CAPÍTULO III: LAS VULNERABILIDADES

Este capítulo no tiene como objetivo analizar cuál es la mejor herramienta para el escaneo de vulnerabilidades, tampoco decir cuál es la mejor para Linux, Windows. El objetivo es poder brindar al lector un mayor conocimiento sobre el análisis de las vulnerabilidades, los tipos, las diferentes formas de escaneos y la detección de las diferentes vulnerabilidades y como poder resolverlas.

3.1. Introducción al análisis de vulnerabilidades

Definiendo a muy grandes rasgos que es una vulnerabilidad, una vulnerabilidad de una manera muy general es un fallo en un sistema que puede ser explotada por un atacante generando un riesgo para la organización o para el mismo sistema.

Existen dos tipos de vulnerabilidades que se mencionan a continuación:

- Las lógicas
- Las físicas

3.1.1. Vulnerabilidades físicas

Las vulnerabilidades físicas son las que van a afectar a la infraestructura de la organización de manera física y se pueden mencionar en este tipo de clasificación a los desastres naturales, como ejemplo se podría mencionar una vulnerabilidad alta de este tipo si se vive en una zona de alto riesgo de sismos, ya que puede presentarse una negación en el servicio, una afectación en la disponibilidad y a partir de ahí se podría empezar con problemas. Si la organización está en una zona que generalmente se inunda, se tiene también otro tipo de vulnerabilidad.

Otra de las opciones físicas son los controles de acceso, en muchas ocasiones se tiene los accesos a la infraestructura crítica y no se tiene los accesos pertinentes, cualquier persona podría abrir una puerta, podría entrar y constituye un gran riesgo para la organización porque cualquier usuario podría ingresar con una USB y copiar información, podría infectar la misma infraestructura.

3.1.2. Vulnerabilidades lógicas

Las vulnerabilidades lógicas son las que van a afectar directamente la infraestructura y el desarrollo de la operación de estos, estas pueden ser de:

- Configuración
- Actualización
- Desarrollo

Las de **configuración** en el sistema operativo, pueden ser las configuraciones por defecto del sistema o incluso de algunas aplicaciones del servidor que se tenga expuesta, puede ser también la configuración de algunos firewalls que no está gestionado de una manera correcta y también de infraestructura perimetral.

Las vulnerabilidades de **actualización**, en muchas ocasiones hay empresas que no actualizan sus sistemas, van saliendo las vulnerabilidades y es un punto que se debe tomar en cuenta.

Actualmente, en los equipos XP de Windows no se les está dando soporte y muchas empresas tienen estos sistemas, cuando se realiza un escaneo en una determinada red al no tener soporte estos equipos ya son vulnerables.

Las vulnerabilidades de **desarrollo**, aquí se puede mencionar las inyecciones de código en SQL, Cross Site Scripting, esto puede variar dependiendo del tipo de aplicación, la validación de los datos. Cada escáner de vulnerabilidades utiliza distintas escalas, en estas escalas se va a poder auditar en base a una metodología de pruebas de penetración, de cumplimiento, si se va a auditar una red interna o una aplicación web, es muy distinto el escáner que se va a utilizar.

3.1.3. Escáneres de vulnerabilidades

Existen una gran gama de escáner de vulnerabilidades, muchos son de pago otros son gratuitos y se los puede utilizar sin mayor problema para su ejecución, hay escáneres como **Acunetix** que son muy buenos en la parte web y no sólo permiten escanear, también permiten la explotación real de ciertas vulnerabilidades o incluso la comprobación de estas.

Muchos de los escáneres web trabajan con proxys y a partir de estos se realiza la captura de las tramas de la información y se puede realizar la modificación. Algunos escáneres utilizan métodos que van a permitir listar el contenido del servidor de acuerdo a los directorios más conocidos, uno de esos escáneres es "**Acunetix**", el cual es una herramienta que está diseñada con el objetivo de encontrar agujeros de seguridad en las aplicaciones web, los cuales puedan ser aprovechados por determinados atacantes para accesar a los sistemas y la información.

También hay herramientas escáner como **netsparker** y **ProxyStrike** que permiten detectar vulnerabilidades. El caso de netsparker es un escáner de pago y ProxyStrike es gratuito, el cual permite identificar inyecciones de SQL Y Cross Site Scripting y el escáner "**VEGA**" que vienen incluidos en Kali Linux, al igual que ProxyStrike y a partir de ahí se puede realizar un propio escaneo.

Hay escáner para CMS, esto se debe al número de vulnerabilidades que se han dado a conocer y sobre de eso la posibilidad de explotar inyecciones SQL, la gestión del administrador entre otras cosas.

También existen escáner de vulnerabilidades en lo referente a sistemas operativos y también algunos son utilizados en la parte web, uno de los más completos es Nessus que se puede integrar con sistemas operativos como Android, se puede escanear desde el teléfono Android alguna red, buscando las vulnerabilidades y a partir de allí empezar a gestionar los resultados, tiene incluso opciones para virtualización.

Nexpose es un escáner que viene de la familia de "**Metasploit**", el cual permite realizar un escaneo y en algunos casos se puede exportar en herramientas como Metasploit en su versión pro o exprés y a partir de ahí tener un vector de ataque más puntual.

También hay tipos de escaneos con herramientas como **LanGuard** u **OpenVAS**, que pueden permitir utilizar el escáner sin tener credenciales o con las credenciales

del administrador, aparte de realizar un escaneo, evaluar las políticas tanto del equipo, como las de seguridad e incluso se podría empezar a ver si realmente las áreas de administración están teniendo un cumplimiento de sus políticas internas y analizar los procedimientos para la gestión de los equipos cuando se hace una prueba de penetración.

Estos escáneres funcionan de una manera sencilla, cuando se da a conocer una vulnerabilidad que tenga gran relevancia se desarrollan las firmas que van a validar estas vulnerabilidades, los escáneres se actualizan directamente, bajan las firmas y a partir de allí se puede realizar el escaneo.

3.2. Tipos de vulnerabilidades

Existen algunos tipos de vulnerabilidades que son mecanismos aprovechados por los atacantes para infectar una red o robar información entre los cuales se puede mencionar a los siguientes tipos:

3.2.1. Desbordamiento de buffer

El desbordamiento de buffer ocurre cuando el programador no controla el espacio de memoria del programa, entonces alguna persona puede introducir su propio código en ese espacio de memoria y la máquina lo va a ejecutar antes que cualquier otra tarea, por ejemplo, eso normalmente se da mucho con los payloads, en los cuales se inyectan cierta cantidad de memoria o inclusive dentro de los backdoor o puerta trasera, los cuales inyectan en la memoria RAM un cierto o una cierta cantidad de código, el cual se arranca antes, inclusive de arrancar toda la parte del sistema operativo o de algunos de los archivos dentro del mismo sistema que se utilizan para arrancar de manera normal. La Figura 14 muestra un ejemplo de desbordamiento de buffer.

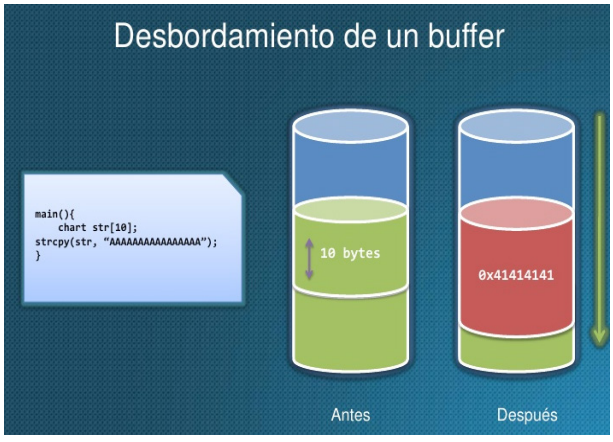

Figura 14. Desbordamiento de buffer.

Fuente: https://www.slideshare.net/RevistaSG/ups-cdigo-inseguro-deteccin-explotacin-y-mitigacin-de-vulnerabilidades-en-software

3.2.2. Errores de configuración

Otra de las principales vulnerabilidades, son los errores de configuración, se puede mencionar, por ejemplo, los password por default, password débiles, usuarios con demasiados privilegios e inclusive la utilización de protocolos de encriptación obsoletos, normalmente una de las cosas más típicas en las organizaciones es que utilizan algún sistema de encriptación web, lo cual con una aplicación de teléfono celular se puede crackear en menos de 10 o 15 segundos o inclusive con una laptop.

Otro error de vulnerabilidad puede ser algún protocolo de SSH que no se haya parchado o actualizado, por ejemplo, con alguna especie de vulnerabilidad se estaría utilizando algún protocolo de encriptación ya sea obsoleto o inseguro, pero normalmente la parte de errores de configuración provienen de la parte del password default, cuando se ingresa a una red con la IP por ejemplo, 198.X.X.X y si se tiene la posibilidad de ingresar a Google y buscar dentro del mismo, lo que sería el usuario y la contraseña por default, se puede cambiar la configuración causando un daño a la empresa.

3.2.3. Errores web

Otros tipos de vulnerabilidades son las WEB, aquí simple y sencillamente se tiene errores de validación de input, Scripts inseguros, errores de configuración de aplicaciones web, entre algunas otras situaciones, que a final de cuenta todos y cada uno de esos errores son los medios para algún ataque de XSS (Cross Site Scripting) o inyección SQL.

Según Cañon Parada (2015), la inyección por SQL es uno de los ataques más utilizados en la actualidad, consiste en acceder a las tablas de la base de datos incluyendo información sobre el usuario y su clave, este ataque está caracterizado porque es fácil de ejecutar porque modifica la cadena de consulta SQL hacia la base de datos.

También Cañon Parada (2015), indica que los ataques tipos Cross Site Scripting consisten en infectar un sitio web mediante scripts maliciosos con el objetivo de obtener acceso a una determinada cuenta de usuario.

3.2.4. Errores de protocolo

Por último, se tiene la parte de las vulnerabilidades de protocolos, existen diversas cantidades de protocolos que normalmente fueron definidos sin la necesidad o sin tener en cuenta precisamente la parte de la seguridad y en muchas veces no se preveo el crecimiento que estos iban a tener y como el internet no estaba preparado para ser tan grande, no se pensó en la parte de la seguridad.

Algunos de los protocolos pueden ser un simple HTTP, el cual no es seguro, dado que realiza solamente la parte de la autenticación, pero sin la encriptación de los datos que a final intercambia, esto puede ser necesario en algunos ambientes, por ejemplo, en las páginas visitadas simple y sencillamente por los usuarios, pero cuando se realizan transacciones bancarias normalmente la parte de este protocolo resultaría muy inseguro, probablemente se requiera de alguna otra acción como

sería algún certificado SSL o TSL, etc. Normalmente el mayor problema es cuando se define el marco de seguridad que tienen las fallas, por ejemplo, alguna especie de utilización de sistema web.

3.2.5. Aprovechamiento de las vulnerabilidades

Normalmente existen dos formas de aprovechar las vulnerabilidades como se muestra a continuación:

- Forma remota
- Ingeniería social

En la forma remota se llega mediante una computadora y se empieza a hacer análisis, ataques a un cierto servidor y tratar de vulnerarlo, si se logra el acceso, quiere decir que ya se hizo alguna explotación remota.

En la parte de la ingeniería social, alguien puede ayudar de manera interna, una persona dentro de la organización puede ayudar a realizar un acceso no permitido.

También existen las partes de ataques directo una vulnerabilidad de forma remota utilizando internet, se aprovecha de que algún software o servicio tiene un puerto abierto volviéndose vulnerable. Otra de las partes es engañando a un usuario, aplicando ingeniería social con alguna memoria o algún archivo infectado se puede aprovechar de alguna vulnerabilidad que se encuentra dentro del mismo sistema.

3.3. Detección de vulnerabilidades

Las vulnerabilidades pueden ser detectadas mediante herramientas de detección, realizar un escaneo de puertos con el objetivo de verificar cuales están abiertos para intentar obtener información sobre el servicio que se encuentre corriendo en ese momento y con esta información buscar vulnerabilidades asociadas precisamente a esos servicios. Se tienen tres formas de detectarse.

1. Escáner de vulnerabilidades.
2. Análisis manuales.
3. Consultando información.

A través del **escáner de vulnerabilidades** se tienen herramientas como Nikto la cual funciona buscando fallos en base a servidores, Nessus, Nmap, etc. Una de las ventajas de la parte del escáner de vulnerabilidades, es que funcionan de manera automática, trabajan ubicando un rango de direcciones IP e inicia el escaneo, la máquina realiza todo el proceso prácticamente sola.

En la parte de los **análisis manuales** es muy importante realizarlos, ya que, todos los análisis automáticos no detectan de forma automática todas las vulnerabilidades que se pueda tener en un sistema, entonces, también es necesario realizar algún análisis manual dentro de las vulnerabilidades encontradas con la finalidad de evitar que se pueda escapar o dejar como tal cabo suelta.

Otra de las partes es **consultar información**, en la parte precisamente de ocultar información se tiene alguna especie de Google hacking por así mencionarlo o algo

por el estilo, utilizar simple servicios, de lo que sería la búsqueda de información en red para poder realizar o encontrar información que pueda servir a la organización, de identificación de algunas vulnerabilidades que puedan tener todos los servicios.

3.4. Métodos de escaneo de vulnerabilidades

Existen varios métodos de escaneo para poder realizar análisis de vulnerabilidades que se mencionan a continuación:

Caja blanca

El método de escaneo de caja blanca tiene una visión total de la red a analizar, así como, acceso a todos los equipos como súper usuario, aquí es donde se tiene la parte de toda la administración de los servicios, la parte de análisis de caja blanca actúa como un usuario legítimo dentro de la red, que puede utilizar los servicios de diversas formas a la que otra persona los pueda estar utilizando. De una manera más detallada, este método utilizará ciertos usuarios con ciertos privilegios dentro de la red y accesando a los servicios, dentro de los productos, dentro de los softwares que se quieren auditar y así poder verificar si se puede realizar alguna acción adicional en base los privilegios que se han brindado.

Caja negra

También existe el método de escaneo de caja negra, aquí es donde normalmente se proporciona información de acceso de red, aquí a los analistas les van a proporcionar sólo información de acceso a red o al sistema, por ejemplo, una sola dirección IP, algún nombre de alguna empresa, etc., a partir de aquí empieza como tal a buscar información, todo lo posible relacionado para la exploración y así poder obtener la mayor cantidad de información posible de dicha dirección IP, del resto de los equipos probablemente que se encuentran dentro de algún rango de direcciones IP asociado, aquí no se realiza ninguna instrucción, solamente se detecta y se documenta la vulnerabilidad.

Hay una diferencia en lo que sería un método de escaneo de análisis vulnerabilidades y un pentesting, en el primero se encuentra las vulnerabilidades y se las documenta, en cambio en el pentesting se busca explotar dichas vulnerabilidades.

3.5. Remediación de vulnerabilidades

En este punto se va a analizar como remediar las vulnerabilidades, se ha analizado en capítulos anteriores las amenazas, como identificar las vulnerabilidades, como clasificar los activos e identificar la amenaza que puede afectar dichos activos, una vez que se ha logrado detectar las vulnerabilidades hay una serie de pasos para tratar de remediarlas que se indican a continuación.

3.5.1. Análisis de activos

Existe un ciclo de vulnerabilidades o de remediación el cual Inicia con la parte de la realización de un inventario y sobre todo la parte importante, la categorización de activos, aquí para corregir dichas vulnerabilidades se debe entender que esos activos

tanto pc, servidores, impresoras, todo lo que podría catalogarse como un activo se realiza para tener un orden de los sistemas, por ejemplo, mantener una lista de direcciones IP que tienen los dispositivos o bien para descubrir, qué equipos se han conectado a la red sin ser detectados, para eso va a servir la parte de la realización de un inventario y la categorización de activos.

3.5.2. Escanear sistemas para detectar vulnerabilidades

Otra de las partes fundamentales dentro del ciclo de remediación de vulnerabilidades, es la parte de escanear los sistemas para detectar fallos, en este paso el escáner normalmente revisará ya sea el software, configuraciones o dispositivos que tenga cada dirección IP y determinará si se tiene alguna vulnerabilidad reportada sobre dicho servicio o software o alguna especie de configuración. Igualmente, si se está haciendo un escaneo de cumplimiento de normas, el escáner revisará el registro de la máquina para detectar las configuraciones que son inseguras. Hay que tomar en cuenta que el escáner necesita tener acceso de administrador a las máquinas que se van a estar escaneando, al terminar dicho escáner va a generar un reporte muy completo de todas las vulnerabilidades detectadas.

3.5.3. Identificar vulnerabilidades

Otro de los puntos muy importantes en la parte del ciclo de remediación de vulnerabilidades, es verificar vulnerabilidades dentro del inventario, una cosa muy importante es escanear los sistemas para detectar vulnerabilidades y otra cosa muy distinta es precisamente verificar vulnerabilidades dentro del inventario de los activos de la empresa. Esto consiste en identificar que las vulnerabilidades que son detectadas por el escáner, primeramente, sean relevantes, generalmente los escáneres pueden generar tener falsos positivos, si un escáner genera demasiados falsos positivos, va a costar muchas horas de trabajo en remediaciones que a final de cuenta no son innecesarias.

3.5.4. Clasificar y priorizar riesgos

Es imposible arreglar todas las vulnerabilidades detectadas, es por eso, que es necesario clasificar y sobretodo priorizar el riesgo que está impondría en la organización. No se puede arreglar todas las fallas encontradas porque se tiene poco tiempo, poco personal y sobre todo poco dinero, por esto la organización se debe enfocar en arreglar primero las vulnerabilidades más graves en sistemas críticos, pero para esto se debe diseñar un esquema de prioridad que combine el nivel de severidad de una vulnerabilidad y qué tan importante es el sistema para la empresa.

Los escáneres inclusive en ocasiones permiten crear un esquema automáticamente como el que se muestra en la Figura 15, que viene a ser una matriz de riesgos.

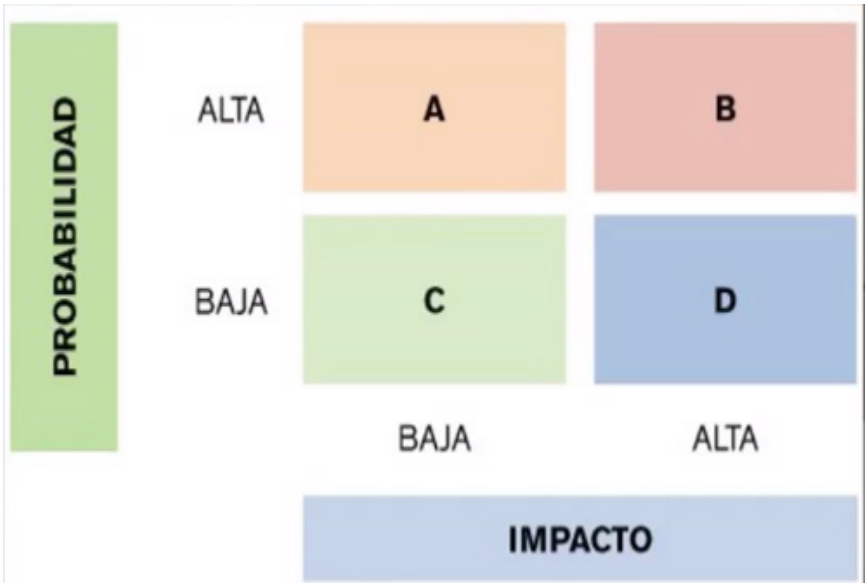

Figura 15. Matriz de riesgos**.**
Fuente: https://www.ics.dait.com.mx

3.5.5. Probar parches y configuraciones

Una vez que se ha detectado las vulnerabilidades dentro del sistema, del inventario y se ha clasificado y priorizado los riesgos que pueden incurrir estas vulnerabilidades, el siguiente paso es probar parches y cambios de configuración. El proceso de parcheo puede poner en riesgo el sistema de la organización, ya que el software parcheado puede traer inclusive errores que aún no han sido detectados.

El parcheo se debe instalar principalmente en una sola máquina y hacer pruebas para verificar si se llega a detectar algún problema, con esto poder evitar al haber instalado en todos los sistemas o en todos los dispositivos que ese error se propague. Hay que tomar en cuenta que el software que está haciendo parchado puede venir con configuraciones por default, ya que probablemente se tendría que ajustarlo nuevamente.

Es importante también descargar los parches de sitios oficiales del fabricante y sobre todo no usar parches de sitios de terceros, como, por ejemplo, up to down o zenet, etc., los cuales al final pueden traer algún software o malware o versiones anteriores con errores.

Los reportes de dicho escáner se incluyen en algunas ocasiones, instrucciones detalladas de cómo proceder con el parcheo o con el cambio de configuración que el sistema requiere para poder estar más seguro.

3.5.6. Aplicar parches y configuraciones

Una vez que ya se ha probado los parches y los cambios de configuración, es necesario aplicarlos, cuando se ha comprobado que dichos partes funcionan correctamente

en una máquina hay que proceder a implementarlos en todas las demás máquinas en la red, dependiendo del tamaño de la red este es el paso más laborioso de la administración de vulnerabilidades, hay soluciones de implementación de manera automáticas de parches que se pueden utilizar, hay que recordar que se tiene que probar primeramente el parche en una sola máquina antes de proceder a instalarlos en las demás.

Algunos escáneres tienen la posibilidad de generar automáticamente tickets para que sean asignados a ingenieros encargados de remediación y se las pueda dar un seguimiento a todo el proceso.

3.5.7. Aplicar parches y configuraciones

Una vez que ya se ha implementado todas las partes, se ha escaneado, clasificado riesgos, buscado los parches, aplicado dichos parches, la última parte es volver a escanear para verificar el parcheo, esto una vez que ya se haya parchado todos los sistemas o se hayan cambiado sus configuraciones inseguras, se debe escanear toda la red de nuevo para asegurarse que los parches estén adecuadamente instalados y no hayan faltado equipos, inclusive con toda esta situación de que se haya realizado un parcheo o una actualización es necesario volver a escanear para verificar si esos parches quedaron debidamente aplicados o bien generaron nuevas vulnerabilidades.