Análisis de Vulnerabilidades en un Entorno de Red Virtual: Implementación de Nessus como Herramienta de Evaluación

Vulnerability Analysis in a Virtual Network Environment: Implementing Nessus as an Assessment Tool

Jhon Alexander Munarco Santos y Cristopher Iván Zambrano Córdoba
Departamento de Ciencias de la Computación, Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador.
{jamunarco, cizambrano1}@espe.edu.ec

Resumen

Este artículo examina las vulnerabilidades en entornos de redes virtuales, en un contexto donde las organizaciones tienen una creciente necesidad de ciberseguridad y optan por soluciones de virtualización. Estos entornos presentan desafíos técnicos, como la complejidad de la infraestructura y la asignación de recursos, que pueden generar vulnerabilidades que no se suelen presentar en redes físicas. Se utilizó la herramienta Nessus para realizar auditorías de seguridad, con la identificación de diversas vulnerabilidades, establecer distintas medidas correctivas, como actualizaciones de software y configuraciones de seguridad más rígidas. Tambien se ha realizado una comparativa con otras herramientas de análisis de vulnerabilidades y de pruebas manuales, para determinar el alcance que pueda dar la herramienta Nessus. Los análisis realizados en una red virtual controlada han demostrado la importancia de un enfoque holístico para proteger la infraestructura de TI, al mitigar los riesgos asociados con configuraciones incorrectas. Este enfoque es primordial para afianzar la seguridad de datos y la integridad operativa en un entorno cada vez más digitalizado.

Palabras clave: *Ataques, Cifrado, Vulnerabilidades, Seguridad, Virtualización*

Abstract

This article examines vulnerabilities in virtual network environments, in a context where organizations have a growing need for cybersecurity and opt for virtualization solutions. These environments present technical challenges such as infrastructure complexity and resource allocation, which can lead to vulnerabilities that are not typically present in physical networks. The Nessus tool was used to perform security audits to identify various vulnerabilities, to establish various corrective actions such as software updates and more rigid security configurations. A comparison has also been made with other vulnerability analysis and manual testing tools to determine the scope that the Nessus tool can give. Analyses performed on a simulated virtual network have demonstrated the importance of a holistic approach to protecting IT infrastructure by mitigating the risks associated with misconfigurations. This approach is paramount to strengthening data security and operational integrity in an increasingly digitized environment.

Keywords: *Attacks, Encryption, Vulnerabilities, Security, Virtualization*

Fecha de Recepción: 05/08/2024 - Aceptado: 24/09/2024 - Publicado: 30/09/2024
ISSN: 2477-9253 – DOI: http://dx.doi.org/10.24133/RCSD.VOL09.N03.2024.05

I. Introducción

En la actualidad, la necesidad de las redes y los sistemas informáticos es cada vez mayor, esto conlleva a que la ciberseguridad se convierta en una prioridad clave para organizaciones de todo tipo. Las empresas, en su necesidad de optimizar recursos y mejorar su eficiencia operativa, recurren ampliamente a soluciones de virtualización. No obstante, con el aumento de entornos virtuales, surge la creciente necesidad de evaluar y mitigar aquellas vulnerabilidades inherentes para estos entornos (McGhin et al., 2019). Un proceso de evaluación es fundamental en la identificación de aquellos puntos débiles en la infraestructura virtual antes que personas mal intencionadas puedan explotarlos.

Un análisis de vulnerabilidades en un entorno de red virtual es una tarea recomendada que sirve para mantener la integridad, confidencialidad y disponibilidad de datos y sistemas. Con las redes virtuales, las cuales ofrecen flexibilidad y escalabilidad, se tienen presente nuevos desafíos los cuales necesitan un enfoque avanzado en materia de seguridad. La arquitectura virtual, que suele ser compartida entre múltiples sistemas y usuarios, puede tener múltiples vulnerabilidades que no siempre son visibles en entornos más comunes. Esta compleja situación añade una capa adicional de riesgo en la cual las organizaciones deben gestionar de manera eficiente para proteger sus sistemas (Lallie et al., 2020).

Las soluciones más comunes de seguridad no siempre son la mejor protección de forma eficiente en entornos virtuales. Las amenazas a las que se enfrentan las redes virtuales pueden ser completamente distintas de las que se utilizan para redes físicas. Por tanto, esto requiere de herramientas especializadas para la identificación de este tipo de riesgos. La capacidad de realizar un análisis profundo de las vulnerabilidades presentes en un entorno virtual puede marcar la diferencia entre prevenir un ataque cibernético o sufrir las consecuencias de una brecha de seguridad (Jeon & Kim, 2021).

Este estudio está enfocado en el uso de Nessus, una herramienta muy conocida en el mundo para análisis de vulnerabilidades, y sobre todo sus técnicas que utiliza para evaluar los riesgos en redes virtuales. Nessus, es acreditada por su capacidad para escanear una gran cantidad de vulnerabilidades, ofrece una solución muy sólida para identificar los puntos débiles en aquellas infraestructuras virtualizadas. Cuando se implementa esta herramienta en entornos virtuales, las empresas pueden conseguir una visión clara de aquellos riesgos a los que están expuestos y con ello tomar medidas correctivas antes de que los atacantes puedan aprovecharse (Chatterjee & Thekdi, 2020).

Con el uso de Nessus, se puede realizar un análisis a detalle de las vulnerabilidades presentes en los sistemas virtualizados. Esto incluye la detección de aquellas configuraciones incorrectas, software no actualizado y puntos de acceso o puertos expuestos. Con estos factores, se puede ser víctima por ciberdelincuentes si no se tratan de manera oportuna (Bays et al., 2015). El informe que se generó por la herramienta proporciona un panorama claro de aquellas vulnerabilidades que se detectan, permitiendo a los grupos de seguridad tomar como prioridad las acciones necesarias para proteger la infraestructura de la red (Russo et al., 2019).

Se ha realizado un enfoque complementario al análisis de vulnerabilidades, con la utilización de otras herramientas adicionales y técnicas manuales, como pruebas controladas utilizando Metasploit para confirmar si las vulnerabilidades detectadas eran reales para fortalecer la precisión y la confiabilidad de los resultados que se obtiene con Nessus. Con esta metodología se pretende abordar la problemática de posibles falsos positivos, una limitación de las herramientas automatizadas de escaneo (Moreno, 2013). Se ha utilizado las herramientas OpenVAS, Nmap y validaciones manuales, los cuales permiten verificar y contrastar los hallazgos, aumentando la robustez del análisis.

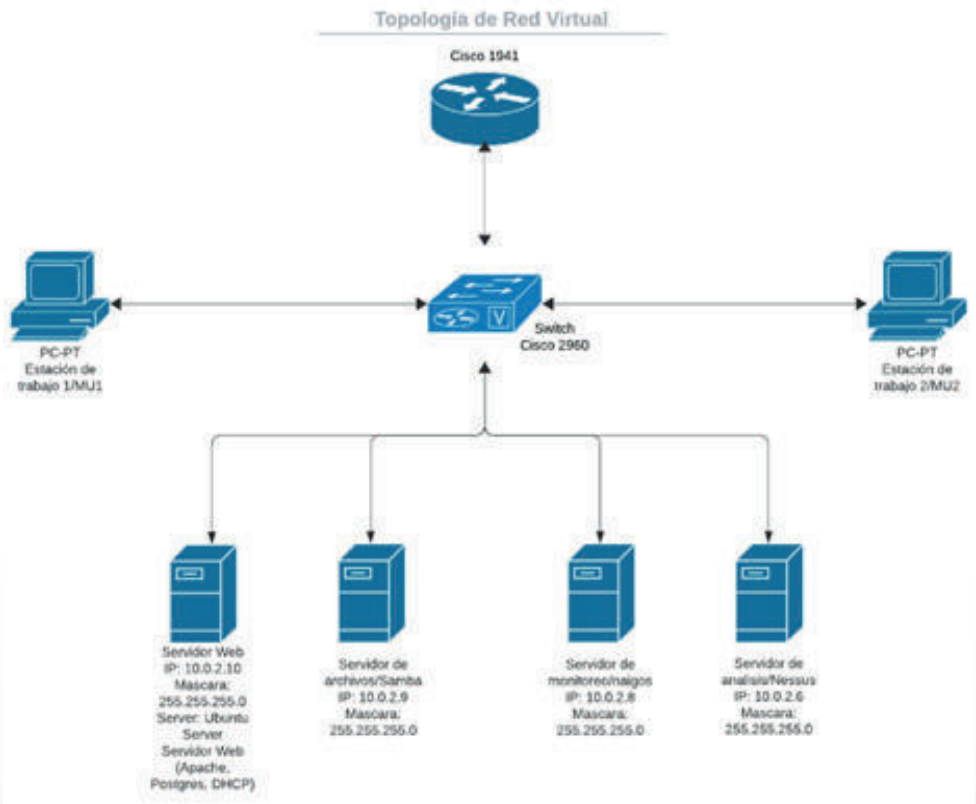
Por último, se discutirán aquellas vulnerabilidades específicas que se encontraron en entornos de red virtual, así como los riesgos que pueden estar asociados a estas. Además, se analizarán aquellas posibles soluciones para minimizar estos riesgos, legitimando que las empresas puedan protegerse y mantener la confianza en sus sistemas. Con un enfoque integral de la herramienta Nessus, se permite que las organizaciones no solo realicen detecciones, sino también solucionen de manera efectiva las vulnerabilidades antes de que puedan ser vulneradas (Yuri Baturin, 2008).

II. Materiales y Métodos

En esta sección se detalla la metodología que ha sido utilizada para llevar a cabo el estudio sobre la identificación y mitigación de vulnerabilidades en entornos corporativo simulados. Este estudio se basa en un enfoque sistemático que integra la configuración de una red virtual en Linux y el uso de herramientas avanzadas de escaneo de vulnerabilidades.

Para ello se ha creado una topología tipo estrella como se ilustra en la Figura 1, que muestra cómo debería conformarse la red virtual. Para esta topología, todos los dispositivos de la red están conectados a un dispositivo central, que para este caso es un switch de capa 2. El conmutador actúa como el punto central de la red, por donde todos los datos deben pasar a través de él para pasar de un dispositivo a otro.

Figura 1: *Topología experiencial con seis máquinas virtuales que conforman la red virtual*



2.1. Revisión literaria sobre el análisis de vulnerabilidades

La virtualización ha permitido crear entornos aislados que simulan hardware físico, lo cual también ha originado nuevas vulnerabilidades. Según Ríos Serrano (2024) un análisis a profundidad de la seguridad en redes virtuales es primordial para identificar amenazas y vulnerabilidades mediante herramientas de código abierto. Este estudio enfatiza la importancia de comprender los protocolos de comunicación y cómo se pueden utilizar para detectar intrusiones en entornos virtuales (Serrano, 2024).

Distintas metodologías han sido presentadas para el análisis de vulnerabilidades en redes virtuales. Un artículo creado por Quishpe Malla (2016) presenta un enfoque práctico que permite detectar vulnerabilidades en redes LAN jerárquicas, utilizando herramientas como NMAP para identificar equipos y evaluar adecuadamente su seguridad. Este estudio enfatiza la importancia de que los administradores de redes conozcan el comportamiento normal del tráfico y con ello detectar anomalías y posibles ataques a sus sistemas (Quishpe, 2016).

Un trabajo publicado en la Revista INSTA describe una metodología con tres fases para la detección de vulnerabilidades en redes de datos, las cuales incluyen, la evaluación de servicios y puertos activos (Franco et al., 2012). Este enfoque permite clasificar las vulnerabilidades encontradas y las soluciones adecuadas, demostrando su utilidad en la mejora de la seguridad organizacional (Franco et al., 2012).

El utilizar herramientas de software libre, se ha convertido en una práctica común en el análisis de vulnerabilidades. Sin embargo, también es importante conocer cuáles son las mejores opciones para realizar un análisis de vulnerabilidades adecuado (Castro et al., 2020). En un estudio realizado por la Universidad Nacional de Loja (Castro et al., 2020), se analizaron diversas herramientas y técnicas, destacando Qualys Guard como una de las que más se utilizan. Con este análisis se permite establecer la eficacia de las herramientas en la identificación de vulnerabilidades y aquellas aplicaciones en pruebas de penetración (Castro et al., 2020).

Las vulnerabilidades en entornos virtuales además de comprometer la infraestructura, también atentan contra la seguridad de los datos. El articulo ‘Análisis de vulnerabilidades en redes inalámbricas: métodos y soluciones’ (Mora Zambrano, 2024) menciona la necesidad de implementar protocolos de seguridad robustos y de aplicar auditorías regularmente, para mitigar riesgos por configuraciones de red incorrectas y debilidades en los dispositivos (Mora Zambrano, 2024).

El escaneo de vulnerabilidades en entornos de redes virtuales es necesario si se desea garantizar la seguridad de la infraestructura. La virtualización si bien ha revolucionado la gestión de redes al permitir la creación de múltiples entornos virtuales que interactúan entre sí dentro de un entorno físico, también introduce nuevos riesgos de seguridad que deben ser mitigados adecuadamente (Mcmahon & Patton, 2018; Yuri Baturin, 2008).

Las arquitecturas de redes virtuales suelen ser vulnerables a distintas amenazas, como ataques DoS y acceso a datos confidenciales por usuarios no autorizados. La utilización de recursos físicos entre múltiples redes virtuales puede llevar a un usuario malintencionado a acceder o interferir con el tráfico de otras redes, atentando así la confidencialidad e integridad de la información (Mora Zambrano, 2024).

En lo que concierne a las herramientas de Análisis de Vulnerabilidades para reducir estos riesgos, es primordial realizar auditorías de seguridad periódicas y utilizar herramientas de escaneo de vulnerabilidades. Herramientas como OpenVAS y Nessus se utilizan muy a menudo para identificar y clasificar vulnerabilidades en redes virtuales (Bays et al., 2015). Estas herramientas funcionan de forma similar, realizando un

escaneo a profundidad de los sistemas en cuestión en la búsqueda de vulnerabilidades conocidas y entregar recomendaciones de posibles soluciones (Jee et al., 2021).

En relación a algunas recomendaciones generales a poner en práctica cabe señalar que realizar adecuadamente un análisis periódico es esencial para mantener la seguridad de la red virtual (Heiding et al., 2023). Estos análisis permiten a los administradores encontrar vulnerabilidades antes de que los atacantes las exploten. Además, permite que los resultados del análisis, puedan documentar aquellas vulnerabilidades encontradas e implementar las medidas correctivas necesarias (Bays & Alegre, 2018).

Cuando han sido identificadas estas vulnerabilidades, es importante tener medidas correctivas. Las cuales pueden incluyen actualizaciones de software (Manjula et al., 2023), configuraciones adecuadas de seguridad más robustas e implementación de políticas de acceso seguro a algún sistema (Kumar Kande, 2023). Documentar estas vulnerabilidades y las acciones tomadas es de vital importancia para auditorias futuras, además que ayuda a mejorar la postura de seguridad de la organización (Kaur et al., 2023).

2.2. Proceso de análisis de vulnerabilidades

2.2.1. Configuración del entorno virtual de red en Linux, simulando un entorno corporativo

Para iniciar con todo el proceso, se configura una red virtualizada, por medio de máquinas virtuales, las cuales simulan un entorno corporativo. Los pasos son los siguientes:

- Creación de máquinas virtuales: Se configuraron varias máquinas virtuales en Oracle VM VirtualBox, cada una representando diferentes roles dentro de un entorno corporativo, tal cual se señaló en la topología. El entorno virtual consta de un servidor web (Apache, PostgreSQL), servidor de archivos (Samba), dos estaciones de trabajo (Ubuntu Desktop), servidor de monitoreo (Nagios), y un servidor de análisis (Kali Linux).

- Máquina virtual de generación de red virtual: Se asignó a una de las máquinas virtuales como la encargada de gestionar el entorno de la red virtual, para este caso la máquina virtual fue la nombrada como “Web Service”, la cual usa Ubuntu Server, y en la cual se configuró el manejo de las direcciones IP que se asignaran a las máquinas de estaciones de trabajo.

- Configuración de red virtual en VirtualBox: La red virtual se configuró mediante NAT y redes internas, asegurando una adecuada comunicación entre todas las máquinas, manteniendo un aislamiento seguro de la red externa.

2.2.2. Instalación y configuración de herramientas de escaneo

Se procedió a la instalación y configuración de las herramientas de escaneo de vulnerabilidades Nessus como la herramienta principal, y se han instalado otras herramientas complementarias para el análisis.

- Selección de herramienta principal: Nessus Essentials ha sido elegida para el escaneo de vulnerabilidades, gracias a su capacidad de identificar una gran cantidad de vulnerabilidades en sistemas operativos y redes.

- Instalación y configuración de Nessus y otras herramientas: Nessus se instaló como herramienta principal junto con OpenVAS y Nmap en la máquina de análisis (Kali Linux). Estas herramientas están configuradas para ejecutar escaneos en las direcciones IP de las máquinas virtuales configuradas, asegurando que se identifiquen todas las posibles vulnerabilidades.

2.2.3. Iniciación de todas las máquinas virtuales

Con las herramientas de análisis ya configuradas, se ponen en ejecución todas las máquinas que conforman la red virtual para realizar los respectivos análisis.

- Iniciar los servicios de cada servidor y estaciones de trabajo: Para el servidor web se inicializaron los servicios de Apache y la base de datos PostgreSQL junto con Samba. Luego se pusieron en ejecución las dos estaciones de trabajo que funcionan con Ubuntu Desktop. Finalmente, para el servidor de monitoreo, se inicializó Nagios.

- Identificación de IP: Luego de inicializadas cada máquina, se verifica que cada máquina posea una dirección IP adecuada en base a la configuración inicial y sobre todo que estén dentro de la misma red. Se realiza una verificación de cada máquina.

- Lista de Ip válidas: Luego de analizar cada máquina se obtuvo las siguientes direcciones IP que se encuentran en la Tabla 1, las cuales fueron las utilizadas para el escaneo de vulnerabilidades.

Tabla 1: *Lista de las direcciones IP que se han obtenido de cada máquina virtual, lo cual muestra que pertenecen a la misma red*

Máquina Virtual	Dirección Ip
Servidor de análisis (Kali Linux)	10.0.2.6
Estación de trabajo 1 (Ubuntu Desktop)	10.0.2.15
Estación de trabajo 2 (Ubuntu Desktop)	10.0.2.11
Servidor Web (Ubuntu Server)	10.0.2.10
Servidor de archivos (Ubuntu Server)	10.0.2.9
Servidor de monitoreo (Ubuntu Server)	10.0.2.8

2.2.4. Proceso de Escaneo de Vulnerabilidades

Con el escaneo de puertos se busca identificar los diferentes tipos de vulnerabilidades que van desde errores de configuración hasta fallas del software. Este proceso permitió obtener un panorama más claro de la seguridad de la red y detectar aquellos posibles puntos de entrada para ataques externos.

- Configuración de herramientas de escaneo: Se estableció a Nessus como herramienta principal para que realice un análisis de vulnerabilidades de forma avanzada. Esto obtuvo una serie de datos más certeros y específicos, donde se selecciona la opción de escáner avanzado. Tambien se ha configurado a OpenVAS y Nmap para que realicen el mismo tipo de escaneo que Nessus.

- Asignaciones de direcciones a escanear: Luego de tener la lista de direcciones IP, se fueron agregando cada una para realizar un análisis de todas las maquinas en pleno funcionamiento.

- Tiempo de escaneo: Luego de configurar todo y poner en marcha el escáner para encontrar las posibles vulnerabilidades se debe esperar a que todos los procesos culminen para tener un informe adecuado, los tiempos de escaneo para cada máquina virtual varía en función de que servicios, puertos, conexiones, etc.

- Pruebas manuales: Además del uso de herramientas automatizadas, se he implementado una técnica manual para verificar vulnerabilidades. Este proceso solo se realiza para validar los resultados obtenidos con otras herramientas. Sin embargo, este proceso puede no ser el más adecuado sobre todo por tiempo de demora en la identificación de vulnerabilidades.

III. Evaluación de Resultados y Discusión

3.1. Evaluación de Resultados Generales

Dentro del entorno de red virtual, los resultados de vulnerabilidad obtenidos con Nessus y con otras herramientas como OpenVAS y Nmap se analizaron para cada dirección IP asignada a cada máquina virtual, como se puede observar en la Tabla 2 mediante Nessus. La dirección IP de cada máquina y servidor representa un nodo específico en la red y toda vulnerabilidad descubierta se evaluó para determinar su nivel de gravedad e impacto en la seguridad de la red virtual.

Tabla 2: *Numero de vulnerabilidades totales por host y severidad encontradas con Nessus*

Host	Criticas	Altas	Medias	Bajas	Informativas
10.0.2.6	0	0	1	0	52
10.0.2.15	1	3	0	1	18
10.0.2.11	0	0	1	1	20
10.0.2.10	1	3	1	1	28
10.0.2.9	0	0	0	1	5
10.0.2.8	0	0	0	1	5
TOTAL	**2**	**6**	**3**	**5**	**128**

Es importante conocer los porcentajes respecto a cantidad de vulnerabilidades encontradas, como se puede ver en la Tabla 3.

Tabla 3: *Distribución de Vulnerabilidades por Severidad (Porcentajes) con Nessus*

Severidad	Cantidad de vulnerabilidades	Porcentaje %
Vulnerabilidades Críticas	2	1.38%
Vulnerabilidades Altas	6	4.14%
Vulnerabilidades Medias	3	2.07%
Vulnerabilidades Bajas	5	3.45%
Vulnerabilidades Informativas	128	88.28%
TOTAL	**145**	**100%**

3.1.1. Análisis comparativo con otras herramientas

El análisis revela que mientras Nessus identificó más vulnerabilidades totales, también generó un número significativo de falsos positivos. Por otro lado, OpenVAS mostró resultados comparables, pero con una menor cantidad de falsos positivos, Nmap mantuvo valores parecidos a los obtenidos con OpenVAS. Mientras que las pruebas manuales confirmaron la existencia real de las vulnerabilidades sin generar falsos positivos,

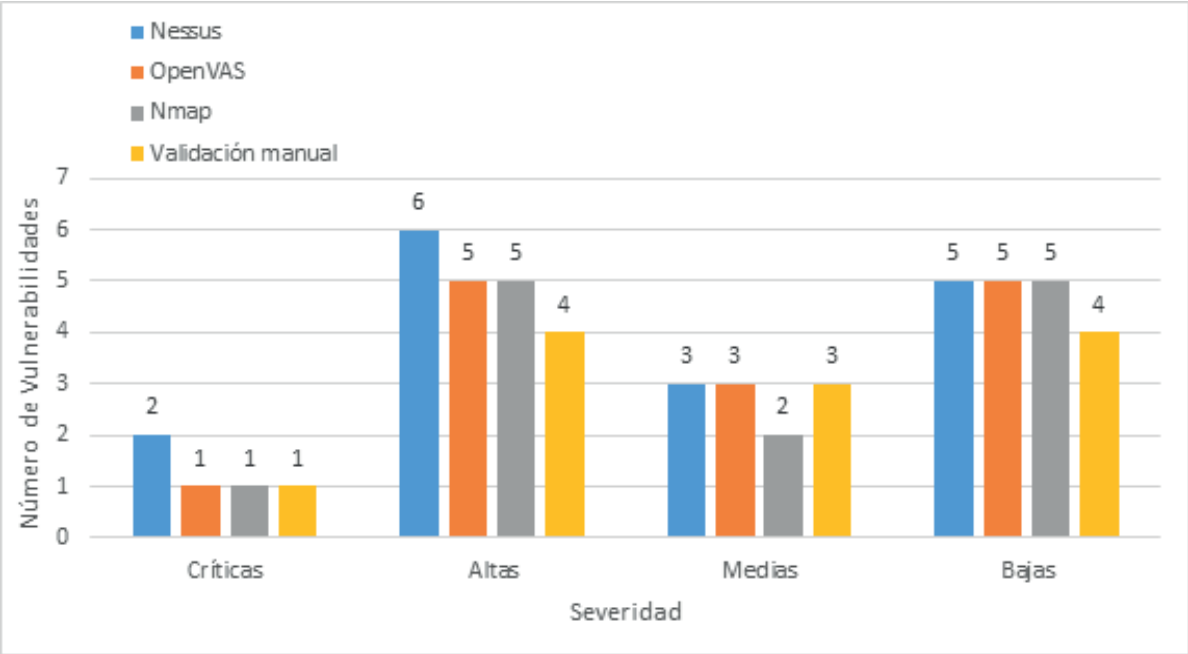
lo que subraya su importancia en el proceso de validación. En la Tabla 4 se puede apreciar esta comparativa a detalle.

Tabla 4: *Comparativa de vulnerabilidades de Nessus con otras herramientas*

Herramienta	Total Vulnerab.	Críticas	Altas	Medias	Bajas	Falsos Positivos
Nessus	16	2	6	3	5	3
OpenVAS	14	1	5	3	5	2
Nmap	13	1	5	2	5	2
Validación manual	12	1	4	3	4	0

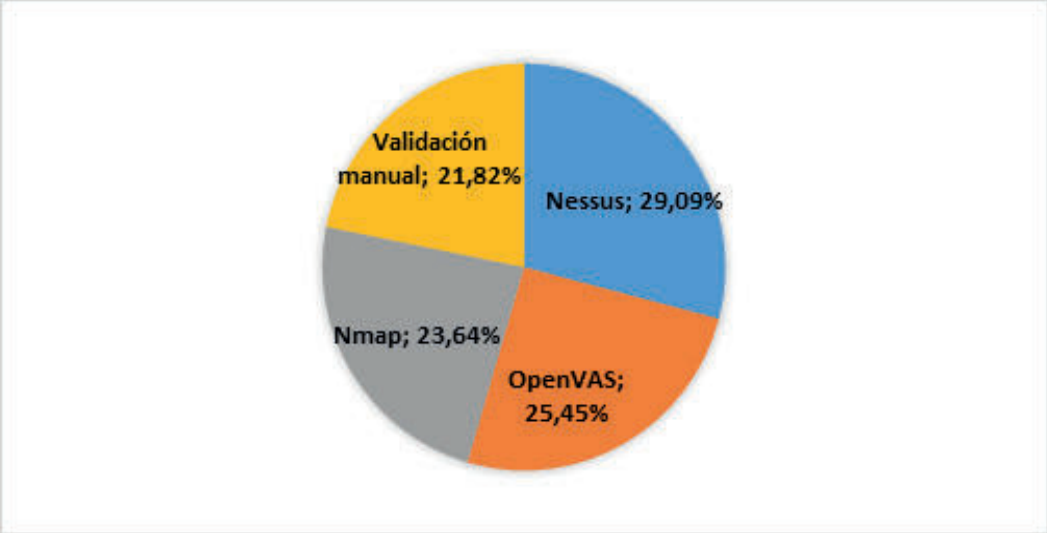
En la Figura 2 se presentan las vulnerabilidades críticas, altas, medias y bajas de mayor impacto. La diferencia entre las herramientas y el proceso manual resalta la importancia de la validación para eliminar falsos positivos.

Figura 2: *Comparación de Vulnerabilidades Detectadas por Herramienta*



En la Figura 3 se muestra el predominio de las vulnerabilidades en base a cada herramienta utilizada junto a la validación manual. Esta ilustración permite identificar rápidamente el porcentaje de la herramienta o técnica que mayor número de vulnerabilidades obtuvo.

Figura 3: *Distribución total de vulnerabilidades por herramienta*



3.1.2. Utilización de la Métrica CVSS

Se utilizó el Sistema de Puntuación de Vulnerabilidad Común (CVSS) como punto de referencia para medir la gravedad de las vulnerabilidades encontradas. La métrica CVSS consta de varios factores que evalúan la facilidad de explotar la vulnerabilidad y el impacto potencial de explotarla:

- Base Score: indica la gravedad general de la vulnerabilidad (en una escala de 0 a 10).
- Temporal Score: refleja la gravedad de la vulnerabilidad en un contexto temporal específico, teniendo en cuenta factores como la disponibilidad de parches.
- Environmental Score: ajusta la puntuación en función del impacto en el entorno específico en el que existe la vulnerabilidad.

En el análisis, se priorizaron las vulnerabilidades críticas y altas (con puntuaciones CVSS superiores a 7) debido a su alto riesgo de explotación y posible impacto en la red. En la Tabla 5, se puede observar los valores obtenidos.

Tabla 5: *Vulnerabilidades Críticas Identificadas según Nessus*

Host	Plugin Nessus	Descripción	CVSS 3.0
10.0.2.8	Apache 2.4.x < 2.4.60 Multiple Vulnerabilities	Vulnerabilidades múltiples en Apache HTTP Server	9.8
10.0.2.10	Apache 2.4.x < 2.4.60 Multiple Vulnerabilities	Vulnerabilidades múltiples en Apache HTTP Server	9.8

3.1.3. Documentación de Vulnerabilidades Críticas y Altas

Una vez identificadas y evaluadas las vulnerabilidades, se documentaron las críticas y de mayor riesgo, con especial atención a aquellas que podrían poner en peligro la seguridad de todo el entorno:

- ***Apache 2.4.x < 2.4.60 Multiple Vulnerabilities (CVSS 9.8, Crítica)***
    - Impacto: esta vulnerabilidad podría permitir la ejecución remota de código, lo que comprometería gravemente la integridad del servidor afectado y potencialmente daría acceso completo al atacante.
- ***SSL Certificate Cannot Be Trusted (CVSS 6.5, Mediana)***
    - Impacto: con esta vulnerabilidad se expone la red a posibles ataques de intermediarios, en la cual un atacante podría interceptar y manipular datos confidenciales.

3.1.4. Medidas de mitigación propuestas

Estas medidas se han desarrollado teniendo en cuenta las características específicas de las vulnerabilidades detectadas y que además son vulnerabilidades comunes en estos sistemas en redes virtuales.

- **Vulnerabilidades Críticas Relacionadas con Apache**

Se pudo identificar una vulnerabilidad crítica con el servidor web Apache, esto debido a versiones anteriores. Con este tipo de vulnerabilidades se puede realizar la ejecución remota de código (RCE) y otras fallas de seguridad que los ciber atacantes podrían aprovechar para comprometer todo el sistema.

Posibles medidas correctivas:

- Actualización a Versiones más recientes de Apache
- Revisión y Fortalecimiento de la Configuración de Apache

- **Vulnerabilidades Relacionadas con SSL/TLS**

Se descubrió un problema con la configuración SSL/TLS, incluido el uso de certificados los cuales no son de confianza y sobre todo la falta de un cifrado seguro. Con esta vulnerabilidad se expone toda la red a ataques de hombre en el medio (MITM) que comprometen la confidencialidad de todas las comunicaciones de esta red.

Posibles medidas correctivas:

- Implementación adecuada de Certificados SSL/TLS Confiables
- Configuración oportuna y correcta de Cifrados SSL/TLS

3.2. Discusión

Los resultados obtenidos destacan que, a pesar de estar en un entorno virtual controlado, las vulnerabilidades en la configuración de servicios comunes como Apache y SSL pueden representar riesgos importantes. En particular, las vulnerabilidades críticas identificadas tienen el potencial de comprometer toda la red si no

se corrigen. Esto resalta la importancia de ejecutar análisis de seguridad exhaustivos y regulares. Exponer estas vulnerabilidades en un entorno virtual permite a los administradores de sistemas y equipos de seguridad detectar y solucionar problemas antes de que afecten a un entorno de producción más amplio, potencialmente más expuesto a amenazas del mundo real.

En lo concerniente a los trabajos relacionados, los investigadores presentan el predominio de configuraciones obsoletas y vulnerables en servidores Apache y certificados SSL en diversos entornos virtuales y reales. Las configuraciones caducas de Apache y SSL se han identificado sistemáticamente como vulnerabilidades comunes en muchas redes, lo que refleja una tendencia generalizada en la que los servicios no se actualizan adecuadamente. La simulación en un entorno virtual permitió la identificación de estas vulnerabilidades en un contexto controlado, lo que proporciona una gran ventaja que permite manejar estas vulnerabilidades antes de desplegar la infraestructura en un entorno de producción.

En relación a las limitaciones de este estudio, la exploración se realizó en una red virtual de pequeña escala, lo cual limita la capacidad de generalizar los resultados a redes más robustas y con un mayor número de usuarios en red. La pequeña escala de la red y el número limitado de máquinas virtuales no reproducen de forma óptima el tráfico real de una red corporativa de gran tamaño y de alta demanda de trafico de usuarios, la complejidad y las condiciones de una red empresarial de producción. Esto podría afectar la generalidad de estos resultados a entornos más grandes y diversos, en la cual las interacciones entre diferentes componentes de la red podrían mostrar nuevas o distintas vulnerabilidades, lo cual podría complicar aquellos problemas ya existentes.

En relación a las implicaciones de esta práctica, se evidencia que el ejecutar análisis periódicos en entornos virtuales es de vital importancia para identificar y solucionar aquellas vulnerabilidades antes de implementarlos en entornos de producción de mayor escala. La detección de vulnerabilidades temprana y la corrección pro activa podrían llegar a evitar que se aprovechen las vulnerabilidades de los servicios y protocolos existentes. Además, es importante que se cumplan las políticas de actualización y de configuraciones segura, para reducir la incidencia de vulnerabilidades comunes. La correcta utilización de prácticas rigurosas de mantenimiento y monitoreo en redes virtuales puede mejorar adecuadamente la seguridad y garantizar que las redes se mantengan protegidas de amenazas antes de que se conviertan en problemas en producción.

Finalmente, el análisis de vulnerabilidades realizado evidenció la generación de posibles falsos positivos al utilizar herramientas automatizadas como Nessus, OpenVAS y Nmap. Para comprender esta problemática se incluyeron herramientas adicionales y técnicas manuales, lo cual permitió validar las detecciones y aumentar la confiabilidad de los resultados obtenidos. La utilización de herramientas complementarias como OpenVAS y Nmap, evidenciaron que es posible corroborar y reducir los falsos positivos que se generaron con Nessus que fueron los que más se reportaron. A pesar que el uso de herramientas adicionales enriqueció el análisis, también incremento el tiempo requerido para completar la validación, y que, al implementar en redes más robustas, podría volverse menos eficiente, lo cual sugiere una mejor optimización con sistemas automatizados.

IV. Conclusiones y Trabajo futuro

En el análisis realizado se identificaron dos tipos principales de vulnerabilidades en esta red virtual: aquellas versiones desactualizadas de Apache y configuraciones SSL/TLS inseguras. Aquellas versiones obsoletas de Apache, como las anteriores a la 2.4.60, poseen vulnerabilidades críticas que pueden ser explotadas para la

ejecución remota de código y con ello comprometer la seguridad del servidor. Por otro lado, las inadecuadas configuraciones SSL/TLS que son inseguras podrían permitir ataques de intermediario, poniendo en grave riesgo la confidencialidad de datos y la integridad en las comunicaciones que se supone están cifradas. Una acción correctiva incluye actualizar urgentemente los Apache a la última versión y configurar adecuadamente los certificados SSL/TLS para así certificar que sean confiables y que se utilicen protocolos completamente seguros. Además, cabe recomendar la implementación de políticas de seguridad de red más estrictas y así protegerse contra la divulgación accidental de información confidencial.

En este estudio se destaca la importancia de implementar análisis de vulnerabilidades en entornos virtuales. Esta exploración presenta una hoja de ruta adecuada para la detección y disminución de vulnerabilidades de seguridad en una red virtual controlada. El estudio además proporciona información valiosa que se pueda aplicar a entornos de producción más amplios y robustos. Tener la capacidad de poder detectar vulnerabilidades en un entorno controlado antes de que se implemente en producción es importante para evitar compromisos con la seguridad y proteger la infraestructura crítica de la empresa.

El uso de Nessus como herramienta de análisis de vulnerabilidades puede generar falsos positivos, pero su combinación de cobertura amplia, su facilidad de utilización y el soporte que brinda, lo han convertido en una herramienta clave y relevante para el análisis de vulnerabilidad de manera automatizada. En este este estudio las otras herramientas utilizadas no buscan sustituir a Nessus, sino por el contrario fortalecer el análisis al proveer enfoques complementarios especialmente en la validación de falsos positivos y detención granular.

Como trabajo futuro se planea integrar técnicas de inteligencia artificial en el análisis de escaneos de puertos vinculándolo con el marco de trabajo Metasploit de Kali Linux

Referencias

Bays, L. R., & Alegre, P. (2018). UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL INSTITUTO DE INFORMÁTICA PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO Virtual Network Embedding in Software-Defined Networks.

Bays, L. R., Oliveira, R. R., Barcellos, M. P., Gaspary, L. P., & Mauro Madeira, E. R. (2015). Virtual network security: threats, countermeasures, and challenges. *Journal of Internet Services and Applications*, 6(1). https://doi.org/10.1186/s13174-014-0015-z

Castro, J., Castro, M., Mercedes Ortiz Hernández, M., Antonio, E., & Lino, M. (2020). UNESUM-Ciencias: *Revista Científica Multidisciplinaria. Publicación Cuatrimestral*, 5(1).

Chatterjee, S., & Thekdi, S. (2020). An iterative learning and inference approach to managing dynamic cyber vulnerabilities of complex systems. *Reliability Engineering & System Safety*, 193, 106664. https://doi.org/10.1016/J.RESS.2019.106664

Franco, D. A., Perea, J. L., & Puello, P. (2012). Metodología para la Detección de Vulnerabilidades en Redes de Datos. *Información Tecnológica,* 23(3), 113–120. https://doi.org/10.4067/S0718-07642012000300014

Heiding, F., Katsikeas, S., & Lagerström, R. (2023). Research communities in cyber security vulnerability assessments: A comprehensive literature review. *Computer Science Review,* 48, 100551. https://doi.org/10.1016/J.COSREV.2023.100551

Jee, S. H., Park, J. S., & Shon, J. G. (2021). Security in Network Virtualization: A Survey. *Journal of Information Processing Systems*, 17(4), 801–817. https://doi.org/10.3745/JIPS.04.0220

Jeon, S., & Kim, H. K. (2021). AutoVAS: An automated vulnerability analysis system with a deep learning approach. *Computers & Security*, 106, 102308. https://doi.org/10.1016/J.COSE.2021.102308

Kaur, R., Gabrijelčič, D., & Klobučar, T. (2023). Artificial intelligence for cybersecurity: Literature review and future research directions. *Information Fusion*, 97, 101804. https://doi.org/10.1016/J.INFFUS.2023.101804

Kumar Kande, S. (2023). Vulnerability Management Best Practices: Developing Strategies for Tracking and Remediating Vulnerabilities within Defined SLAs. http://creativecommons.org/licenses/by/3.0/,whichper-mitsunrestricteduse,providedtheoriginalauthorandsourcearecredited.

Lallie, H. S., Debattista, K., & Bal, J. (2020). A review of attack graph and attack tree visual syntax in cyber security. *Computer Science Review*, 35, 100219. https://doi.org/10.1016/J.COSREV.2019.100219

Manjula, M., Venkatesh, & Venugopal, K. R. (2023). Cyber Security Threats and Countermeasures using Machine and Deep Learning Approaches: A Survey. *Journal of Computer Science*, 19(1), 20–56. https://doi.org/10.3844/jcssp.2023.20.56

McGhin, T., Choo, K. K. R., Liu, C. Z., & He, D. (2019). Blockchain in healthcare applications: Research challenges and opportunities. *Journal of Network and Computer Applications*, 135, 62–75. https://doi.org/10.1016/J.JNCA.2019.02.027

Mcmahon, E., & Patton, M. (2018). *Benchmarking Vulnerability Assessment Tools for Enhanced Cyber-Physical System* (CPS) Resiliency.

Mora Zambrano, E. R. (2024). *Análisis de vulnerabilidades en redes inalámbricas: métodos y soluciones*. Instituto Superior Universitario Japón.

Moreno, M. (16 de Octubre de 2013). Security At Work. Obtenido de Security At Work: https://www.securityartwork.es/2013/10/16/nessus-report-paranoia/

Quishpe, H. D. (2016). Análisis de Vulnerabilidades en la Red LAN Jerárquica de la Universidad Nacional de Loja, en el Área de la Energía, Industrias y los Recursos Naturales No Renovables.

Russo, E. R., Di Sorbo, A., Visaggio, C. A., & Canfora, G. (2019). Summarizing vulnerabilities' descriptions to support experts during vulnerability assessment activities. *Journal of Systems and Software*, 156, 84–99. https://doi.org/10.1016/J.JSS.2019.06.001

Serrano, J. R. (2024). Trabajo Fin de Grado Análisis de Redes y Vulnerabilidades.

Yuri Baturin. (2008). *Situational Analysis with Analytical Support in Virtual Environment for Decision Making Process Under High-Risk and Crisis Conditions*.