - Ejemplo: Instalación mediante asistente guiado.