UNIVERSITAT POLITÈCNICA DE VALÈNCIA

etsinf
Escola Tècnica Superior d'Enginyeria Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Seguridad en el DataCenter
Antivirus Web

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: José Luis Lozano Barrachina

Tutor: Germán Vidal Oriola

2014-2015

Índice

RESUMEN INTRODUCTORIO

Puesta en escena: ¿Por qué el Antivirus Web?

Hoy en día todos tenemos algún que otro Antivirus activo en nuestros ordenadores, Tablets, Smartphones…

Y esto nos protege relativamente de la mayoría de amenazas a nivel de sistema, sobretodo de las clásicas vulnerabilidades, troyanos, gusanos, virus, etc… Previamente documentados en las bases de datos de estos Antivirus.

Las bases de la seguridad informática han sido asentadas en 3 aspectos que se han considerado importantes e imprescindibles a la hora de poder afirmar que algo es “Seguro”.

Estas bases han sido la “Confidencialidad”, “Integridad”, y “Disponibilidad” para poder ofrecer cierta “Fiabilidad” y así proceder a afirmar que nuestro sistema es seguro.

Pero a día de hoy la mayoría de amenazas han subido de nivel, dejando de actuar principalmente en los sistemas físicos y así empezar a actuar a nivel Web.

Los Antivirus actuales generalmente se centran en examinar el código X que está alojado en nuestro disco duro, buscando coincidencias con una base de datos bien documentada de las amenazas existentes de las que se tiene constancia, y reflejando las coincidentes en una lista de amenazas encontradas para que el usuario posteriormente proceda a actuar sobre estas, generalmente de tres maneras “Borrar”, “Reparar” o “Cuarentena”.

Generalmente se procede a intentar repara el archivo infectado mediante técnicas previamente validadas por el antivirus en casos previos, muchas veces esto no es posible ya sea porque es nuevo, o porque aún no se conoce la solución.

Es el momento de plantearse si es importante para nosotros el archivo o no, procediendo a bloquearlo en la cuarentena o borrándolo.Este tipo de barrera de seguridad es muy buena para los comunes USB Pen Drive, que nos prestaban y al conectarlos para copiarnos cierto archivo, como la memoria USB estaba infectada, la amenaza se pasaba, trasladaba, copiaba a nuestro sistema pretendiendo infectarlo.

Dónde los riesgos podrían ser menores, a lo sumo perder toda la información alojada en la memoria del ordenador.

Pero cuando estas amenazas trabajan a nivel Web, hablamos de casos mucho más preocupantes, dado que infectar una web es un medio perfecto para acceder a millones de máquinas que se descargan cierto archivo/imagen a la hora de cargar la web en sus navegadores.

Aparecen nuevas amenazas con la finalidad de obtener datos bancarios o de tarjetas, ya sea por suplantación de identidad o mediante otro sistema de hackeo.

Es en este contexto en el que toma verdadera importancia el Antivirus a nivel Web, y más aún si consideramos que cada vez es más rentable este negocio para este tipo de delincuentes, dado que cada vez más y más información es almacenada a nivel digital y por tanto almacenada en DataCenters contratados dónde impera la seguridad.

Puesta en escena: Finalidad del Catálogo de Amenazas

El catálogo será una de las partes programadas por el alumno en su totalidad.

Este catálogo pretende recuperar información previamente almacenada sobre las amenazas web existentes y las más utilizadas, para así ayudar a la persona encargada de combatir y eliminar las amenazas que no puedan ser automáticamente detectadas y eliminadas.

Dado que a nivel web cada amenaza es distinta y "personalizada" es muy difícil automatizar el proceso de Búsqueda y Destrucción por lo que habitualmente a pesar de que la detección y búsqueda resulta satisfactoria, un experto en seguridad deberá de entrar a mirar el código y buscar código malicioso, ya sea por vulnerabilidades de versión, vulnerabilidades a la hora de usar algunas funciones no recomendadas o por código ofuscado y malintencionado.

Por lo que este catálogo Web con acceso a una Base de Datos privada y propia, pretende ofrecer una ayuda teórica y práctica a la hora de informarse para proceder a la eliminación de "X" tipo de amenaza en "Y" ámbito y "Z" entorno.

SEGURIDAD INFORMÁTICA

¿Que es la Seguridad Informática?

La seguridad informática comprende software, hardware y todo lo signifique un riesgo en caso de que la información se extraviara.

También se considera un área de la informática como tal, enfocada en la protección de la infraestructura computacional y todo lo que tenga algún tipo de relación con esta, especialmente lo que llamamos información contenida.

Cabe destacar que a pesar de tener puntos comunes no tienen el mismo significado los conceptos “Seguridad Informática” y “Seguridad de la Información”.

La seguridad informática tiene como apartado y meta garantizar la seguridad de la información, entre otros muchos objetivos.

Es la disciplina que se ocupa de diseñar las normas, métodos y técnicas destinadas a conseguir un sistema de información seguro y fiable.

Referencias Bibliográficas: [1]

Objetivos de la Seguridad Informática

Debe establecer los horarios de funcionamiento, las restricciones, autorizaciones o denegaciones a los usuarios en los accesos a la información.

Diseñar planes de emergencia y protocolos, permitiendo así un buen nivel de seguridad informática lo que minimizará el impacto en el desempeño de los trabajadores y de la organización en general.

Proteger los activos informáticos es la tarea más importante y el foco de la seguridad informática.

Los activos informáticos los conforman la infraestructura computacional, los usuarios y la información.

Referencias Bibliográficas: [1]

Activos Informáticos

La infraestructura Computacional

- Parte fundamental para el almacenamiento y gestión de la información, de igual modo lo es para el correcto funcionamiento de la organización.
- La seguridad informática deberá asegurar los equipos de tal manera que funcionen adecuadamente y deberá anticiparse a los problemas, tales como fallos, robos, desastres naturales, o cualquier otro factor que atente contra la infraestructura computacional que hay que proteger.

Los usuarios

- Personas que utilizan la estructura tecnológica de la organización y gestionan la información dentro de esta.
- La seguridad informática en este caso deberá proteger el sistema en general para que el uso inadecuado, malintencionado o irresponsable por parte de los usuarios no pueda afectar a la seguridad de la información y ésta permanezca segura e inalterada.
- Adicionalmente la información que manejan los usuarios o que almacenan debe de ser asegurada para que no sea vulnerable.

La información

- **Es el activo principal.**
- Se utiliza y se encuentra en la infraestructura computacional y es utilizada por los usuarios.
- Su protección debe de ser la prioridad principal en los objetivos a nivel de seguridad de la organización.

Referencias Bibliográficas: [1]

Causas de las Amenazas

A pesar de que hay infinidad de amenazas que atentan contra la seguridad de la información cada día, y que es imposible bloquear y asegurar en su totalidad cada una de ellas, las causas principales pueden bloquearse o al menos se pueden prever para intentar minimizar los daños.

Las causas más comunes son:

Usuarios

- Son la mayor causa de problemas ligados a la seguridad de los sistemas informáticos, sus acciones pueden llegar a causar graves problemas de seguridad.
- A pesar de que en la mayoría de los casos el motivo de estas acciones es por tener permisos sobredimensionados, es decir, no se les han restringido las acciones innecesarias o indebidas.

Programas Maliciosos

- Son aquellos programas destinados a perjudicar o a hacer un uso ilícito de los recursos del sistema.
- Se instalan, ya sea por falta de atención o maldad, en el ordenador objetivo, abriendo una puerta a intrusos o bien modifican datos del sistema, en algunos casos robando la información de este.
- Estos programas tienen muchas variaciones y dependiendo de estas y de sus características o finalidad, son clasificados en un grupo u otro, como lo son por ejemplo:
  - Virus informático, Gusano, Troyano, Spyware…
- Generalmente y a pesar de la infinidad de tipos, características y finalidad de estos programas maliciosos, se les suele llamar virus, pero su denominación genérica correcta sería “Malware”.

Errores de Programación

- En su mayoría, lo errores de programación considerados dentro del grupo de amenazas informáticas, es por motivo de explotación de terceros con finalidades ilegales o ilícitas.
- Estos errores pueden ser usados como exploits o puertas de entrada para almacenar otro tipo de amenazas.
- La actualización correcta y programada de parches con la finalidad de arreglar problemas previamente detectados tanto a nivel de Sistema Operativo como de programas y aplicaciones es la medida de seguridad más eficiente y por ello recomendada para evitar este tipo de amenazas.

Intrusos

- Son aquellas personas que consiguen acceder a los datos, o en su defecto programas automatizados.
- Ninguno de los dos está autorizado a leer, modificar, o copiar la información.
- Hay muchos tipos, como pueden ser
  - Crackers, defacers, hackers…

Siniestro

- Robos, incendios, inundaciones… Ya sea lo uno o lo otro, una mala manipulación deriva en la pérdida de material e información.

Personal Técnico Interno

- Los motivos más comunes por los que un Técnico de Sistemas, o un Administrador, etc… atenta contra la empresa, son:
  - Despidos, Problemas laborales, Disputas internas, Espionaje…

Fallos

- Ya sean de carácter electrónico o lógicos de los sistemas informáticos

Catástrofes Naturales

- Son las causas menos comunes, dependientes tanto de la zona de residencia como de la época del año, pueden llegar a generar las mayores pérdidas económicas si no se prevén, y pueden ser:
  - Terremotos, Inundaciones, Tormentas…

Referencias Bibliográficas: [1]

Análisis de Riesgos

El análisis de riesgos informáticos es un proceso muy importante para evitar problemas futuros, comprende la identificación de activos informáticos, sus vulnerabilidades y amenazas.

La probabilidad de que las amenaza en cuestión ocurra o aparezca, al igual que el impacto de cada una de ellas por separado o en combinación.

Con el fin de determinar exactamente los controles adecuados para en caso de ocurrencia de estas, evitarlas en su totalidad, o en caso de no poder, aceptarlas, disminuyendo el impacto de las mismas.

La necesidad de poder estimar la magnitud del impacto del riesgo a la que una empresa se encuentra expuesta va ampliando cada día.

Lo que es obvio dado que la explotación de un riesgo no controlado causaría daños o pérdidas financieras o administrativas a la empresa.

Para poder controlar estos riesgos son necesarios los controles de riesgos.

Si queremos que estos controles sean efectivos, deben ser implementados en conjunto formando una arquitectura de seguridad.

Consiguiendo así preservar las propiedades de **Confidencialidad**, **Integridad** y **Disponibilidad**, de las que hablaremos más abajo.

Es el momento de hablar de un documento en el que se muestran los elementos identificativos, la relación entre estos y los cálculos que se realizan sobre estos.

Este documento se llama Matriz de Riesgo y es el documento obtenido posteriormente a aplicar el proceso de análisis de riesgos previamente mencionado.

Si pretendemos lograr una correcta administración de los posibles riesgos es absolutamente imprescindible este tipo de análisis.

Referencias Bibliográficas: [1]

Análisis del Impacto al Negocio

Para un correcto establecimiento de prioridades, el sistema de gestión de incidentes ha de saber el valor de diferentes sistemas de información que posee la organización, para determinar cuál de ellos es potencialmente más probable que sufra algún tipo de incidente de seguridad.

Es probable que alguien de dentro de la organización sea el encargado de asignar un valor monetario o relativo a cada equipo, cada sistema y a cada tipo de información.

Los valores para el sistema son:

- **Confidencialidad de la información**
- **Integridad de las aplicaciones y de la información**
- **Disponibilidad del sistema**

Cada uno de los valores previamente mencionados es un sistema independiente del negocio.

Los incidentes individuales pueden variar en gran medida tanto a nivel del alcance como al de importancia.

Referencias Bibliográficas: [1]

Respaldo de la información

La información es con diferencia el activo más importante para las empresas, y esta puede verse afectada por diversos factores que hemos mencionado con anterioridad, tales como robos, incendios, ataques de Malware…

Uno de los problemas más importantes que quieren solucionar las empresas, es proteger de manera correcta y permanente la información crítica.

Las copias de seguridad o Backups, son generalmente la medida más eficiente para proteger los datos, y se debe de establecer una buena normativa y política de copias de seguridad.

Debe de tener dos tipos de copias de seguridad:

- **Completas**
  - La totalidad de los datos es almacenada por primera vez
- **Incrementales**
  - Sólo se copiará aquellos ficheros creados o modificados desde la última copia de seguridad realizada

Es muy importante para las empresas el realizar bien y de manera regular estas copias de seguridad, elaborando así un plan de Backups que dependerá del volumen de la información generada y la cantidad de equipos críticos.

Todo sistema de respaldo completo y seguro debe de contar con algunas características indispensables, como lo son:

- **Continuo**
  - Ha de ser transparente a los usuarios y debe ser completamente automático y continuo.
- **Seguro**
  - Previo al envío de información se han de cifrar los datos, comúnmente este paso de cifrado se realiza a nivel local.
- **Remoto**
  - Los datos se guardarán en dependencias alejadas de la empresa de tal manera que sean independientes
- **Mantener versiones anteriores de los datos**
  - Recovery > Sistema de recuperación de versiones anteriores, ya sean diarias, semanales y/o mensuales de los datos.

Cabe destacar el hecho de que hoy en día los sistemas de respaldo de información online, están aumentando respecto a otros modelos más clásicos, dada la independencia, seguridad y posibilidades que ofrecen.

La mayoría de estos sistemas tienen las máximas medidas de seguridad y disponibilidad respecto a los datos. Estos sistemas son perfectos para las empresas dado que así pueden crecer en volumen de información derivando la carga de las copias de seguridad al proveedor del servicio.

Referencias Bibliográficas: [1]

Tipos de Protección contra Virus

Si queremos evitar que los equipos y medios se infecten, tenemos que tenerlos bajo una vigilancia estricta y constante.

Hay que destacar las medidas de protección siguientes por su variedad:

Control del Software instalado

Sí tenemos instalado solamente el Software estrictamente necesario y de procedencia conocida, reducirá la probabilidad de aparición de amenazas.

Control de la red

Si establecemos un número de acceso a recursos de red en modo lectura, evitaremos y reduciremos la probabilidad de propagar las amenazas.

Reduciendo los permisos de los usuarios conseguimos un control parecido.

Es importante centralizar, controlar y monitorizar el acceso a internet para detectar así en las fases de recuperación de datos, exactamente cómo y por dónde se había introducido la amenaza.

Técnica parecida al **HoneyPot**.

Protección Física de acceso a la red

Redes Cableadas

- Los accesos físicos deben de estar protegidos y vigilados.
- Evitando tener puntos de red conectados a Switches.
- Listas de control de los equipos autorizados por su MAC.

Redes Inalámbricas

- Se deberán de controlar los cifrados, contraseñas compartidas…
- Medidas de contención de la emisión electromagnética para excluirla donde no sea necesaria.

Sanitización

- Ya sea de carácter físico/lógico, se deberá eliminar la información considerada sensible o confidencial, de sus usos locales y temporales.

Referencias Bibliográficas: [1]

Principios de la Seguridad Informática

Cuando hablamos de Seguridad Informática, quizá la definición que más rápido nos llega a la cabeza es la de los principios en los que se basa y rige:

Confidencialidad

Hablamos de privacidad de la información almacenada.

El sistema ha de estar protegido mediante herramientas y estrategias competentes para evitar invasiones y accesos por parte de personas o programas no autorizados.

Este principio es especialmente importante en sistemas distribuidos, dado que los usuarios, computadores y datos están interconectados.

Integridad

Nos referimos a la validez y consistencia de la información almacenada y procesada.

Todos los elementos del sistema posiblemente manipulan entre otros, datos compartidos, y la integridad exige que los procesos de actualización estén bien sincronizados y no se dupliquen para trabajar así con los mismos datos siempre.

Sobre todo en sistemas descentralizados la Integridad es realmente importante dado que diferentes usuarios, computadores y procesos comparten la misma información a todas horas.

Disponibilidad

Mencionar Disponibilidad es hablar de continuidad de acceso a la información almacenada

Se ha de enfocar en que el sistema sea permanente y por tanto este el máximo tiempo posible disponible, y para asegurarlo haremos uso de herramientas de seguridad informática.

Sin importar las condiciones de actividad o sobrecarga de un sistema, los usuarios han de poder acceder a este teniendo transparencia sobre la carga del sistema, o sobre el mantenimiento que se esté realizando sobre este.

Y así se podrá acceder a los datos con la frecuencia y la dedicación que se requieran, este principio es importante en sistemas informáticos cuyo compromiso con el usuario, es prestar servicio permanente, comúnmente llamado 24/7.

Referencias Bibliográficas: [1]

DATACENTER

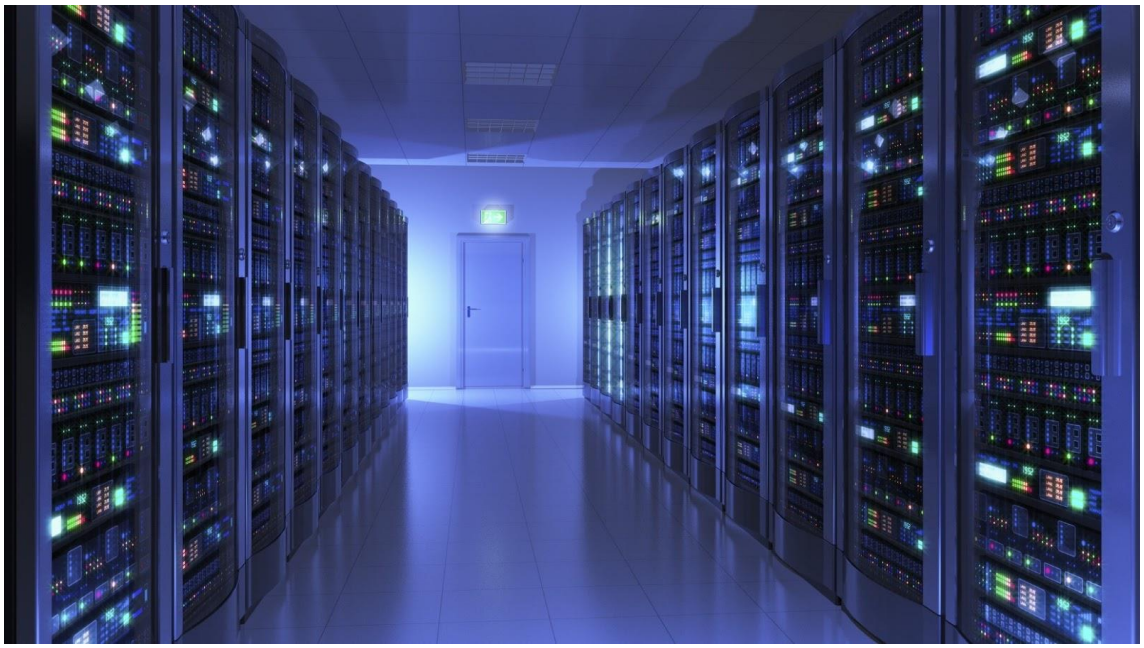
¿Qué es el DataCenter?

Es un Centro de Procesamiento de Datos (CPD), es decir, un espacio o lugar donde se concentran los recursos para procesar información de la organización.

Dependiendo del tamaño puede ser desde una sala de gran tamaño hasta un edificio.

Su tamaño suele ser grande dado el equipo electrónico que necesita, tanto para la computación de datos, como para mantenerse en así.

Suele estar situado en una zona climatizada a baja temperatura, comúnmente llamada “Nevera”.



Una Nevera adecuada para el DataCenter ha de tener controlada la temperatura, para evitar problemas y averías en los ordenadores y equipos de computación, que suelen presentar problemas en caso de sobrecalentamiento.

La temperatura adecuada varía desde los 20 Grados hasta un máximo de 23 Grados, según varios estudios de eficiencia energética.

Como es obvio, para mantener los equipos a pleno funcionamiento y refrigerados a la temperatura adecuada, es necesaria una gran carga energética, por lo que suelen tener por motivos de seguridad y disponibilidad, equipo energético de repuesto con la finalidad de no parar el servicio en caso de que haya algún tipo de fallo energético.

Referencias Bibliográficas: [2]

Estructura y Evolución

Si retrocedemos a la visión y estructura de los primero DataCenters, estos fueron diseñados siguiendo arquitecturas clásicas de redes de informática, que consistía en apilar equipos informáticos en armarios.

Hoy en día debido a la falta de espacio, algunas cosas han tenido que cambiar, y ahora los equipos tienen dimensiones que permiten aprovechar al máximo el volumen disponible de los racks, se les suele llamar “Equipos Enracables”, así se consigue en una misma unidad de espacio mayor densidad de equipos que de la manera habitual.

La abrumadora evolución de internet, y la necesidad de conectar estos equipos ha obligado a que en las empresas se adopten niveles altos de fiabilidad y de seguridad, con la finalidad de proteger los datos almacenados en los DataCenters.

Pero a pesar de tenerlos asegurados se pretende poder tener acceso a ellos y disponer de ellos sin interrupciones.

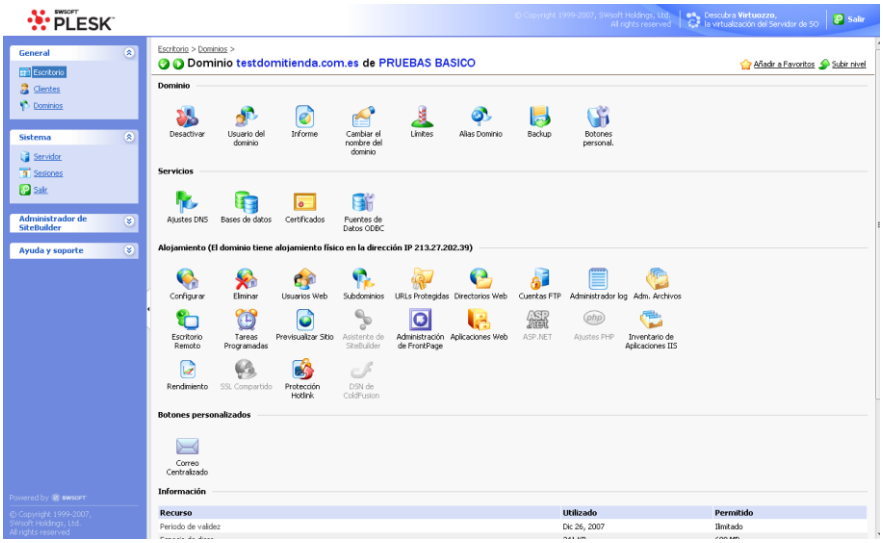
Por ello, dada la gran inversión económica que se necesita para el correcto despliegue, ha ido apareciendo otro nivel de DataCenter llamado Cloud DataCenter.

Es el mismo concepto pero aplicando la distancia entre el DataCenter y la organización, lo que además aumenta la seguridad.

Actualmente existen empresas expertas en el sector del almacenamiento web, o comúnmente llamado Hosting, que proporcionan un servicio profesional, encargándose tanto del mantenimiento como de la gestión de los DataCenters.

Estas empresas por lo general suelen ofrecer un mayor nivel de fiabilidad y seguridad, que tenerlo “apañado” en una esquina.

La gestión por parte de la empresa contratante de tal servicio se realiza a través de un control web, llamado panel de control.



Firewall Clásicos

¿Por qué no sirven?

El elemento que más publicidad ha tenido a nivel de seguridad para nuestros equipos, han sido los Firewalls y Antivirus que los incluyen.

Los Firewalls están diseñados con la finalidad de proteger una red interna contra los accesos no autorizados, este mecanismo se denomina Gateway.

En caso de que los paquetes de información, los que pasan varias inspecciones de seguridad, sean correctos y validados, pasarán por la puerta teniendo acceso al sistema, y los que no serán rechazados.

Los Firewalls son sistemas ubicados entre dos redes que ejercen políticas de seguridad previamente establecidas, encargados de proteger una red fiable de la que no lo es, como lo es Internet.

Cuando hablamos de Firewall tenemos varios tipos, en los que no voy a centrarme particularmente, dado que normalmente todo Firewall es una mezcla de varias técnicas y tipos, y ya sea en menor o mayor medida suele emplear todas de las que dispone.

Los tipos son:

- Filtrado de Paquetes
- Proxy-Gateways de Aplicaciones
- Dual-Homed Host
- Screened Host
- Screened Subnet
- Inspección de Paquetes

El principal problema del Firewall, es que no defiende de ataques provenientes del interior (Para eso tenemos Antivirus), ni puede ofrecer protección una vez el intruso traspasa el Firewall.

Para este último problema no hay solución, excepto que el intruso se quede a nivel local y el Antivirus se dé cuenta de ello.

La ventaja de un Firewall generalmente reside en que es económico con un alto nivel de utilidad y desempeño, y a su vez son prácticamente transparentes para los usuarios conectados a la red.

De igual modo tiene desventajas:

- No protege las capas superiores a nivel OSI.
- Las necesidades aplicativas son difíciles de traducir como filtros de protocolos y puertos.
- No son capaces de esconder la topología de redes privadas, por lo que exponen la red al mundo exterior.
- No soportan políticas de seguridad complejas como autentificación de usuarios y control de accesos con horarios prefijados.

Como podemos observar de una manera bastante sencilla el problema de un Firewall es poniendo un ejemplo:

Tenemos una aplicación llamada Google Chrome, que es el navegador que utilizamos, nuestro querido navegador intenta acceder a la red, pero el Firewall pregunta ¿Google Chrome es de fiar? ¿Le dejo acceder?, en ese momento nosotros contestamos, ¡ Sí !... Y todo va fantástico hasta que un día accedemos a una web de fotografías para ponernos un fondo de pantalla chulo, la cual al cargarla se ejecuta un SCRIPT mediante el cual se nos pregunta si queremos descargar el archivo fondo18237.png.

En ese momento no se nos ocurre… Porque es obvio que es una imagen en formato PNG y que estamos en una página de descarga de fotos, por lo que descargar fotos es lo más normal del mundo, pero al abrirla se ejecuta un SCRIPT oculto en la misma que da permisos e instala cierto tipo de Software con finalidades ocultas, que suele ser malware.

Ese tipo de conexión no la regula un Firewall, y en caso de que en lugar de un PC, el malware entra en un servidor con miles de webs, puede afectar a todas ellas, infectándolas, mutando para no ser encontrado y causando problemas allí por donde va.

Para bloquear y tener ejecutándose a tiempo real en cada servidor, un Antivirus que compruebe cada archivo modificado, que compruebe cada conexión, por ello se necesita un Antivirus Web.

La finalidad es tener un archivo introducido de incógnito que compruebe cada conexión entrante que el Firewall no bloquea y que en caso de que esa conexión descargue algún fichero, este Antivirus pueda eliminarlo antes de que ocurra nada. **Necesitamos un infiltrado**.

Referencias Bibliográficas: [1]

Malware Web

Definición y Características

Las amenazas web, más conocidas como malware web, es la versión del malware de toda la vida, los virus, gusanos, etcétera.

Pero que pueden atacarte cuando utilices internet.

MALWARE viene de juntar las palabras **MAL**+iciois soft+**WARE**.

Este tipo de Software tiene como finalidad introducirse en el sistema o máquina objetivo, y hacer daño sin el consentimiento de su propietario.

Es muy común que la gente generalice y le llame a todo tipo de malware, con la denominación común de "Virus".

Pero pueden diferenciarse en una gran variedad de categorías.

Referencias Bibliográficas: [4]

Tipos de Malware

Malware Infeccioso

Este tipo de malware se identifica por tu particular tipo de propagación.

Los dos más conocidos de los que hablaremos son el Virus y el Gusano.

Virus

Suelen reemplazar archivos ejecutables con cierto tipo de finalidad como instalar un programa benigno mediante el archivo **setup.exe**.

En lugar de ejecutar el código original del archivo, en su lugar ejecuta su propio código, generalmente para poder ser ejecutado se requieren permisos.

Pero el usuario al ser víctima de un engaño, dada la suplantación de identidad del archivo original, suele conceder los permisos requeridos.

Y todo este proceso de activación suele pasar desapercibido.

El código ejecutado se carga en la RAM del ordenador, y desde allí se busca un sitio para almacenarse en el disco duro. Por ello al desinstalar el programa que lo ha ejecutado, el Virus no se borra, dado que ya se ha movido y ocultado.

Lo normal es que este empiece a buscar los controles de varios servicios básicos del sistema operativo, infectándolos.

El nivel de peligrosidad, y el de impacto en consecuencia, se ve determinado por la finalidad con la que haya sido programado.

Desde borrar archivos PNG, a mover las carpetas de sitio, o simplemente ralentizar el ordenador, que es normalmente la situación inicial desde la cual los usuarios por norma general empiezan a pensar que algo se les ha metido en el ordenador.

Dado que este no funciona como debiera.

Gusano

La principal diferencia respecto al Virus que hemos visto antes, es la independencia de activación, dado que el Gusano no requiere que sea el usuario quien acciona y activa el malware, este se reproduce replicándose por el sistema y transmitiéndose.

Utilizan partes del sistema invisibles para el usuario para almacenarse, lo que complica en gran medida su detección.

Y el gran dilema viene cuando le sumamos a este amenaza tan particular, la posibilidad de replicarse y enviarse, además de por discos, USB, etc… Por Internet.

La propagación deja de ser exclusivamente a nivel local, y empieza a ser a nivel global.

Utilizan métodos como SMTP, IRC o el famoso P2P para pasar de un ordenador a otro, hasta que llegan a algún servidor donde pueden saltar a miles o millones en lugar de ser uno a uno.

Si esto lo aplicamos y suponemos un malware de este tipo afectando a una página web o recurso online altamente solicitado, el efecto sería a gran escala, llegando a ser devastador el daño que puede hacer.

Son el perfecto ejemplo de un malware que se detecta por ralentizar el sistema, dado que se replica a si mismo por todas partes, sobrecargando el sistema, la red, el disco, y la RAM.

Si tus tareas diarias se vuelven más lentas, llegado a ser incapaces de realizarse, posiblemente el malware que está activado sea un gusano.

Malware Oculto

Este tipo de malware se distingue por su facilidad y persistencia a la hora de evitar ser detectado a toda costa, pasar desapercibidos e introducirse sin ser avistados en el sistema.

Destacaremos algunos como BackDoor, Rootkits y Troyanos entre otros.

BackDoor

Se les suele llamar también “Puertas Traseras”.

En sí son agujeros en la seguridad, provocados de una manera u otra por el atacante, con la finalidad de sobrepasar la seguridad establecida.

Estas puertas traseras permiten el acceso a tu ordenador sin ser detectado, una vez se cumple este propósito, normalmente son otros tipos de malware los que empiezan a provocar el caos.

Drive-By Downloads

Se suelen ubicar dentro de las webs o correos electrónicos.

El funcionamiento es realmente simple de entender, lo primero, el hacker coloca un SCRIPT ejecutable en la web, así al entrar el usuario en la web, el SCRIPT se descarga sin su consentimiento.

Posteriormente el SCRIPT empieza a enviar peticiones a un servidor repleto de exploits, los que tienen como objetivo, encontrar las fallas y debilidades del ordenador desde el cual el SCRIPT está enviando.

Cuando esas debilidades sean encontradas, entrará en acción otro malware, mucho más específico, creado para explotar la debilidad que haya sido detectada, quedando así el ordenador infectado.

Rootkits

En este caso hablamos de una amenaza que utiliza a otras para hacer daño, como en muchas anteriores que ya hemos comentado, algunas son simples puertas para dejar a otras actuar, en este caso, hablamos de un panel de control, muchas veces muy complejo y encriptado.

Son paneles de control que permiten realizar una infinidad de acciones, de una forma mucho más rápida y cómoda.

En lugar de escribir, mandar y ejecutar el SCRIPT, se presiona un botón que está enlazado a ejecutar la orden previamente preparada, dado que los archivos necesarios ya habrán sido cargados en el sistema.

Normalmente estos sistemas son instalados a sistemas donde se tiene acceso ROOT (Administrador), de tal manera que puedas realizar cualquier acción sin que el Sistema Operativo te de problemas.

Otra motivo del porque previamente se ha de tener acceso ROOT, es por el hecho de que al tener acceso de Administrador puedes realizar acciones de manera oculta, consiguiendo que los procesos que inicias, no aparezcan en la lista de procesos, lo que hace realmente difícil poder detectarlos.

Troyanos

Su nombre proviene del famoso caballo de Troya, a causa de que su funcionamiento y finalidad son idénticos.

Son aquellos programas, archivos, ya sean adjuntado en un correo o pasados por USB, que a simple vista y juicio del usuario son absolutamente normales y fiables, pareciendo inofensivos, pero al llegar a tu ordenador, montan la fiesta.

Abren una puerta trasera (BackDoor) por lo general.

Es otro malware que es un puente que da acceso a otros tipos de ataques con finalidades de robo de información o de destrucción de archivos.

Referencias Bibliográficas: [4]

Malware para Obtener Beneficios

El malware ha evolucionado mucho en los últimos años, pasando desde finalidades como bromas o travesuras, hasta hoy en día que llegan a generar beneficios, ya sea por robo de información o por interceptación de información bancaria.

En base al enfoque que la mayoría utilizan los podemos clasificar en:

Mostrar Publicidad

Spyware

Software diseñado con la finalidad de recolectar información de tu ordenador y de tus operaciones con este, y enviar esta información a otra persona.

Pueden ser usados tanto para conseguir información y usarla en tu contra, como para averiguar información y mejorar campañas publicitarias.

Este tipo de malware puede estar tanto en correos electrónicos como en las cookies de las webs.

La manera de identificar su actuación suele ser la genérica, es decir, observar que tu ordenador va más lento de lo normal, dado que el malware consume recursos y puedes notar su ausencia.

Adware

El malware publicitario por excelencia, su misión se resume en las palabras “Publicidad Intrusiva”.

Mediante elementos emergentes. pop-ups… De manera constante y sin descanso, muestra publicidad intrusiva, llegando a hacer imposible la navegación en algunos casos.

No es necesario ser atacado para sufrir este tipo de malware, muchos productos comprados los llevan incorporados e incluidos en la descarga.

Y si aceptas en las condiciones de instalación o descarga (Shareware), acabas infectado con este tipo de malware.

No se considera un malware de carácter dañino, sino molesto.

Hijacking

Cuando hablamos de este malware estamos hablando de secuestro.

Su finalidad es robar o adueñarse de algo de manera ilícita, afectando a la víctima.

Generalmente recaudan información a través del router o modem.

Dado que todos los paquetes de datos son leídos por este malware, pueden obtener la IP, sesiones del navegador, páginas de inicio, contraseñas…

Robar Información Personal

Keyloggers

Un Keylogger puede ser tanto software como hardware instalado sin permiso con acceso a la red.

Recopilar las teclas pulsadas del teclado, para poder posteriormente leer todo lo que se haya escrito, y con la finalidad de obtener contraseñas, conversaciones, o datos de importancia que hayan sido escritos por el teclado.

Este tipo de malware suele tener como finalidad obtener contraseñas, generalmente las de las cuentas bancarias, en operaciones rutinarias como lo es el ver el saldo de tu cuenta bancaria por la web.

Hay empresas que ya han tomado medidas para bloquear este tipo de intrusiones, como ejemplo vamos a destacar “Bankia”.

Cuando accedemos a Bankia, tendremos que teclear con el teclado nuestro DNI = NNNNNNNNL el cual se podría obtener mediante el Keylogger, pero a la hora de introducir la contraseña tenemos dos posibilidades.

La primera es mediante el ratón seleccionar los números. Por lo que al no escribirlos con el teclado, el hacker no podría tener acceso a ellos.

La segunda es introducirlos por teclado, pero no directamente, hay una tabla con valores, de tal manera que si quieres meter un 4 y en la tabla pone 4 = W, tendríamos que pulsar la tecla del teclado “W” para poder escribir un 4.

Aunque el Hacker recibiera nuestros datos, solo podría ver el DNI y una contraseña numérica de 4 dígitos como esta: KWTZ.

La que es imposible de desencriptar excepto que se tenga acceso a la tabla de correspondencias que se utilizó en el momento de introducirla, la cual cambia constantemente en cada acceso.

Stealer

Es el complemento perfecto para el Keylogger, dado que realiza funciones que el anterior no puede.

En lugar de esperar al usuario y al teclado, este malware recorre todo el ordenador en busca de contraseñas almacenadas, al igual que el navegador con sus campos de auto rellenado guardados, y sus contraseñas guardadas.

Una vez las obtiene, procede a descifrarlas.

En el momento en el que las tiene ya descifradas, las envía al hacker, dejando al usuario totalmente a merced de este.

Realizar llamadas telefónicas

Dialer

Es un malware un tanto extinto, debido a que antiguamente se utilizaban los modem “dial-up” que compartían señal con el teléfono fijo.

El funcionamiento de un Dialer se resume en dos pasos.

Primero, toman el control de un modem, y por último realizan llamada a teléfonos con tarificación especial, al extranjero…

Desde la aparición del ADSL este tipo de malware ha ido decayendo en su aparición y los problemas que plantea.

Ataques Distribuidos

Botnets

Un bot es algo así como un Zombi de una película, sólo que tiene un dueño, por ello un Bot es algo como un ordenador esclavizado que sigue las órdenes que le han pautado, formando parte de una red de bots.

Esta red conforma un ejército de bots. Y se ha creado infectando a muchos ordenadores en distintos momentos.

Cuando el hacker o dueño de los Zombis Bots, está listo para atacar, lo primero es dejar de aumentar el número de bots de la botnet.

Lo segundo es enviar Spam para atacar webs mediante el conocido DDOS.

Adicionalmente cabe mencionar que el malware de los ordenadores se puede actualizar de manera remota, automática, sencilla y oculta, dado que el hacker permanece en el anonimato.

Extorsión y Amenazas

Rogue Software

En sí no es un malware.

Es un modelo de negocio que hemos sufrido todos.

Un software de este tipo procede a engañarte, haciendo uso del efecto placebo de la persona que va a ser víctima del engaño, para posteriormente cobrarte por una solución falsa.

La mejor manera de verlo, es con un ejemplo típico:

Te levantas temprano un día que tienes que trabajar en el despacho, y conectas el ordenador para trabajar, al arrancarlo te metes en tu web favorita de noticias.

Una vez dentro, te recomiendan instalarte un Antivirus gratuito que te avisa de si hay algún tipo de problemas…

En ese momento tú piensas:

"Si es gratuito, total, no me hará daño probarlo y así me aseguro"

En ese momento te lo descargas, este hace un falso análisis, y te dice que ha encontrado 27 virus, de los cuales hay 5 que son súper peligrosos, y te recomienda actualizar a la versión PRO del Antivirus para repararlos.

Tú en ese momento, salvo que seas desconfiado, te asustas, piensas que vas a perder todo tu trabajo, que vas a ser víctima de amenazas y extorsión, cuando realmente ya lo estás siendo…

Por lo que al final decides pagar esos 5€ que no van a ninguna parte para arreglar los problemas (Inexistentes).

El Antivirus de mentira se lleva tus 5€ y tú te llevas un efecto placebo muy interesante, dado que seguramente ni siquiera haya realizado una búsqueda de Virus real.

Y así funciona este tipo de estafa.

El % de ganancias obtenido por los hackers, se basa exclusivamente en la suerte o la **ingeniería social** previa, de la que hablaremos más tarde, para descubrir que personas tienen más facilidad para creerse la estafa, y proceder a pagar.

Por mucho que una persona sea una víctima fácil, si está en el paro y tiene problemas de impago, no será fácil que la víctima pague 20€ para sentirse seguro…

En estos casos la **Ingeniería Social Previa** es la clave.

Ransomware

Es uno de los malware más recientes.

Primero es transmitido en el ordenador objetivo, infectando el Sistema Operativo como si de un Gusano se tratase.

Posteriormente el Malware encripta todos los archivos mediante una combinación de Clave Pública/Privada, que no puede conseguirse.

Una vez ha dejado todos los archivos inaccesibles, es cuando entra en movimiento el Hacker, solicitando un pago para desbloquear los archivos, devolverte la contraseña de encriptación, para así poder tener acceso a los archivos previamente encriptados.

Cabe destacar que es una forma de extorsión en toda regla, llegando a grabar por la webcam actividad diaria del usuario, con la finalidad de que la extorsión se produzca como ha sido previamente planeada.

Casi en el 100% de los casos, no se trata de un ingreso a fondo perdido, es decir, normalmente y casi siempre, si pagas obtienes el código de desbloqueo.

Dado que si el Hacker encima de encriptarlo todo, te engañara y no desbloqueara los archivos, las redes se inundarían de palabras y expresiones como “timo”, “estafa”…

Consiguiendo que los beneficios, objetivo final de todo el proceso, sean menores, llegando a no ser rentable el tiempo dedicado.

Un ejemplo parecido, que realmente es un troyano, pero el funcionamiento y el objetivo final se acerca mucho más al de Ransomware es el caso del Virus Policía Nacional.

Se muestra una imagen como esta:

En la que se comunica que por actividad de pornografía infantil se ha bloqueado el ordenador y que debes de pagar una multa para desbloquearlo.

Por miedo o vergüenza, la víctima acaba pagando a menudo.

Malware Web

El malware web es una caracterización del malware original que aprovecha el potencial de internet a su favor.

Explota vulnerabilidades de los servidores web, repercutiendo negativamente en el servidor.

Pierde confidencialidad, fiabilidad, estabilidad y disponibilidad…

Por lo que se podría clasificar como una amenaza importante y prioritaria a la hora de combatirla, estando atento sobre todo a las actualizaciones que a nivel de web, se dan más a menudo.

Los ataques pueden producirse por fallos o descuidos:

- Contraseñas o números de tarjeta de crédito con poca seguridad, llegando a ser obtenidos mediante programas diccionario que comprueban y testean listados de contraseñas más comunes de manera automática.

- Sistema susceptible a suplantación de sesiones o de contenido, favoreciendo la suplantación de usuarios tanto por sesión como por contraseña.

- Recuperación de contraseñas con poca seguridad o fallos.

- Sistema susceptible a suplantación de sesiones o de contenido, favoreciendo la suplantación de usuarios tanto por sesión como por contraseña.

- No ocultar el árbol de archivos del servidor web, revelando así los archivos de los que dispone la red, y favoreciendo el ataque a estos.

- Fugas de información sensible de los desarrolladores, comentarios, mensajes de erros, defectos.

- Niveles insuficientes de autenticación.

- No protegerse de las inyecciones de datos en los formularios, haciendo posible que el malware SQL Injection tenga mayores probabilidades de tener éxito tanto en búsquedas como en Log In/Log Out.

  Permitiendo así el acceso a datos sensibles almacenados en las bases de datos, violando la ley de protección de datos.

Malware Web - Ataques Web

Defacing o Defacement

Es un ataque que utiliza una web modificada o desfigurada.

El hacker mediante técnicas de acceso, ha obtenido los archivos de la web, copiando íntegramente el código y cambiando campos con la finalidad de copiar datos y sacar beneficio de ello.

Un perfecto ejemplo es el correo de Apple/Bankia (Phishing):

Te llega un correo de la compañía Apple, diciéndote que ha habido un cambio en tu cuenta y que han intentado acceder a ella.

Posteriormente te pide los datos para poder verificar que eres tú y pedirte participación en las acciones tomadas contra el atacante.

Te redirige a una página absolutamente idéntica a la de Apple.

Con la diferencia de que en el código de la página, hay modificaciones como:

Creación de variables → Nombre, Usuario, NIF, Contraseña…

Y una petición oculta de enviar correo electrónico a hacker@yopmail.com.

Obteniendo así una copia de los datos introducidos.

Al darle a Enter, la página se recarga, como si hubiese ocurrido algún tipo de error en el envío de la información, y te los pide de nuevo.

Esta vez, es la página oficial de Apple, a la que realmente accederás y verás que no hay ningún tipo de problema.

Pero el Hacker ya ha obtenido tus datos.

Ha habido muchos casos de este tipo de Ataque Web llamado Defacing.

En particular, en España con las compañías Apple y Bankia.

Pero en el caso particular de un banco, el Hacker ha obtenido tus datos de acceso a tu cuenta bancaria, y el daño puede ser mucho mayor.

O en tu cuenta Apple, de comprar 10 iPhone 6 Plus con todos los extras, y asustarte con un cargo de 12.000€ en la cuenta bancaria hasta que procedas a la devolución y anulación.

Referencias Bibliográficas: [4]

DDoS

Son las siglas de **D**istributed **D**enial **o**f **S**ervice, o como se le conoce en España "Ataques de Denegación de Servicios".

Ya habíamos hablado de este problema cuando hemos tratado el malware conocido como Botnet.

Y es que, gracias a una Botnet como hemos comentado con anterioridad, el hacker puede orientar las peticiones de sus ordenadores Zombis a un mismo punto, saturando el servicio.

Todas las máquinas de la Botnet se conectan al mismo tiempo a una web, comenzando a hacer peticiones de todo tipo, provocando una saturación masiva de puertos y del ancho de banda.

Haciendo posible la caída del servidor, inhabilitando así la página o páginas que estuviesen alojadas en este.

Es muy utilizado, dado que no requiere coste adicional una vez se tiene una Botnet preparada y lista para acatar órdenes y solicitar peticiones a la página web en cuestión.

Al provocar que la web no esté disponible, el dueño pierde control sobre ella y de estar de baja durante mucho tiempo o varias veces consecutivas, puede repercutir arduamente sobre el negocio.

SPAM

Término que hace referencia a los "Correos no deseados", ya sea por el contenido, repleto de enlaces y publicidad, o por el remitente, anónimo o sospechoso.

Son enviados en grandes cantidades con la intención de infectar, ya sea por publicidad, o por archivos como un PNG donde va incorporado un SCRIPT, generalmente se le conoce como la amenaza más molesta.

Y la que más se sufre, pues es la que mayor número de casos sufridos existen.

Se utilizan técnicas para obtención de direcciones de correo electrónico, ya sean públicas o robadas, o por combinaciones de letras y números, usando la famosa técnica de "Prueba y Error".

Si se devuelve la respuesta, el correo existe y pasa a formar parte de la lista de correos. En caso de fallar, lo descarta.

El SPAM es una molestia, llegando a ser un verdadero problema para empresas que se dedican al Hosting, o Administración de Correos Electrónicos de grandes organizaciones.

Exploits

Un exploit es todo ataque que explote y aproveche las vulnerabilidades que representa un servidor web.

Previamente se ha de detectar el fallo o vulnerabilidad con otro tipo de malware y mandar los resultados al Hacker.

Posteriormente este pondrá en activo un exploit para aprovechar esta vulnerabilidad a su favor, tanto tiempo como pueda, dado que la mayoría de vulnerabilidades de las que se aprovechan los exploits son reparadas en cada actualización.

Phishing

El Phishing es un ataque web del estilo al Defacing/Defacement que hemos visto con anterioridad.

Consiste en la falsificación de correos electrónicos de empresas de confianza, incitándote a hacer click en los enlaces, obteniendo así información confidencial del usuario.

Es una técnica muy utilizada, como hemos visto con anterioridad en el ejemplo del apartado de Defacing, que realmente es un ejemplo de Phishing.

De ahí la importancia de leer bien los correos y comprobar con la mayor certeza posible la certeza y fiabilidad tanto del correo como del remitente.

Recordar que nunca hay que aportar ningún dato personal a un trabajador de una empresa, ya que estos disponen de los datos necesarios, no deben ni pueden preguntar sus datos personales.

SQL Injection

Es un método que mediante una cadena SQL introducida en campos de búsqueda pretende aprovechar debilidades y obtener toda una base de datos por ejemplo.

La web espera que tú le pongas la cadena de caracteres a buscar, pero si en esa cadena hay una orden de cierre de sentencia SQL y el buscador la acepta, el hacker puede ejecutar cualquier código.

Descargándose la base de datos entera, borrarla, o modificando sus valores.

Generalmente que una persona ajena a una organización tenga acceso a la base de datos, suele ser un tema preocupante, dejando a la empresa en una posición comprometida, requiriendo de una solución rápida y efectiva ante un ataque tan sencillo de realizar.

XSS

El XSS se refiere a un agujero en la seguridad permitiendo a los hackers insertar código en las webs, evitando medidas de control y pasando desapercibidos.

Suelen atravesar los controles a través de vulnerabilidades previamente analizadas.

Es parecido al SQL Injection.

La primera manera de ejecutar código sería desde la URL de la página:

www.digitalvalue.es/home?q=rootkit3.0

Ejecutando una función JavaScript después del "?q=".

La segunda manera es añadir al código propio de la web un SCRIPT o un iFRAME que permita al hacker robar la información de las cookies, del usuario, de la sesión.

También suele ser utilizado para en el iFRAME creado mostrar publicidad.

No hay que perder nunca de vista la finalidad de todo ataque web.

O es para obtener beneficios, lo que debería preocuparnos y estar más atentos a actualizaciones.

O es por el simple hecho de hacer daño.

Generalmente suele ser por el beneficio, llegando a ser únicamente el beneficio el mostrar la publicidad intrusiva de las empresas que se muestren en los diferentes apartados de publicidad.

Referencias Bibliográficas: [4]

Otros ataques web…

Hay miles y miles de diferentes ataques web, al inicio del documento explicábamos el porqué es tan difícil de hacer frente al problema.

Los ataques web, generalmente están personalizados, por lo que al igual que si un malware hace destrozos en una máquina y en otra directamente ni se ejecuta porque es inservible.

Hay que entender que las soluciones a estos problemas personalizados han de ser también personalizadas.

Ingeniería Social

Definición

Hay muchas definiciones para el concepto de Ingeniería Social.

Pero a mí en particular me parece muy correcta la siguiente:

Es la práctica de obtener información confidencial a través de la manipulación de usuarios legítimos.

En otras palabras, la Ingeniería Social, es aquel proceso que pasa por diferentes fases como el espionaje, las encuestas, el historial, etc…

Con la única finalidad de obtener información, de tal manera que si el Hacker en cuestión quisiera estafar a los usuarios mediante un ataque de Phishing o de Defacing, y pretende cobrar el "Rescate" por Paypal…

Seguramente su beneficio y % de acierto se verán aumentados si enfoca los ataques contra usuarios que sabe con anterioridad que tienen cuenta Paypal y que saben cómo transferir dinero.

Utilización

La ingeniería social se sustenta en el principio y la afirmación de que "El usuario es el eslabón débil". Ya sea por teléfono fingiendo ser un empleado o un técnico o un cliente.

O desde Internet, donde adicionalmente se mandan peticiones pare renovar permisos de páginas web, o solicitando renovar la contraseña por ataques web, cuando realmente el ataque estás siendo ejecutado en ese mismo momento.

O simplemente desde el punto de vista cara a cara, obteniendo acceso a sistemas informáticos o móviles, o cuentas de correo, respondiendo a la pregunta:

¿Qué contraseña introduciría yo si yo fuese la víctima?

Pregunta mucho más fácil de contestar si se tiene acceso al pasado o presente de la víctima, si ha habido algún fallecimiento reciente, si tiene mascotas, o preguntarle por su número de la suerte.

Los usuarios no tienen cuidado, simplemente observando mientras introduce la contraseña o gastando la broma de yo te digo la mía (Falsa) y tú me dices la tuya, a ver cuál es más segura. La gente cae, la gente es descuidada, y por ello la Ingeniería Social tiene éxito, y es a día de hoy de las mejores maneras de obtener información

Referencias Bibliográficas: [3]

Digital Value

Proyecto – Antivirus Web

Centrándonos ya en el proyecto desarrollado como tal y dejando a un lado por ahora la investigación.

He realizado las prácticas y el Trabajo Final de Grado en la empresa Digital Value.

Una empresa dedicada desde el año 2000 al sector de Internet.

Ofreciendo soluciones a problemas tanto a nivel de almacenamiento web (Hosting), como de seguridad, al igual que afrontando prácticamente cualquier tipo de proyecto a nivel de redes, comunicaciones y seguridad.

Hemos trabajado en equipo en un gran proyecto al que hemos llamado:

Antivirus Web

Este gran proyecto se ha subdividido en otros proyectos más pequeños por varios motivos, tanto por la facilidad de comprensión del problema al dividir este, como por el reparto de tareas entre los integrantes del equipo.

Por comentar unas pocas de las piezas del rompecabezas, estas son las que quisiera destacar entre todas:

- Heurísticas Avanzadas de Detección
- Desencriptación de los Virus
- Listas Blancas y Listas Negras
- Catálogo de Amenazas
- Uso de herramientas externas como apoyo
- HoneyPot
- Blog Comunicativo
- …

El Catálogo de Amenazas ha sido desarrollado en su totalidad por mí.

Y adicionalmente he formado parte del desarrollo tanto de las Heurísticas Avanzadas de Detección, como de la Desencriptación de Virus y del Uso de herramientas externas como apoyo.

Cada uno de estos proyectos por sí solos ya son interesantes y merecedores de páginas y páginas para explicarlos con mayor detenimiento, pero me centraré en los que he participado y en uno más por su curiosidad, el HoneyPot.

Problemas de Seguridad

En Digital Value se toman muy en serio la seguridad, pero al dedicarse al Hosting o Almacenamiento Web, como todas las empresas en este sector, se otorgan permisos de administrador a los clientes en sus espacios contratados.

Dependiendo del plan escogido, Digital Value asiste de manera técnica, actualizando y parcheando los problemas o fallas que haya con las actualizaciones del CMS (WordPress, Drupal, Joomla, PretaShop, Magento…).

Pero en otros casos, son los clientes los que tienen que gestionar su seguridad, sus actualizaciones y por tanto son ellos los encargados tanto de estar atentos cuando haya actualizaciones como de aplicarlas.

Generalmente el procedimiento es el siguiente.

Caso 01 – Digital Value se encarga de la Seguridad y el Mantenimiento

- El Cliente no se preocupa de nada porque todo estará al día y en perfectas condiciones.

Caso 02 – El cliente se encarga de la Seguridad y el Mantenimiento

- Digital Value avisa al cliente de la mayoría de actualizaciones, este adicionalmente puede seguir en la página web el “Blog Comunicativo” para estar al tanto de noticias de importancia a nivel de seguridad o proyectos.
- El Cliente no siempre gestiona la incidencia y decide que la mejor idea es “Dejarlo estar… Total… No va a pasar nada…”

Muchos Hackers esperan oportunidades como estas, para infectar las máquinas que tienen vulnerabilidades y de por sí ya tienen brechas de seguridad.

Porque el esfuerzo para infectarlas es menor.

Como adicional a las páginas webs y otros proyectos, Digital Value mantiene y gestiona los servidores de correo electrónico de multitud de empresas y ayuntamiento, el mayor problema de Digital Value es el SPAM.

Aquí es donde entra el Antivirus Web a solucionar los problemas.

Posteriormente hablaremos de las heurísticas, del código, del catálogo y de los resultados obtenidos desde que está funcionando en la mayoría de webs el Antivirus Web.

Ahora es momento de hablar del **¿Por qué?**

¿Por qué Digital Value necesita el Antivirus web?

Si asumimos el concepto de la máquina de un cliente que está descuidada por este, y por ende, infectada.

Pero lo aplicamos a miles de máquinas virtuales y que algunas de ellas gestionan miles de cuentas de correo electrónico y no solamente una…

Tenemos un gran problema de SPAM si los Hackers hacen de nuestros sistemas sus Zombis convirtiendo las máquinas en parte de la Botnet.

Para reducir el SPAM, se diseña una nueva medida de seguridad.

¿Cómo funciona?

El antivirus web es un archivo PHP, cuyo funcionamiento simplificaré en una definición:

Está ubicado en cada máquina/web, controla y monitoriza todo tipo de conexiones entrantes, archivos creados, archivos modificados, o eliminados.

Llevando un registro de todo lo anterior que envía a los técnicos cada cierto periodo de tiempo.

Desencripta si es necesario el código, para buscar posteriormente cadenas o IPs problemáticas, almacenadas previamente en una base de datos.

Tiene muchas más particularidades de las que hablaremos más tarde.

Y al final veremos en qué medida ha mejorado el problema de SPAM.

Tecnologías Empleadas

Catálogo de Amenazas

Para el Catálogo de Amenazas he utilizado las siguientes tecnologías:

- PHP
- HTML 5
- CSS 3
- JavaScript
- JQuery
- FrameWork Bootstrap

Heurísticas Avanzadas

Para desarrollar el código en el que se encuentran las heurísticas, se han utilizado los siguientes lenguajes:

- PHP
- Parl

Herramientas Externas

Para aprovechar otras fuentes de información, las bases de datos del Antivirus se han nutrido de varias bases de datos públicas.

Adicionalmente también mencionar que se ha interactuado con otros blogs de la competencia para compartir información sobre ciertas amenazas.

Dos herramientas que nos han servido muchísimo en el tema de Desencriptación de virus son:

- DDecode - d**decode**.com
- UnPHP - un**php**.net

Equipo Físico - Hardware

Para el correcto desarrollo del HoneyPot, que es un tipo de trampa para malware con la finalidad de estudiarlo posteriormente, se ha utilizado un sistema dedicado.

Es decir, se ha montado un servidor dedicado, con vulnerabilidades y fallas previamente estudiadas para aumentar la probabilidad de que cierto tipo de malware lo infecte vía web.

Y así poder observar su desarrollo, sus efectos y su expansión.

Implementación – Catálogo de amenazas

Requisitos

Después de reunirnos con varias personas de la empresa, integrantes en varios proyectos del Antivirus Web.

Estas personas, que en el futuro serán las encargadas de catalogar y de revisar la base de datos, con la finalidad de saber cómo afrontar las diferentes amenazas que se puedan presentar en sistemas y webs de las que ellos son encargados de asegurar la seguridad, la fiabilidad y total disponibilidad.

Se determinó que querían un Catálogo de Amenazas que fuera:

- Amigable y fácil de utilizar
- Completo
  - Que tuviera todos los apartados importantes en los que se puede especificar diferente información sobre las amenazas.
- Online
  - Pero con acceso restringido por IP, y otras medidas de seguridad integradas en la empresa.
- Que tuviera un nivel de riesgo
- Que se pudiera guardar la información de un analizador web
  - En este caso el analizador es www.virustotal.com
  - Del que se guarda la fecha del análisis y los resultados de varias pruebas.
- Que tuviera herramientas y estadísticas
  - Este último apartado no ha sido desarrollado todavía dado que aún no se han especificado que herramientas serían útiles y se sigue debatiendo sobre el tema.

Por ello uno de los proyectos que conforman en su totalidad el Antivirus Web es el Catálogo de Amenazas. Los datos a almacenar en el catálogo se determinaron previamente:

- Nombre y Autor de la Amenaza
- Tipos de la Amenaza
- Fechas de Primera Vez Observada y Obtención del Parche
- Descripción de la Amenaza
- Ejemplo del Malware
- Parche para la Amenaza
- Análisis Virus Total
- Firma
- Referencias
- Observaciones
- Nivel de Riesgo

Manual de Usuario

En el manual de usuario hablaremos de la interacción con el catálogo, comentando las capturas de pantalla.

Destacar que la base de datos del catálogo a mostrar no es la oficial de la empresa, siendo imposible mostrarla por temas de privacidad.

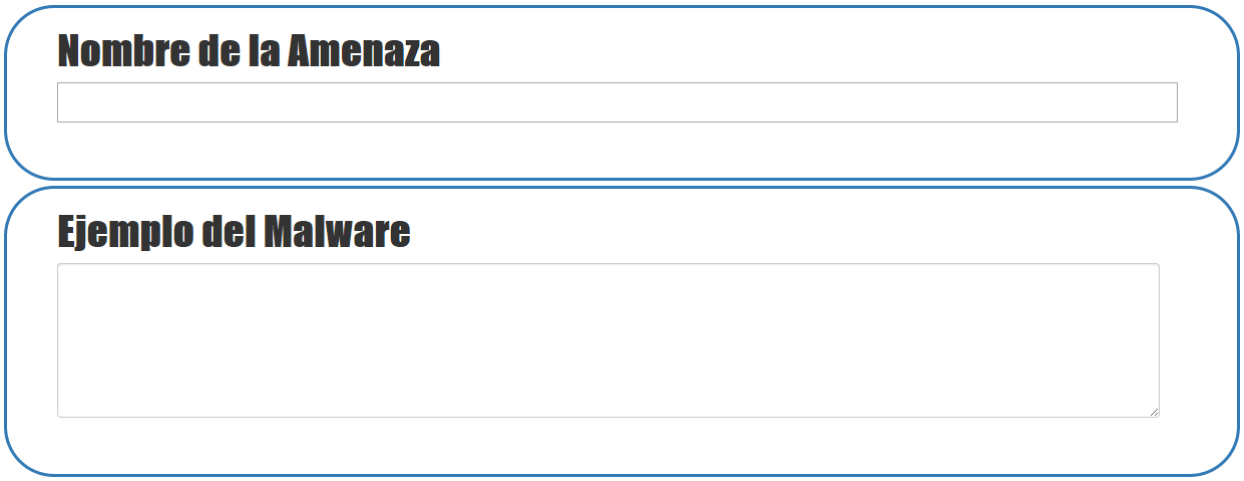
Menú Principal

Agregar Amenaza – Campos

Los campos pueden ocultarse o mostrarse a voluntad.

Pero como se puede observar, aquellos que empiezan mostrados, son los obligatorios, si alguno de estos no está completo, o el formato no es el correcto, no dejará agregar la amenaza.

Agregar Amenaza – Input y TextArea

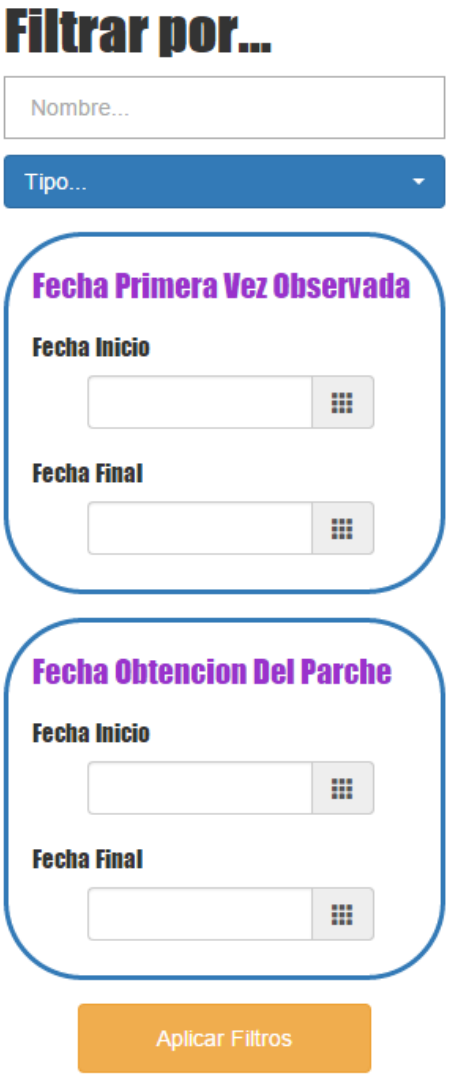


Agregar Amenaza – Fechas

Agregar Amenaza – Análisis Virus Total

Agregar Amenaza – Nivel de riesgo

Editar Amenaza - Filtrado



Se puede filtrar la búsqueda para que se muestre en la tabla (Imagen Siguiente).

Por nombre, tipo o fechas.

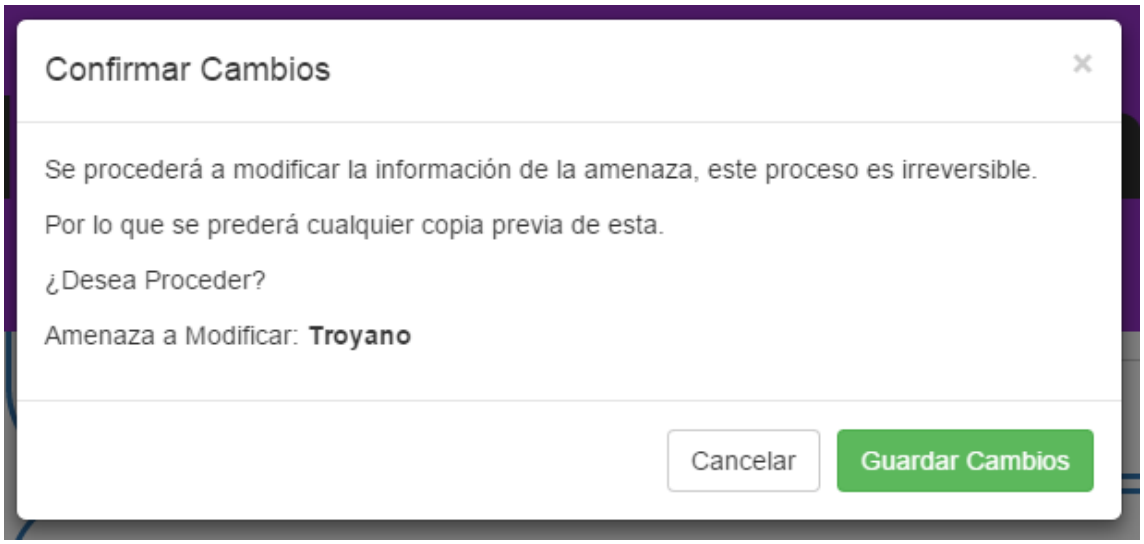
Editar Amenaza - Listado

Una vez se ha elegido la amenaza, se recogen sus datos por la ID.

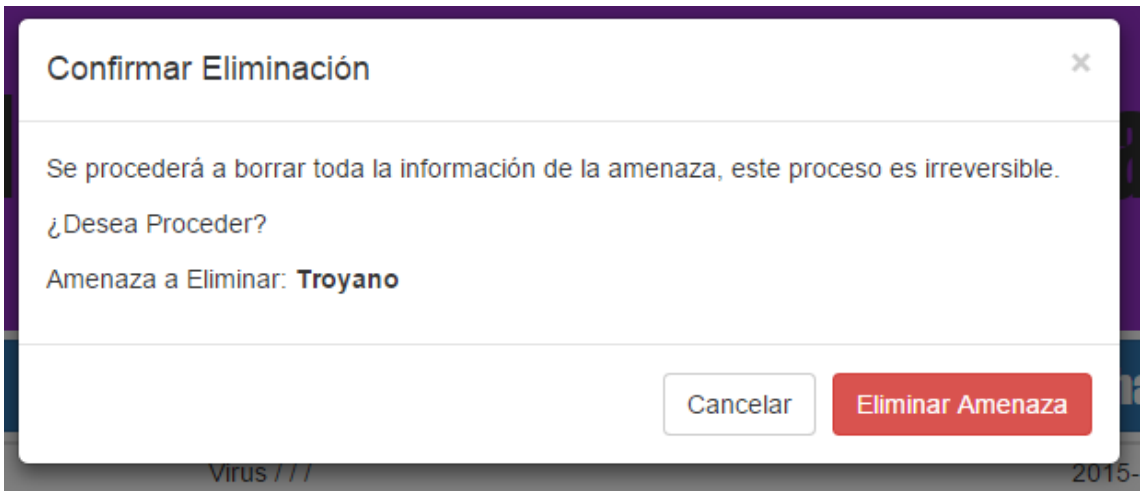
Y se abren los mismos campos que en el agregar amenaza, pero con sus valores almacenados en la base de datos, si los tuviera.

Se procede a realizar los cambios, y se le da a “Guardar Cambios”.

Editar Amenaza – Guardar Cambios



Eliminar Amenaza – Confirmación



Implementación – AntiVirus Web

Funcionamiento

El Antivirus Web ha sido programado y diseñado para cumplir necesidades de seguridad, el Destructor de Amenazas Web (DAW), como a nosotros nos gusta llamarlo.

Protegiendo cualquier web que esté alojada en nuestros servidores ante la mayoría de amenazas web conocidas o recientes.

La primera versión del **DAW** fue programada en **PHP + JavaScript** para la parte de la funcionalidad y la conexión de la base de datos.

HTML y CSS para la estética de la web desde la que se utiliza y se gestiona. Adicionalmente comentar que se ha utilizado el Framework Bootstrap al igual que en el Catálogo de Amenazas.

El Antivirus es capaz de instalarse desde una máquina ajena al servidor, conectándose por SSH y enviando el ZIP previamente generado.

Una vez se ha ubicado el Antivirus, este dispone de varios módulos:

- Un Escáner
  - Para detectar las posibles amenazas
- Un Inspector
  - Para descodificar los archivos que parecen infectados
  - Para así poder ver el código original y entenderlos
- Acciones Root
  - Editar o Eliminar el Archivo
- Modo Debug
  - Mediante un archivo de configuración se puede activar el Modo Debug, viendo así con más profundidad los mensajes de aviso
- Archivos de definiciones y firmas
  - Varios archivos con cadenas peligrosas, o firmas, palabras, o expresiones regulares que nos lleven a catalogar el archivo como preocupante, y tras una revisión, clasificarlo como amenaza o como archivo libre de virus
- Reportes programados
  - Que se envían a los encargados de las webs en cuestión
- Cuarentena
  - Donde se copian los archivos preocupantes, eliminando el original y pudiendo así volver a recuperar el archivo en caso de que no sea una amenaza
- Cron
  - Genera nombres aleatorios para el Antivirus, comprueba la integridad de este, y puede restaurarlo o reenviarlo.

Gracias a la comprobación del md5 del Antivirus Web, podemos saber con certeza si ha sido modificado, procediendo a restaurarlo o a reenviarlo si ha sido eliminado.

Para cada web se dispone de un Antivirus Web diferente, recopilando los md5 de los archivos almacenados en la Lista Blanca.

Es decir, los archivos que se instalan por defecto del CMS o del software instalado en las máquinas.

Posteriormente tras los reportes, se procede a analizar y a catalogar en el Catálogo de Amenazas.

Destacar que la versión más reciente del Antivirus Web ha sido desarrollada en PERL, por motivos de compatibilidad entre JavaScript y servidores antiguos.

En los Anexos se podrá ver más código de la versión en **PERL**.

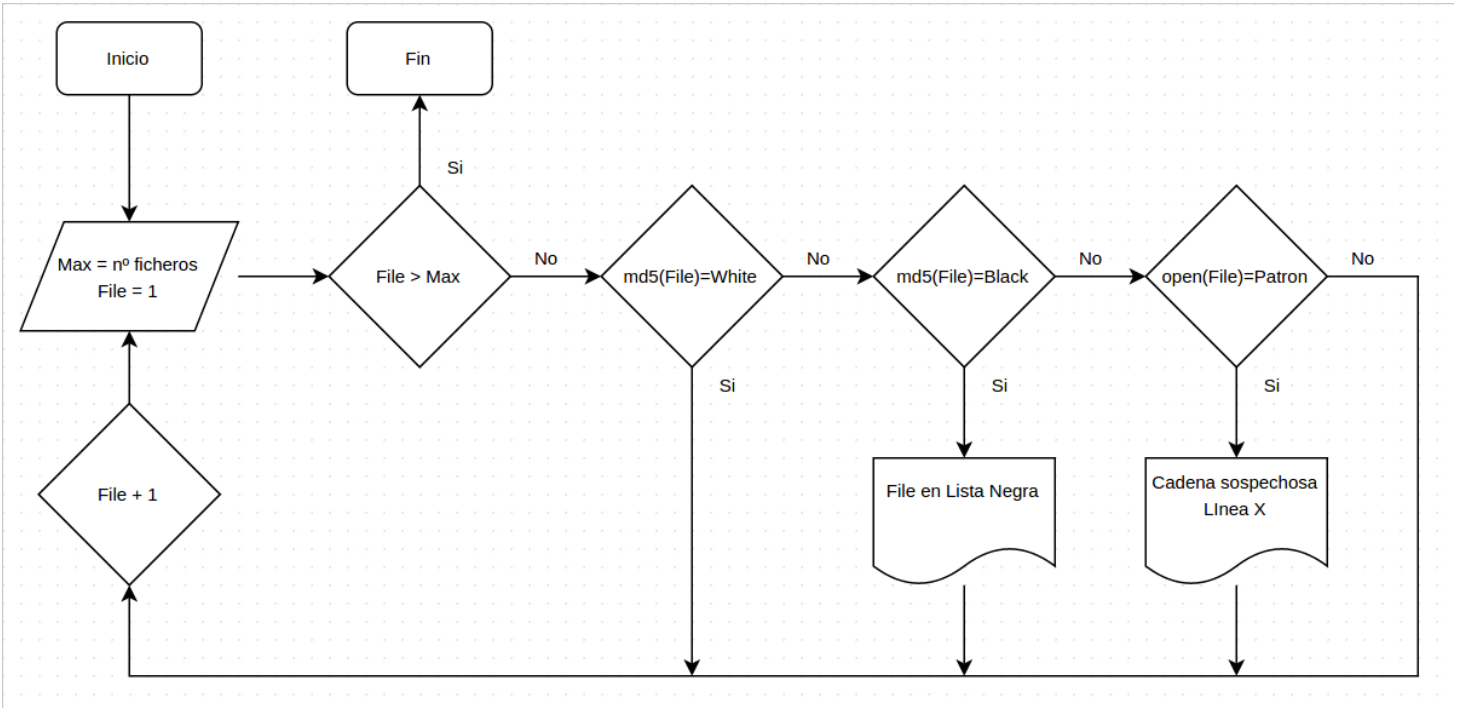
Uso del Antivirus Web

Findbot.pl

Es un escáner de malware creado en PERL suministrado por cbl.abuseat.org con carácter gratuito.

El que cual ya contiene su propio listado de firmas de virus.

Adicionalmente también tiene un pequeño listado md5 de virus utilizados en los sistemas operativos Windows.



Modificaciones de Findbot.pl

Al ser las firmas del código original demasiado genéricas, dan lugar a muchos falsos positivos, para solucionar estos falsos positivos se han utilizado nuevas firmas de virus más fiables.

De igual modo se han ido ampliando estas firmas según se han ido descubriendo nuevos archivos malware o nuevos exploits.

Para generar las listas de md5 tanto blancas como negras se han creado dos nuevas subrutinas de perl, que contienen los códigos md5.

Y para otorgar una mayor rapidez en el escaneo se ha modificado la secuencia de escaneo de archivos para pasar al siguiente archivo al encontrar una coincidencia en las listas md5.

Gracias a la introducción de estas listas, la fiabilidad del escáner ha aumentado y seguirá aumentando gradualmente, pudiendo ser adaptada en cualquier momento de una manera más rápida y fácil.

Cuarentena, Firmas, Listas Blancas y Listas Negras

La carpeta de **cuarentena**, es lo que llamamos copia de seguridad.

En la que se copian los archivos infectados, encriptándolos para anular cualquier tipo de funcionalidad que tuvieran, para analizarlos más adelante en un entorno preparado, como puede ser el HoneyPot.

Al copiar los archivos en la cuarentena se procede a borrar los archivos originales, y en caso de equivocación o falso positivo, se podrá restaurar y recuperar el archivo “perdido”.

Las **Listas Blancas** son bases de datos que contienen los datos referentes de los CMS más utilizados, ya sea Drupal o Wordpress, aquellos archivos que vienen por defecto con la instalación de estos.

Gracias a su md5 podremos saber si ha sido modificado o si coincide en su totalidad con el archivo original.

Datos como el Nombre del fichero, la ruta donde está ubicado, y su md5 nos ayudarán a determinar esto.

En contraposición a las Listas Blancas, tenemos las **Listas Negras**.

Son bases de datos de nombres de archivos, de rutas de instalación, o de cualquier tipo de información sensible que nos asegura al 100% que el archivo que coincida con alguno de estos datos es un malware.

Para poder mejorar la detección y agilizar el proceso, estas listas negras se actualizan constantemente.

¿Qué es una firma de Virus?

Una firma de virus es una parte del código que se utiliza para identificar los diferentes malware:

| preg_replace('/.*/e' |
Este es un trozo de código php usado para ejecutar código cifrado.

| PCT4BA6ODSE |
Esta es una variable recurrente en la creación de un webshell.

| Hacked by |
Perfecto ejemplo de las cadenas que hay que buscar.

Adicionalmente comentar que se utilizan los códigos md5 de loa archivos para definirlos como malware, puesto que en algunas ocasiones una expresión regular puede no resultar fiable.

¿Qué es el código md5?

Es una cadena de 32 caracteres que define un archivo.

Así podemos sacer con certeza que se trata de un archivo en concreto. Sin tener que depender de su contenido o de su nombre.

Dado que previo a la aplicación del Antivirus Web, la empresa tenía muchísimos casos archivados de malware, se han estudiado y analizado obteniendo listados de firmas de alta calidad.

A día de hoy se poseen más de 250 Patrones y más de 1.500.000 códigos md5 para contrastar.

Desencriptación de Virus

El código ofuscado puede llegar a ser un verdadero contratiempo a la hora de decidir si un fichero está infectado o no, a pesar de que normalmente un fichero benigno, no debería de estar ofuscado.

Pero la ofuscación de código se utiliza como protección muchas veces, y esto puede llevarnos a falsos positivos, por ello es tan importante desencriptar el código.

Nos hemos valido de herramientas externas como son:

- DDecode - **ddecode**.com
- UnPHP - un**php**.net

Para crear nuestro propia herramienta de decodificación, capaz de traducir el código php cifrado con gzinflate, gzuncompress, base64 y los códigos que utilizan preg_replace para ser ejecutado.

El texto introducido es filtrado eliminando las etiquetas de php “<?php” y “?>”, a continuación se busca la cadena “eval” la cual es sustituida por “echo”, de esta manera se evita la ejecución del código.

Adicionalmente hay enlaces para acceder a las webs comentadas como un añadido a la Desencriptación.

eval(gzinflate(base64_decode('88jPSVTwLc1LyQcA'))) >>> “Hola Mundo”

Variabilidad Vírica

Tenemos que tener muy presente, que normalmente las amenazas a nivel Web son enfocadas a los diferentes usuarios.

Cada web es distinta, y por ello cada amenaza muta y se transforma para así llegar a infectar con el mayor porcentaje de éxito la mayoría de webs.

Y por el mismo motivo, el Antivirus Web ha de mutar y adoptar diferentes puntos de vista, con la finalidad de abarcar al máximo todas las amenazas que van surgiendo.

Para combatir esta realidad detrás de un buen software como es el Antivirus Web, hay un equipo de técnicos que van mejorando las heurísticas y aprendiendo de los casos de amenazas que nos llegan a diario a la empresa, y con los que tenemos que lidiar.

Es el momento de hablaros un poco más de cerca del **HoneyPot**.

El HoneyPot, es como bien dice la palabra, un tarro de miel.

Un foco perfecto para atrapar amenazas y estudiarlas, vamos a ver cómo sería su funcionamiento:

1- Se medita sobre un cierto tipo de brecha de seguridad de la que se pretende obtener más datos sobre las amenazas que pretenden explotar estas vulnerabilidades.

2- Se monta un equipo absolutamente alejado e independiente de la red de equipos, para evitar que la amenaza salte.

3- Se instala el software necesario en el equipo sin parchear, en caso de que se pudiera, la vulnerabilidad que queremos estudiar.

4- Se ponen diferentes tipos de software para capturar instantáneas del sistema cada cierto periodo, y para estudiar cada cambio que se lleva a cabo en el sistema.

5- **Esperar** **>>>** Es realmente sorprendente la velocidad con la que una amenaza encuentra una vulnerabilidad.

6- Posteriormente se procede a esperar que la vulnerabilidad haga lo que tenga que hacer, hasta que acabe o entre en bucle infinito.

7- Se recauda la información para saber el cómo lo hace, para saber por dónde ha entrado, a que zona ha atacado, y en sí cada iteración en el recorrido y finalidad de la amenaza.

Conclusiones

Sobre el Proyecto

Al empezar el proyecto, con un equipo competente y preparado, y nada más salir de la carrera…

Puede asustar un poco, sin haber dado prácticamente nada de seguridad, ponerse a hacer un Antivirus, quizá no parezca lo más razonable.

Pero luego te encuentras un gran proyecto, interesante y complejo, pero sobretodo útil, y poco a poco te pones a desarrollarlo junto a tu equipo, cada uno se encarga del módulo escogido, pero ayudándose mutuamente por ejemplo a la hora de la interconexión entre estos.

Y al final, después de un gran proyecto, y de una gran aventura, lo pruebas y lo pone en práctica, en cada servidor de la empresa, para sacar estadísticas y de pronto descubres…

¡Que funciona!

Que todo el esfuerzo tenía un sentido, y que estás orgulloso se haber formado parte de un proyecto, que tiene una utilidad, y que está generando beneficios, al reducir las pérdidas.

Se necesitaba un Antivirus Web, para reducir el SPAM como principal amenaza, y para combatir otras muchas.

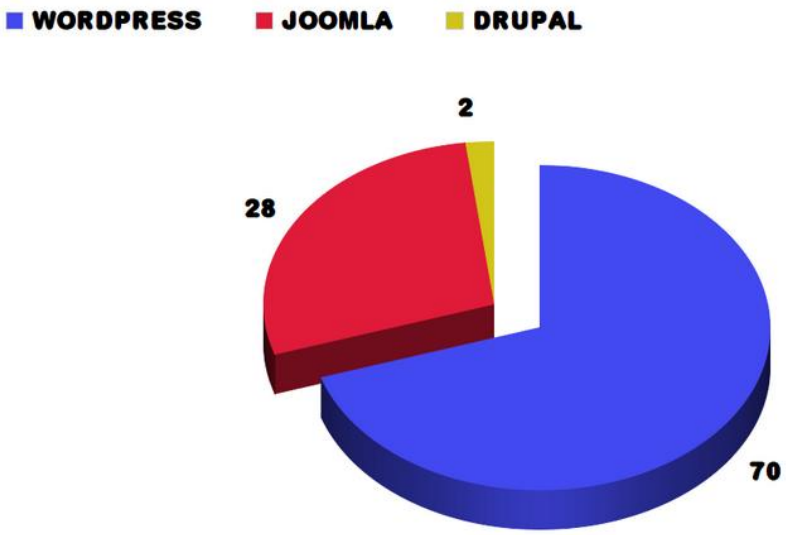
El DAW aún sigue en proceso, se sigue mejorando y aún está evolucionando…

Quedan otros módulos por completar, y partes que ampliar.

Como un gran equipo hemos creado DAW, Destructor de Amenazas Web.

Estadísticas

Después de dedicar 6 meses a la utilización del DAW, se han podido obtener las estadísticas siguientes:



Ocurrencias de las Amenazas por CMS

Se han detectado **9495** Amenazas en los 6 meses que lleva en funcionamiento.

Reduciendo un **92%** el SPAM en los servidores.

Bibliografía

Seguridad Informática [1]

- http://es.wikipedia.org/wiki/Seguridad_inform%C3%A1tica
- http://www.monografias.com/trabajos82/la-seguridad-informatica/la-seguridad-informatica.shtml
- http://recursostic.educacion.es/observatorio/web/ca/software/software-general/1040-introduccion-a-la-seguridad-informatica?start=1
- http://www.ia.urjc.es/cms/sites/default/files/userfiles/file/SEG-I/2012/introduccion.pdf
- http://es.wikibooks.org/wiki/Seguridad_inform%C3%A1tica/Introducci%C3%B3n
- http://educacionadistancia.juntadeandalucia.es/profesorado_taller/mod/book/tool/print/index.php?id=16034
- http://www.csirtcv.gva.es/es/formacion/introducci%C3%B3n-la-seguridad-inform%C3%A1tica.html
- http://es.kioskea.net/contents/622-introduccion-a-la-seguridad-informatica

DataCenter [2]

- http://es.wikipedia.org/wiki/Centro_de_procesamiento_de_datos
- http://www.acens.com/blog/que-es-un-data-center.html
- http://www.acens.com/cloud/cloud-datacenter/
- http://www.aprendaredes.com/blog/que-es-un-data-center-4/
- http://es.slideshare.net/Complethost/qu-es-un-data-center-centro
- http://www.la.logicalis.com/soluciones-servicios/excelencia-data-centers/conceptos-basicos-data-center/
- http://blog.hostalia.com/que-es-un-data-center-centro-de-datos/
- http://blog.guebs.com/2013/08/22/que-es-un-datacenter-o-cpd/
- http://www.firmesa.com/web/datacenter/que-es-datacenter
- https://gigas.com/cloud-datacenter

Ingeniería social [3]

- http://es.wikipedia.org/wiki/Ingenier%C3%ADa_social_%28seguridad_inform%C3%A1tica%29

Amenazas web [4]

- http://es.wikipedia.org/wiki/Malware
- http://support.kaspersky.com/sp/viruses/general/614
- http://www.pandasecurity.com/spain/homeusers/security-info/about-malware/general-concepts/concept-2.htm
- https://climbo.wordpress.com/2007/01/31/tipos-de-ataques-en-aplicaciones-jueb/
- http://roble.pntic.mec.es/jprp0006/tecnologia/4eso_informatica/peligros_internet/
- http://es.wikipedia.org/wiki/Virus_inform%C3%A1tico
- http://listas.20minutos.es/lista/los-tipos-de-malware-310619/
- http://www.welivesecurity.com/la-es/2013/01/02/ataques-aplicaciones-web/
- http://infoynet.blogspot.com.es/2009/04/malware.html
- https://www.infospyware.com/articulos/que-son-los-malwares/
- http://www.idearius.com/es/tipos-de-malware-virus-troyano-spyware-gusano/
- http://www.kaspersky.es/internet-security-center/threats/malware-classifications
- http://es.scribd.com/doc/37114969/Catalogo-de-Virus#scribd

En los **Anexos**, estarán adjuntadas imágenes de otros módulos del Antivirus Web, mi compañero Rafa ha sido el encargado principalmente de desarrollar los códigos de los Anexos.

Están adjuntados con la única finalidad de mostrar algo de código del Antivirus Web.

Teniendo que destacar, que no es la versión actual, sino la primera, y que por temas de privacidad de la empresa, no se mostrarán ni las firmas conseguidas, ni las listas blancas o negras.

Así mismo cabe destacar que ciertos mecanismos y procedimientos de búsqueda y borrado del Antivirus Web, no estarán presentes en el proyecto.

Anexos

Código del AntiVirus Web – Primera Versión

Escáner (index.php):
Carga de ficheros.def:

```
//CARGADO DE FICHEROS
function load_defs($file, $x, $debug) {

    $defs = file($file);
    $counter = 0;
    $counttop = sizeof($defs);

    while ($counter < $counttop) {
        $defs[$counter] = explode('     ', $defs[$counter]);
        $counter++;
    }
    if ($debug && $x=="viruses")
        echo '<p>Consultados <strong>' . sizeof($defs) . '</strong> virus de la base de datos</p>';
    if ($debug && $x=="signatures")
        echo '<p>Consultadas <strong>' . sizeof($defs) . '</strong> heurística de la base de datos</p>';

    return $defs;
}
```

Esta función simplemente abre los ficheros .def para la comparación de cadenas posterior (si hay debug, muestra un texto explicativo).

Recorrer carpetas y acceder a sus ficheros:

```
// ESCANEADO DE FICHEROS
function file_scan($folder, $defs, $signatures, $debug) {
    global $dircount, $report;
    $dircount++;
    if ($debug)
        $report .= "<p class=\"d\">Escaneando carpeta $folder ...</p>";
    if ($d = @dir($folder)) {
        while (false !== ($entry = $d->read())) {
            $isdir = @is_dir($folder.'/'.$entry);
            if (!$isdir and $entry!='.' and $entry!='..' and stripos($folder,'cuarentena')==false) {
                virus_check($folder.'/'.$entry,$defs,$signatures,$debug);
            } elseif ($isdir  and $entry!='.' and $entry!='..') {
                file_scan($folder.'/'.$entry,$defs,$signatures,$debug);
            }
        }
        $d->close();
    }
}
```

Se encarga de recorrer las carpetas desde la raíz (indicada en "config.php"), ignorando algunas excepciones añadidas por seguridad ante falsos positivos, y ver si son escaneables.

Comprobar la existencia de virus en ficheros:

```
// COMPROBACIÓN DE VIRUS/EXPRESIONES REGULARES
function virus_check($file, $defs, $signatures ,$debug) {
    global $filecount, $infected, $report, $CONFIG, $clean_files, $new_log,$link;

    $scannable = 0;
    foreach ($CONFIG['extensions'] as $ext) {
        if (substr($file,-3)==$ext)
            $scannable = 1;
    }

    if(basename($file) == "mylog.txt" || basename($file) == "inspector.php" || basename($file) == basename($_SERVER["SCRIPT_FILENAME"]))
        $scannable = 0;

    if ($scannable) {
        $filecount++;
        $data = file($file);
        $data = implode('\r\n', $data);
        $clean = 1;
        $print = 1;
        $file_infected = 0;

        // Comparación contra: Base de Datos de Virus
        for ($i = 0; $i < sizeof($defs); $i++) {
            $pos = stripos($data, trim($defs[$i][1]));

            if ($pos !== false ){

                $report .= '<p class="r">Infectado: ' . $file . ' (' . $defs[$i][0] . ')</p>';
                if($new_log==1)
                    shell_exec('echo "Infectado: ' . $file . '" >> mylog.txt');

                $report .= '<p class="r2">Cadena comprometida--> "' . trim($defs[$i][1]) . '" en la posición: ' . $pos . '</p>';
                if($new_log==1)
                    shell_exec('echo "Cadena comprometida--> ' . trim($defs[$i][1]) . ' en la poisición: ' . $pos . '" >> mylog.txt');

                $fecha = New DateTime();
                date_timestamp_set($fecha, filemtime($file));

                $report .= '<p class="r2">Última fecha modificación: ' . date_format($fecha, 'd-m-Y H:i:s') . '</p>';
                if($new_log==1)
                    shell_exec('echo "Última fecha modificación: ' . date_format($fecha, 'd-m-Y H:i:s') . '" >> mylog.txt');

                $owner = posix_getpwuid(fileowner($file));
                $permisos = fileperms($file);
```

```
        // Formatear permisos
        if (($permisos & 0xC000) == 0xC000) {
            $info = 's';
        } elseif (($permisos & 0xA000) == 0xA000) {
            $info = 'l';
        } elseif (($permisos & 0x8000) == 0x8000) {
            $info = '-';
        } elseif (($permisos & 0x6000) == 0x6000) {
            $info = 'b';
        } elseif (($permisos & 0x4000) == 0x4000) {
            $info = 'd';
        } elseif (($permisos & 0x2000) == 0x2000) {
            $info = 'c';
        } elseif (($permisos & 0x1000) == 0x1000) {
            $info = 'p';
        } else {
            $info = 'u';
        }

        $info .= (($permisos & 0x0100) ? 'r' : '-');
        $info .= (($permisos & 0x0080) ? 'w' : '-');
        $info .= (($permisos & 0x0040) ?
                    (($permisos & 0x0800) ? 's' : 'x' ) :
                    (($permisos & 0x0800) ? 'S' : '-'));

        $info .= (($permisos & 0x0020) ? 'r' : '-');
        $info .= (($permisos & 0x0010) ? 'w' : '-');
        $info .= (($permisos & 0x0008) ?
                    (($permisos & 0x0400) ? 's' : 'x' ) :
                    (($permisos & 0x0400) ? 'S' : '-'));

        $info .= (($permisos & 0x0004) ? 'r' : '-');
        $info .= (($permisos & 0x0002) ? 'w' : '-');
        $info .= (($permisos & 0x0001) ?
                    (($permisos & 0x0200) ? 't' : 'x' ) :
                    (($permisos & 0x0200) ? 'T' : '-'));

        $report .= '<p class="r2">Permisos: ' . substr(sprintf("%o",fileperms($file)),-4) . ' ' . $info . '</p>';
        if($new_log==1)
            shell_exec('echo "Permisos: ' . substr(sprintf("%o",fileperms($file)),-4) . ' ' . $info . '" >> mylog.txt');
        $report .= '<p class="r2">Propietario: ' . $owner['name'] . '</p>';
        if($new_log==1)
            shell_exec('echo "Propietario: ' . $owner['name'] . '\n" >> mylog.txt');
        $report .= '<a id="' . $file . '"><button>Ver Código</button></a><br><br>';

        $file_infected = 1;

        $clean = 0;
    }
}
```

```
        // Comparación contra: Expresiones regulares
        for ($i = 0; $i < sizeof($signatures); $i++) {
            $matches = preg_match(trim($signatures[$i][1]),$data);

            if ($matches !== 0){
                $report .= '<p class="r">Código malicioso: ' . $file . ' (' . $signatures[$i][0] . ')</p>';
                if($new_log==1)
                    shell_exec('echo "Código malicioso detectado: ' . $file . '\n" >> mylog.txt');
                $file_infected = 1;
                $clean = 0;
            }

        }


        if (($debug)&&($clean))
            $report .= '<p class="g">Limpio: ' . $file . '</p>';

        if($file_infected == 1){
                $md5=shell_exec('md5sum '.$file.'| cut -d " " -f -1');
                $nombre=basename($file);

                $nombre=trim($nombre);
                $md5=trim($md5);
                $file=trim($file);

                $sql="INSERT IGNORE INTO archivos (nombre, md5, ruta) VALUES ('".$nombre."','".$md5."','".$file."')";
                $resultado=mysqli_query($link,$sql) or die ('Fallo al hacer Insert'.mysqli_error());
                if(is_dir('cuarentena')==false)
                        mkdir('cuarentena',0777);
                copy($file,'cuarentena/'.$nombre);
                $infected++;
        }
    }
}
```

Esta es la parte más importante, la encargada de recorrer cada fichero del directorio, y comprobar contra cada posible cadena maliciosa alojada en “virus.def” y “signatures.def”.

Si existen coincidencias marca el fichero como peligroso (para asignarle un color rojo, añadirlo al diario y copiarlo a cuarentena) mientras que si parece no estar infectado lo marca como limpio (para asignarle un color verde y descartarlo).

Si es marcado como infectado, a parte, se genera una lista de detalles sobre el archivo, y se crean unos botones para poder acceder al “inspector.php” y decidir si modificarlo o borrarlo, de manera manual.

Eliminar el fichero seleccionado:

```
////////////////////////////////////////////////////////////////////////////

$file_to_remove = $_GET["r"];

if($file_to_remove!=''){

    $file_to_remove = htmlspecialchars($_GET["r"]);
    $shell_code = 'rm -f ' . $file_to_remove;
    shell_exec($shell_code);

}
```

Si se ha seleccionado un fichero para borrar desde el inspector, éste se encarga de eliminarlo.

Variables globales y carga de configuración:

```
////////////////////////////////////////////////////////////////////////////////////////////////////Configuración Inicial

$CONFIG = Array();
$CONFIG['debug'] = 0;
$CONFIG['scanpath'] = $_SERVER['DOCUMENT_ROOT'];
$CONFIG['extensions'] = Array();

@include("config.php");

if (!check_defs('virus.def'))
    trigger_error("Sobrescritura vulnerable en la base de datos de virus, porfavor cambia los permisos.", E_USER_ERROR);

if (!check_defs('signatures.def'))
    trigger_error("Sobrescritura vulnerable en la base de datos de virus, porfavor cambia los permisos.", E_USER_ERROR);

////////////////////////////////////////////////////////////////////////////////////////////////////Inicialización de variables

$report = '';
$dircount = 0;
$filecount = 0;
$infected = 0;
$new_log = 0;
```

Esta parte del código es la encargada de inicializar las variables globales del antivirus, y cargar los ".def" que servirán para detectar las infecciones.

Creación del diario una vez cada 24 horas:

```
//Comprobar si el fichero de log tiene mas de 7 dias y borrarlo

$today_ts = time();
$log_ctime = shell_exec("stat -c '%Z' mylog.txt");

$diff_sec = $today_ts - $log_ctime;
$diff_days = $diff_sec / (60 * 60 * 24);
$diff_days = abs($diff_days);
$diff_days = floor($diff_days);

if($diff_days>=1){
    shell_exec("rm -f mylog.txt");
    $new_log = 1;
}
```

Se encarga de comprobar si han pasado las 24 horas y si esto es cierto borra el archivo “mylog.txt” para que la siguiente pasada del escáner lo vuelva a crear nuevo.

Inspector (inspector.php):

Editar fichero:

```
$new_text = htmlspecialchars($_GET["nc"]);
file_put_contents($filename, $new_text);
```

Función encargada de cambiar el contenido del fichero que está siendo editado, si se ha apretado el botón “Guardar Cambios”.

Mostrar el fichero y ejecutarlo:

```
renderhead();

$filename = htmlspecialchars($_GET["p"]);

$fp = file($filename);

echo '<h1 id="prueba" class="top-labels">Archivo "</h1>';
echo '<h1>Código Original:</h1><br>';
echo '<textarea id="bloque1" class="col-md-10 col-md-offset-1 code" style="margin-top=0px">';

foreach ($fp as $num_línea => $fp) {
    echo htmlspecialchars($fp);
}
```

```
echo '</textarea><button id="save_btn" class="col-md-2 col-md-offset-3 btn btn-primary">Guardar Archivo</button>
<button id="delete_btn" class="col-md-2 col-md-offset-2 btn btn-danger">Eliminar Archivo</button><br>';

echo '<h1 style="display:inline-block" class="col-md-4 col-md-offset-4">Código Decodificado:</h1><br>';
echo '<div id="bloque2" class="col-md-10 col-md-offset-1 code" style="min-height:0px">';

$fp = file_get_contents($filename);

$cadena="preg_replace";

$php_code = str_replace("<?php","",$fp);
$php_code = str_replace("?>","",$php_code);
$php_code = str_replace("<?","",$php_code);
$php_code = preg_replace("/GIF.*?\n/","",$php_code);
$php_code = preg_replace('/\s+/', '', $php_code);
$php_code = preg_replace("/.*eval/"," eval",$php_code);
```

```
$busqueda=strpos($php_code,$cadena);
if($busqueda !== false){
    $segu = str_replace("eval","echo",$php_code);
    ob_start();
    eval($segu);
    $a = ob_get_clean();
    $text = stripcslashes(nl2br(htmlentities($a)));
    echo $text;
}else {
    $segu = str_replace("eval","echo",$php_code);
    ob_start();
    eval($segu);
    $a = ob_get_clean();
    $text = stripcslashes(nl2br(htmlentities($a)));
    echo $text;
}

echo '</div>';

echo '<div id="texto_intento" class="col-md-4 col-md-offset-1" style="display:none; padding-left:0px; margin-top:10px">Sin embargo,
prueba a ejecutar en <a href="http://ddecode.com/phpdecoder/" style="text-decoration:none">http://ddecode.com/phpdecoder/</a> el código original.</div>';
echo '<h1 style="display:inline-block" class="col-md-4 col-md-offset-4">Código Ejecutado:</h1><br>';
echo '<div id="bloque3" class="col-md-10 col-md-offset-1 code" style="max-height:800px; min-height:0px; margin-bottom:50px">';

if($text!='' || $php_code!=''){
    ob_start();
    eval($php_code);
    ob_end_flush();
}

echo '</div>';
```

En esta sección se crean 3 bloques, el encargado de mostrarnos el fichero original sin ninguna modificación, un segundo bloque con el contenido tras una decodificación, y por último un bloque para ver como sería lo que se visualizaría si se ejecutara dicho código. Si falla la decodificación te recomienda un enlace donde puedes intentar copiar y pegar el original y ver si ese decodificador consigue un resultado. Por último te muestra los botones de editar y borrar.

Función para la decodificación del código ofuscado:

```
//DECODIFICACIÓN DE LOS FICHEROS
function preg_decode($texto){
    $l1= preg_replace('/preg_replace\(\"\/\.\*\/e\"\,\"/', '', $texto);
    $l2= preg_replace('/\'.*/', '', $l1);
    $l3= 'print "'.$l2.'";';
    $l4= preg_replace('/^.*\'/' ,'', $l1);
    $l5= preg_replace('/\".*/' ,'' ,$l4);
    $l6= 'print "'.$l5.'";';
    $l7= preg_replace('/^.*?\'/','"',$l1);
    $l8= preg_replace('/\'.*/','"',$l7);
    ob_start();
    echo eval($l3);
    echo "$l8";
    echo eval($l6);
    $resultado=ob_get_clean();
    return $resultado;
}
```

Función utilizada junto la pagina de “UnPHP.net” para decodificar los ficheros ofuscados.

Configuración (config.php):

```
// MODO DEBUG
// ----------
// Esta opción nos permitirá ver el resultado del scaneo en
// profundidad, o simplemente un resumen a grandes rasgos.
// Resumen -> $CONFIG['debug'] = 0;
// Detalles -> $CONFIG['debug'] = 1;

$CONFIG['debug'] = 1;


// RUTA DEL DIRECTORIO A ESCANEAR
// ------------------------------
// Aquí se trabaja sobre el directorio que queremos escanear
// que por lo general es la raíz del servidor web en el que
// el antivirus trabaja, pero también se puede cambiar el
// directorio por otro diferente al que se tenga acceso.

$CONFIG['scanpath'] = $_SERVER['DOCUMENT_ROOT'];
```

```
// ARCHIVOS ESCANEABLES
// --------------------
// Aquí se especifícan que archivos se podrán escanear
// al pasar el antivirus, dado que puede ser que no te interese
// escanear ciertos archivos con un tipo de extensión concreta.
// Ejemplo : ".html" -> $CONFIG['extensions'][] = 'html';

// Archivos WEB

$CONFIG['extensions'][] = 'htm';
$CONFIG['extensions'][] = 'html';
$CONFIG['extensions'][] = 'shtm';
$CONFIG['extensions'][] = 'shtml';
$CONFIG['extensions'][] = 'xml';
$CONFIG['extensions'][] = 'json';
$CONFIG['extensions'][] = 'css';
$CONFIG['extensions'][] = 'js';

// Archivos PHP

$CONFIG['extensions'][] = 'php';
$CONFIG['extensions'][] = 'php3';
$CONFIG['extensions'][] = 'php4';
$CONFIG['extensions'][] = 'php5';

// Archivos de TEXTO

$CONFIG['extensions'][] = 'txt';
$CONFIG['extensions'][] = 'pdf';
$CONFIG['extensions'][] = 'rtf';
$CONFIG['extensions'][] = 'doc';
$CONFIG['extensions'][] = 'docx';
$CONFIG['extensions'][] = 'odt';
$CONFIG['extensions'][] = 'log';
$CONFIG['extensions'][] = 'conf';
$CONFIG['extensions'][] = 'config';

// Archivos de DATOS

$CONFIG['extensions'][] = 'db';
$CONFIG['extensions'][] = 'odb';
$CONFIG['extensions'][] = 'csv';
$CONFIG['extensions'][] = 'dat';
$CONFIG['extensions'][] = 'sql';

// Archivos de PROGRAMACIÓN

$CONFIG['extensions'][] = 'perl';
$CONFIG['extensions'][] = 'sh';
$CONFIG['extensions'][] = 'bat';
$CONFIG['extensions'][] = 'exe';
```

```
// Archivos de IMAGENES

$CONFIG['extensions'][] = 'png';
$CONFIG['extensions'][] = 'jpeg';
$CONFIG['extensions'][] = 'jpg';
$CONFIG['extensions'][] = 'gif';
$CONFIG['extensions'][] = 'bmp';
$CONFIG['extensions'][] = 'raw';
```

Este fichero es el que contiene la configuración del antivirus, que nos permite decirle al antivirus que archivos abrir, la ruta de escaneo, y la profundidad de la información a mostrar por el antivirus.

Definición de expresiones regulares (signatures.def):

Ejemplo:

```
Dangerous regex matching        /\<\?php.*(eval\((gzinflate|base64_decode)\((gzinflate|base64_decode))|(if.*\(isset)/
```

Simples expresiones regulares con su texto asignado que nos permiten localizar posibles peligros, si al pasar el "preg_match" se encuentra alguna coincidencia en los archivos del servidor web.

Definición de Virus (virus.def):

Ejemplo:

```
Suspicious code at 90%  preg_replace('/.*/e"
Suspicious code at 90%  preg_replace('/.*/e'
Suspicious code at 90%  preg_replace("\x2F\x2E\x2A\x2F\x65"
Suspicious code at 90%  \x65\x76\x61\x6C\x28
Suspicious code at 90%  \x67\x7A\x69\x6E\x66\x6C\x61\x74\x65\x28
Suspicious code at 90%  \x62\x61\x73\x65\x36\x34\x5F\x64\x65\x63\x6F\x64\x65\x28
Suspicious code at 90%  \x2F\x2E\x2A\x2F\x65
Suspicious code at 90%  die(PHP_OS.chr(49).chr(48).chr(43).md5(0987654321
Suspicious code at 90%  strrev('edoced_46esab')
Suspicious code at 90%  if (( preg_match ('/Gecko|MSIE/i', $wp_lcr95795) && !preg_match ('/bot/i',
Suspicious code at 90%  $wp_lcr95795))){
Suspicious code at 90%  PCT4BA6ODSE
Suspicious code at 90%  eval(base64_decode(
Suspicious code at 90%  eval((base64_decode(
Suspicious code at 90%  system(base64_decode(
Suspicious code at 90%  eval(gzinflate(
Suspicious code at 90%  eval((gzinflate(
Suspicious code at 90%  eval(gzuncompress(base64_decode(
Suspicious code at 90%  edoced_46esab
Suspicious code at 90%  shell_exec (esto marca demasiados scripts)
```

Estas cadenas de texto, han sido encontradas como causantes de infección en web, por tanto han sido clasificadas como maliciosas, y su completa detección en un archivo nos indica que casi casi seguro dicho archivo es un virus, y ha sido montado para que se refleje como texto del virus la parte izquierda de los pares de datos.

AntiVirus Web - Pearl

Se cambió el antiguo Antivirus en PHP y JavaScript por PERL para que fuera compatible con servidores más antiguos.

```
sub recursion {
 my ($dir) = @_;
 my (@list);
 if (!opendir(I, "$dir")) {
  return if $! =~ /no se encuentra el fichero/i;
  print STDERR "$dir: Permiso denegado: $!, pasando al siguiente\n";
  return;
 }
 @list = readdir(I);
 closedir(I);
 for my $mfile (@list) {
  next if $mfile =~ /^\.\.?$/;
  my $cf = $currentfile = "$dir/$mfile";
  $cf =~ s/'/'"'"'/g;
  $cf = "'$cf'";
  ##### LISTA BLANCA #####
  if (-f $currentfile) {
   my $checksum = `$md5sum $cf`;
   chomp($checksum);
   $checksum =~ s/\s.*//;
   if ($white{$checksum}) {
    next;
   }
  }
         ##### LISTA NEGRA #####
  if (-f $currentfile) {
   my $checksum = `$md5sum $cf`;
   chomp($checksum);
   $checksum =~ s/\s.*//;
   if ($black{$checksum}) {
    print "$currentfile: Archivo en Lista Negra!\n";
    next;
   }
  }
  ####################
  if (-d $currentfile && ! -l $currentfile) {
   &recursion($currentfile);
   next;
  }
```

```
 next if ! -f $currentfile;
 if ($mfile =~ /$scripts/) {
  &scanfile($currentfile, $scriptpat);
  } elsif ($mfile =~ /$access/) {
   &scanfile($currentfile, $accesspat);
  }
  next if -s $currentfile > 1000000 || -s $currentfile < 2000;
  my $type = `$file $cf`;
 }
}
sub scanfile {
 my ($currentfile, $patterns) = @_;
 open(I, "<$currentfile") || next;
 my $linecount = 1;
 while(<I>) {
  chomp;
  if ($_ =~ /$patterns/) {
   my $pat = $1;
   my $string = $_;
    print "$currentfile: Cadena sospechosa ($pat):\n $string\n\n";
    last;
   }
   last if $linecount++ > $MAXLINES;
  }
  close(I);
 }
 sub inithelpers {
  if (-x '/usr/bin/md5sum') {
   $md5sum = '/usr/bin/md5sum';
   } elsif (-x '/sbin/md5') {
    $md5sum = '/sbin/md5 -q';
   }
   for my $x (('/bin', '/usr/bin')) {
    if (-x "$x/strings") {
     $strings = "$x/strings";
    }
    if (-x "$x/file") {
     $file = "$x/file";
    }
   }
   die "No se ha podido encontrar una herramienta para comparara los codigos md5" if !$md5sum;
   die "No se ha encontrado ninguna cadena de comparacion" if !$strings;
   die "No se ha podido encontrar el archivo especificado" if !$file;
  }
```