Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

ANALISIS PENGUJIAN KEAMANAN *FIREWALL* PADA SISTEM X DI UNIVERSITAS Z

Benedictus Rafael Lesmana[1]* Achmad Junaidi[2]
Andreas Nugroho Sihananto[3]

Program Studi Informatika[1,2,3]
Fakultas Ilmu Komputer[1,2,3]
Universitas Pembangunan Nasional "Veteran" Jawa Timur[1,2,3]

20081010091@student.upnjatim.ac.id[1],
achmadjunaidi.if@upnjatim.ac.id[2],
andreas.nugroho.jarkom@upnjatim.ac.id[3]

Received: June 18, 2024**. Revised:** July 23, 2024**. Accepted:** July 24, 2024**. Issue Period:** Vol.8 No.3 (2024), Pp.557-571

Abstrak: Penggunaan sistem X di ruang lingkup kampus Z semakin sering digunakan baik oleh mahasiswa maupun tenaga pendidik di sekitar kampus. Dengan terkoneksinya sistem ke jaringan komputer dan internet, maka peluang berubah atau rusaknya data akan semakin terbuka lebar, karena *user* dari sistem X yang berpotensi berbahaya (*malicious user*) akan mudah masuk ke sistem melalui jaringan komputer atau internet. *Firewall* adalah alat keamanan jaringan yang mengawasi lalu lintas (*traffic*) yang masuk dan keluar dari jaringan dan menentukan apakah paket data boleh diterima atau diblokir menggunakan aturan khusus. Pengujian keamanan firewall perlu dilakukan untuk melihat seberapa rentan *firewall* yang dimiliki oleh sistem X. Dengan menggunakan Kali Linux untuk melakukan *penetration testing* dan *nessus* sebagai alat untuk memindai kerentanan, maka didapatkan seberapa kompleks proses *penetration testing* suatu *firewall* pada sistem X serta hasil kerentanan yang rinci dari pemindaian *nessus*. Hasil yang didapatkan setelah melakukan pemindaian kerentanan adalah didapatkan beberapa kerentanan yang dimiliki dari sistem X serta beberapa informasi yang perlu diperhatikan untuk menjaga keamanan jaringan. Sistem X memiliki tiga kerentanan tingkat sedang satu kerentanan tingkat rendah serta tiga puluh lima informasi keamanan yang perlu diperhatikan serta solusi yang ditawarkan untuk mengatasi kerentanan yang dimiliki sistem X. Dari pengujian yang dilakukan, dapat disimpulkan sistem X memiliki keamanan yang cukup baik dengan beberapa kerentanan yang perlu diperhatikan.

Kata kunci: *Firewall*, *Kali Linux*, Keamanan Jaringan, *Nessus*, *Penetration Testing*

Abstract: *The use of system X in the scope of campus Z is increasingly being used by both students and educators around the campus. With the connection of the system to computer networks and the internet, the opportunity to change or damage data will be wide open, because users of the X system who are potentially dangerous (malicious users) will easily enter the system via computer networks or the internet. A firewall is a network security tool that monitors traffic entering and leaving the network and determines whether data packets are acceptable or blocked using special rules. Firewall security testing needs to be done to see how vulnerable the firewall owned by system X is. By using Kali Linux to perform penetration testing*

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

and nessus as a tool for scanning vulnerabilities, it is obtained how complex the penetration testing process of a firewall on system X is and the detailed vulnerability results of the nessus scan. The results obtained after performing a vulnerability scan are several vulnerabilities owned by system X and some information that needs to be considered to maintain network security. System X has three medium-level vulnerabilities, one low-level vulnerability and thirty-five security information that needs to be considered as well as solutions offered to overcome the vulnerabilities of System X. From the tests conducted, it can be concluded that system Xs has good enough security with some vulnerabilities that need attention.

Keywords: *Firewall, Kali Linux, Network Security, Nessus, Penetration Testing*

I. PENDAHULUAN

Universitas Z merupakan salah satu universitas yang berlokasi di Surabaya, Jawa Timur. Universitas Z telah lama menggunakan sistem X dan semakin sering digunakan khususnya pada saat pandemi COVID-19 masih marak di Surabaya. Pada tahun 2022, beberapa situs yang dimiliki oleh Universitas Z diretas oleh sekelompok peretas. Peretasan ini dilakukan dengan melakukan perubahan wajah (*deface*) situs yang ada pada beberapa situs yang dimiliki Universitas Z. Meski tidak ada berita berapa data yang berhasil dicuri atau diambil pada peristiwa ini, namun hal tersebut menandakan bahwa keamanan yang dimiliki Universitas Z masih perlu ditingkatkan lagi. Penelitian yang dilakukan oleh Fernanda Tinambunan pada sistem X dengan menggunakan metode OWASP TOP 10 mengungkapkan bahwa pada sistem X memiliki beberapa celah keamanan dengan satu celah keamanan tingkat tinggi yang perlu diperhatikan.

Keamanan jaringan atau sistem informasi terdiri dari seperangkat kebijakan dan pelaksanaan yang diterapkan untuk mencegah dan memantau akses tidak sah, modifikasi dalam sistem, penyalahgunaan, atau penolakan jaringan komputer dan sumber daya yang dapat diakses jaringan. Implementasi teknologi keamanan sebagai tindakan perlindungan menjadi pilihan dalam upaya melindungi aset informasi dari ancaman atau serangan teknologi keamanan hadir sebagai perlindungan keamanan atas ancaman atau serangan pada jaringan atau sistem informasi antara lain *firewall, cryptographic system, IDS, SSL, antivirus system, IPSec, authentication* dan lain sebagainya. [5]

Firewall adalah alat keamanan jaringan yang mengawasi lalu lintas (*traffic*) yang masuk dan keluar dari jaringan dan menentukan apakah paket data boleh diterima atau diblokir menggunakan aturan khusus. Dengan *firewall filtering*, konten atau paket yang ilegal, tidak pantas atau tidak sah, aksesnya akan diblok dan disaring dengan cara menyaring paket tersebut. Dengan membuat aturan yang baik, *firewall* akan lebih mudah memfilter lalu *traffic* dan *bandwith* jaringan, mengatasi masalah penyebaran *malware* dalam jaringan yang menyebabkan jaringan lambat. Salah satu dampak *malware* pada jaringan adalah kelebihan *bandwith* yang dapat dengan cepat menguras atau membuat data masuk dan keluar menjadi lebih lambat dari biasanya. [11]

Salah satu cara yang dapat digunakan untuk menguji keamanan jaringan adalah dengan percobaan penetrasi atau *penetration testing*. Percobaan penetrasi atau *penetration testing* adalah sebuah simulasi dari sebuah serangan untuk memeriksa keamanan suatu sistem atau *environment* untuk dianalisis. Lebih lanjut, percobaan penetrasi ini tidak disamakan dengan pemindaian *port*. Misalnya jika pemindaian *port* seperti melihat titik masuk rumah melalui sebuah teropong, maka percobaan penetrasi seperti seseorang yang ingin memaksa masuk ke dalam rumah [6]. Dari segi operasional, *penetration testing* membantu membentuk strategi keamanan informasi melalui identifikasi kerentanan yang cepat dan akurat. *Penetration testing* membagikan informasi rinci tentang ancaman keamanan secara aktual, yang dapat dieksploitasi jika tercakup dalam aliran dan proses keamanan organisasi. Hal ini akan membantu organisasi untuk mengidentifikasi dengan cepat dan akurat potensi kerentanan yang nyata [7].

Penelitian sebelumnya yang dilakukan oleh [21] menguji untuk mengatasi permasalahan yang sering ditemui pada *opensource firewall*. Pada penelitian ini, dilakukan pendekatan dengan menggunakan layanan *Nessus* dan *Common Vulneralbility Scoring* System (CVSS) atau sistem penilaian kerentanan sistem untuk melihat risiko yang berhubungan dengan kerentanan data. Kemudian, ditampilkan beberapa solusi di antaranya,

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

menggunakan alat untuk menangkap paket yang masuk seperti *Wireshark* dimana hal ini efektif, namun terdapat beberapa alat pendeteksi lain yang menggunakan teknik pendeteksian berbeda yang dapat menembus sistem. Kemudian yang kedua dengan menggunakan konfigurasi khusus untuk *firewall* yang dibangun dengan menggunakan *Nmap* yang digunakan untuk mengidentifikasi *router* dari jaringan dan *port* yang terbuka dan menggunakan *pfSense* yang sudah ditingkatkan yang dapat menyediakan beberapa pertahanan untuk beberapa tipe permasalahan yang ada pada jaringan lokal dimana hal ini menunjukkan bahwa hal ini efisien. Dari hasil yang dilakukan kemudian ditemukan bahwa keamanan jaringan pada *opensource firewall* sangatlah sulit untuk mendapatkan keamanan jaringan yang penuh. Pengujian pada keamanan jaringan yang dilakukan secara manual dan secara otomatis perlu diadakan secara bersamaan untuk mendapatkan efisiensi yang tinggi dalam pengujian. Pada penelitian ini, disarankan untuk menggunakan aplikasi yang berbayar untuk melakukan pengujian, akan tetapi hal ini membutuhkan biaya yang sangat mahal serta pengetahuan yang tinggi untuk mengetahui penggunaannya. Penulis berharap pada penelitian selanjutnya untuk menggunakan alat *opensource* lainnya untuk mencari kerentanan data.

Dengan penjelasan yang telah dijabarkan, maka penulis melakukan analisis untuk menguji keamanan *firewall* yang ada pada sistem X di Universitas Z dengan harapan setelah dilakukan pengujian dan ditemukannya kerentanan yang dimiliki, pihak universitas dapat memperbaiki keamanan jaringan *firewall* sistem X yang dimiliki serta melakukan pengujian secara rutin untuk menjaga keamanan jaringan sistem-sistem yang dimiliki agar data-data penting yang dimiliki dapat semakin terjaga keamanannya.

II. METODE DAN MATERI

2.1. Metode

Metode penelitian yang dilakukan dengan beberapa tahapan yang digunakan untuk menunjang penelitian untuk mencapai hasil yang diharapkan dimana tahapan-tahapan ini diilustrasikan oleh Gambar 1.

Gambar 1. Alur penelitian

2.1.1. Pengumpulan Data

Tahapan pengumpulan data dilakukan dengan cara melakukan beberapa pemindaian untuk mendapatkan informasi yang berhubungan dengan *firewall* sistem X. Pemindaian dilakukan dengan menggunakan aplikasi *Nmap* (*Network Mapper*) yang merupakan aplikasi *opensource* dan biasa digunakan untuk melakukan pengujian keamanan. Beberapa pemindaian yang dilakukan untuk mengumpulkan data diantaranya:

1. Pencarian IP *address*

   Tahapan pencarian IP address dilakukan sebagai langkah awal untuk mengetahui IP *address* yang digunakan pada sistem X,
2. *Tracerouting*

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Tahapan *tracerouting* dilakukan untuk memberikan informasi mengenai rute yang diambil paket di antara sistem dan menentukan semua *router* yang melibatkan proses pembentukan koneksi. Hasil yang diberikan dari tahapan ini adalah urutan beberapa alamat IP mengenai domain halaman yang akan diuji.

3. Mengirim *Ping*
   Tahapan *Ping* dilakukan untuk mengetahui apakah domain yang ingin diuji aktif atau tidak. Bila sistem mengembalikan paket yang dikirimkan dengan menggunakan *ping*, maka dapat diketahui bahwa sistem yang ingin diuji sedang aktif begitupun sebaliknya bila sistem tidak mengembalikan paket, maka sistem sedang tidak aktif.
4. *Fingerprinting* OS
   Tahapan *Fingerprint* OS dilakukan untuk mencari tahu sistem operasi yang digunakan oleh target mesin sistem.
5. Pemindaian *port*
   Pemindaian *port* dilakukan untuk mengetahui apakah terdapat *port* yang terbuka atau tidak dalam suatu *firewall*. Selain itu, pemindaian *port* juga dilakukan untuk mengetahui layanan apa yang diberikan pada *port* yang terbuka.
6. Pemindaian versi
   Pemindaian versi dilakukan untuk mengetahui versi yang digunakan oleh masing-masing *port* pada tiap domain.
7. Pemindaian agresif
   Pada tahapan ini, dilakukan beberapa pemindaian secara bersamaan, yakni pemindaian versi, pemindaian sistem operasi, *scripting*, dan *tracerouting*. Meski sudah dilakukan beberapa pemindaian yang sama sebelumnya, akan tetapi tidak jarang pemindaian memiliki hasil yang berbeda antara melakukan pemindaian secara satu-persatu dengan melakukan pemindaian secara agresif sehingga dengan melakukan pemindaian agresif, didapatkan informasi tambahan yang dapat menunjang pengujian. *Scripting* merupakan fitur untuk membuat skrip atau naskah untuk menjelaskan informasi tambahan yang tidak dapat diberikan dengan menggunakan perintah sederhana *Nmap*. *Scripting* menggunakan *Nmap Scripting Engine* (NSE) dimana NSE sudah ada dalam paket instalasi *Nmap*.

2.1.2. Pemindaian Kerentanan

Pemindaian kerentanan dilakukan dengan menggunakan aplikasi *nessus*. Dengan menggunakan aplikasi *nessus*, dilakukan pemindaian dengan memasukkan IP *address* yang sudah didapatkan pada tahap sebelumnya. Setelah itu, dapat juga dilakukan beberapa opsi pengujian lainnya seperti pengaturan jadwal pemindaian yang ingin dilakukan, pengaturan pencarian dilakukan untuk sebagian atau keseluruhan *port* sistem, dan pengaturan penilaian pengujian kerentanan yang ada pada sistem secara sebagian atau keseluruhan serta secara singkat atau secara kompleks. Setelah melakukan pengaturan pemindaian yang diinginkan, *nessus* akan melakukan pemindaian selama beberapa waktu lalu akan memberikan hasil pemindaian yang sudah dilakukan.

2.1.3. Laporan

Setelah dilakukan pemindaian pada tahap sebelumnya, maka didapatkan hasil berupa kerentanan-kerentanan yang ada pada *firewall* sistem beserta penjelasan kerentanan yang ada serta dampak yang dihasilkan bila kerentanan ini diabaikan. Selain itu juga ditawarkan solusi yang dapat diterapkan untuk mengatasi kerentanan yang ada pada *firewall* sistem. Dengan hasil yang diberikan, maka dapat dibuat suatu laporan hasil pengujian yang telah dilakukan untuk dilaporkan kepada pihak universitas.

2.2. Materi

2.2.1 *Firewall*

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Firewall merupakan mekanisme yang diimplementasikan pada perangkat keras atau lunak dengan tujuan utama melindungi jaringan. Fungsinya adalah untuk membatasi, menyaring, atau menolak lalu lintas dari segmen jaringan lokal ke jaringan eksternal di luar batas yang ditetapkan. [16] menguraikan bahwa *firewall* bertujuan untuk menjamin tidak ada akses yang melampaui atau dimasukkan tanpa izin sesuai dengan batasan keamanan yang telah ditentukan.

2.2.2 *Penetration Testing*

Penetration testing adalah metode evaluasi keamanan yang sangat strategis untuk bisnis dan operasional. Secara operasional, metode ini mendukung perancangan strategi keamanan informasi dengan mengidentifikasi kerentanan yang ada secara efisien dan tepat. [7] menyatakan bahwa temuan dari *penetration testing* menyediakan detail tentang "ancaman keamanan aktual dan potensial yang bisa dieksploitasi akibat adanya celah dalam prosedur dan proses keamanan suatu organisasi." Oleh karena itu, penetration testing memfasilitasi organisasi untuk mengenali dan mengatasi kerentanan dengan segera dan tepat.

2.2.3 *Kali Linux*

Kali Linux merupakan sebuah sistem *Linux* yang dirancang dengan fokus dalam tugas melakukan penetration testing. *Kali Linux* sebelumnya dikenal dengan nama *BackTrack* yang merupakan hasil gabungan dari tiga sistem penetration testing Linux yang berbeda, yakni *IWHAX*, *WHOPPIX*, dan *Auditor*. *Kali Linux* memiliki beberapa alat yang dapat digunakan pada saat proses penetration testing dilakukan. Beberapa alat yang dapat terdapat dalam *Kali Linux* dapat dikategorikan menjadi beberapa kategori, diantaranya: [2]

a. "*Information gathering*", Alat-alat dalam kategori ini bertujuan untuk mengumpulkan data terkait jaringan, sistem, atau aplikasi target dalam pengujian penetrasi.
b. "*Vulnerability assessment*", Alat-alat ini berfungsi untuk mengevaluasi dan mengidentifikasi potensi kerentanan dalam sistem atau aplikasi yang diuji.
c. "*Web applications*", Kategori ini termasuk alat-alat yang dirancang untuk menilai keamanan aplikasi web, meliputi pengujian kerentanan dan penetrasi.
d. "*Password attacks*" , Alat-alat ini digunakan untuk menyerang kata sandi menggunakan teknik *brute force*, *dictionary attack*, atau metode lainnya.
e. "*Exploitation tools*", Alat-alat ini dimanfaatkan untuk mengeksploitasi kerentanan yang teridentifikasi dalam sistem atau aplikasi guna memperoleh akses tidak sah.
f. "*Sniffing and spoofing*", Kategori ini meliputi alat-alat yang digunakan untuk *sniffing* (*monitoring* trafik data) dan *spoofing* (pemalsuan identitas atau alamat IP) pada jaringan.
g. "*Maintaining access*", Alat-alat ini membantu dalam mempertahankan akses yang telah diperoleh melalui penetrasi sistem yang sukses.
h. "*Reporting tools*", Alat-alat ini digunakan untuk membuat laporan tentang hasil pengujian penetrasi, termasuk rincian kerentanan yang ditemukan dan rekomendasi perbaikan.
i. "*Sytem services*", Kategori ini mencakup alat-alat yang berkaitan dengan manajemen dan konfigurasi layanan sistem yang mendukung uji penetrasi.

2.2.4 *Nmap* (*Network Mapper*)

Nmap, atau "*Network Mapper*", adalah alat *open source* untuk eksplorasi dan audit keamanan jaringan. Alat ini memanfaatkan paket IP *raw* untuk mendeteksi *host* yang terhubung ke jaringan, serta memberikan informasi tentang layanan aktif, termasuk nama dan versi aplikasi, sistem operasi, jenis *firewall* atau filter paket, dan karakteristik lainnya. Hasil dari *Nmap* ditampilkan dalam daftar *host* yang diperiksa, lengkap dengan informasi tambahan berdasarkan opsi yang dipilih. Informasi penting yang disediakan oleh *Nmap* adalah "tabel *port* menarik", yang berisi daftar *port* dengan protokolnya, nama layanan, dan statusnya. Status *port* bisa terbuka, difilter, tertutup, atau tidak difilter. Status terbuka menandakan adanya aplikasi di mesin target yang mendengarkan koneksi atau paket pada *port* tersebut.
"*Port scanning* adalah fungsi utama *Nmap*, yang merupakan proses penyelidikan otomatis untuk mengidentifikasi *port* yang aktif atau terbuka pada jaringan target," seperti yang dijelaskan oleh [12].

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

2.2.5 *Traceroute*

Traceroute atau *Tracert* merupakan perintah yang digunakan untuk menampilkan rute yang ditempuh oleh paket data menuju destinasi. Proses ini dijalankan dengan mengirimkan pesan ICMP *Echo Request* ke tujuan dengan nilai *Time to Live* yang bertambah. Rute yang terlihat adalah daftar antarmuka *router* yang paling dekat dengan host, yang berada di jalur antara *host* dan tujuan. Melalui traceroute, maka dapat menganalisis informasi tentang lokasi router, jenis dan kapasitas antarmuka, fungsi *router*, serta batasan jaringan yang dilewati, berdasarkan DNS yang terlibat. [17]

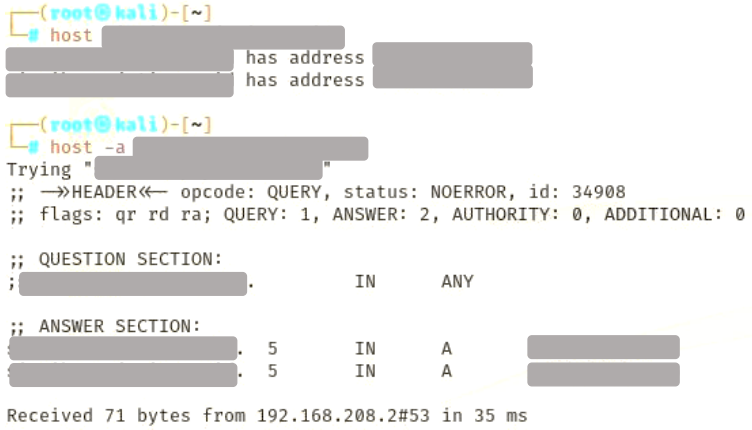
2.2.6 *Nessus*

Nessus adalah sebuah alat pemindaian keamanan jarak jauh yang digunakan secara otomatis untuk melakukan pengujian pada keamanan, dalam hal ini untuk mencari kerentanan yang dapat digunakan penyerang untuk mendapat akses kepada host yang terhubung dengan internet. *Nessus* melakukan pemindaian berdasarkan perangkat lunak kebijakan keamanan yang kita aktifkan sebelum pemindaian. Kebijakan keamanan sendiri merupakan kumpulan peraturan yang mendefinisikan hal yang diperbolehkan dan hal yang dilarang untuk digunakan atau dimanfaatkan dari sebuah akses kepada sistem saat operasi normal sedang dijalankan. Misalnya, *Nessus* dapat mengetahui berapa jumlah *port* yang terbuka pada sebuah komputer yang terkoneksi pada jaringan seperti internet. Dengan mengetahui *port* mana yang terbuka, kita dapat mencari tahu kemungkinan dari kerusakan atau mengetahui rute mana yang memungkinkan untuk mengakses komputer kita [3].

III. PEMBAHASA DAN HASIL

3.1 Pengumpulan Data

Pada tahap awal ini dilakukan pencarian IP *address* yang digunakan oleh sistem X. Pencarian IP *address* dilakukan dengan menggunakan *host* sebagai aplikasi untuk mencari IP *address* serta memasukkan DNS (*Domain Name System*) sistem X.

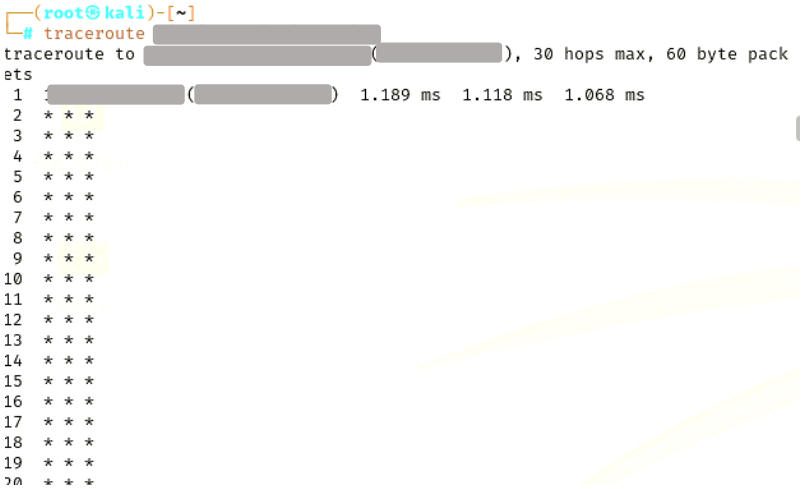


Gambar 2. Pencarian IP *Address* Menggunakan *Host*

Gambar 2 menunjukkan hasil yang didapat setelah melakukan pencarian IP *address* dengan menggunakan *host*. Dapat diketahui bahwa sistem X menggunakan dua IP *address* yang berbeda. Setelah mendapatkan IP *address* dari sistem X, maka dapat dilakukan *tracerouting*.

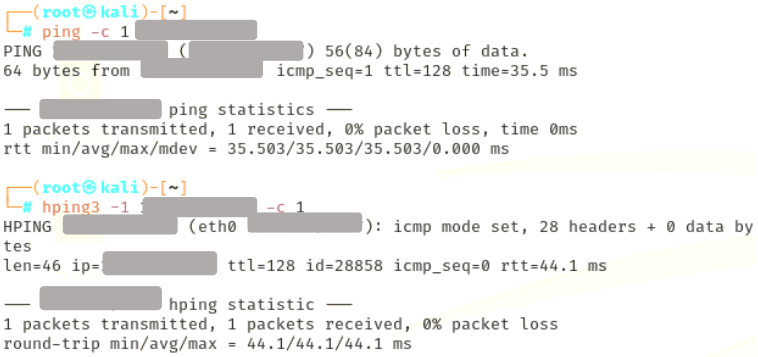
DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)



Gambar 3. *Traceroute* Sistem

Gambar 3 menunjukkan hasil yang didapat setelah melakukan *traceroute* terhadap target. *Traceroute* mengirimkan *hops* sebanyak 30 yang ditujukan pada target, kemudian target akan menerima *hops* tersebut dan mengirimkan balasan. Pada gambar 3 ditunjukkan bahwa target mengirmkan balasan akan tetapi tidak dapat terbaca dikarenakan pada target terdapat suatu alat yang akan melakukan *filter* sehingga paket yang dikirim ter*filter* dan balasan yang dikirim juga ter*filter* sehingga tidak dapat terbaca, meskipun demikian terdapat satu paket balasan yang dapat terbaca. Setelah dilakukan *traceroute*, maka dikirmkan *ping* untuk mengecek apakah sistem aktif atau tidak. Pengiriman *ping* menggunakan dua aplikasi yang berbeda, yakni *ping* dan *hping3*. Alasan digunakan dua aplikasi untuk mengirimkan *ping* adalah dengan menggunakan dua aplikasi yang berbeda maka hasil yang didapatkan akan semakin kuat untuk menunjukkan apakah sistem aktif atau tidak.



Gambar 4. *Ping* Sistem

Gambar 4 menunjukkan hasil yang diberikan oleh sistem X setelah dikirimkan *ping* dimana dengan menggunakan *ping* dan *hping*, sistem dapat mengembalikan paket yang dikirimkan. Dengan demikian, dapat disimpulkan bahwa sistem X sedang aktif. Kemudian dilakukan *fingerprint* OS dengan menggunakan *Nmap*.

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

```
┌──(root㉿kali)-[~]
└─# nmap -O [redacted]
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-25 03:37 WIB
Nmap scan report for [redacted] ([redacted])
Host is up (0.013s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
80/tcp   open  http
443/tcp  open  https
1723/tcp open  pptp
3389/tcp open  ms-wbt-server
Warning: OSScan results may be unreliable because we could not find at least
1 open and 1 closed port
Aggressive OS guesses: Actiontec MI424WR-GEN3I WAP (96%), DD-WRT v24-sp2 (Lin
ux 2.4.37) (96%), Microsoft Windows XP SP3 or Windows 7 or Windows Server 201
2 (96%), VMware Player virtual NAT device (96%), Linux 3.2 (93%), Linux 4.4 (
93%), Microsoft Windows XP SP3 (93%), BlueArc Titan 2100 NAS device (90%), Mi
crosoft Windows Server 2003 SP2 (89%), Hitachi BlueArc OS 7.0 (88%)
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.o
rg/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 64.76 seconds
```

Gambar 5. *Fingerprint* OS Sistem

Gambar 5 menunjukkan hasil yang didapat setelah melakukan *fingerprint* OS target. Setelah mengirimkan paket, maka target akan mengirimkan balasan seperti perintah yang dikirimkan. Pada gambar 5 ditunjukan bahwa sistem mengirimkan informasi berupa *port* mana saja yang terbuka serta adanya perkiraan OS yang digunakan oleh target sistem. Meski pada *fingerprint* OS didapatkan informasi *port* mana saja yang terbuka, perlu dilakukan pemindaian ulang untuk menguatkan informasi *port* mana saja yang terbuka pada sistem X.

```
┌──(root㉿kali)-[~]
└─# nmap [redacted]
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-25 03:40 WIB
Nmap scan report for [redacted] ([redacted])
Host is up (0.0010s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE
21/tcp   open  ftp
80/tcp   open  http
443/tcp  open  https
1723/tcp open  pptp
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 94.35 seconds
```

Gambar 6. Pemindaian *Port* Sistem

Gambar 6 menunjukkan hasil yang didapat setelah melakukan pemindaian *port* sistem. Pada gambar ditunjukkan bahwa target sistem memiliki lima *port* yang terbuka yakni, *port* 21, *port* 80, *port* 443, *port* 1723, *port* 3389, dan *port* 3389 dengan masing-masing layanan yang diberikan dari masing-masing *port*.

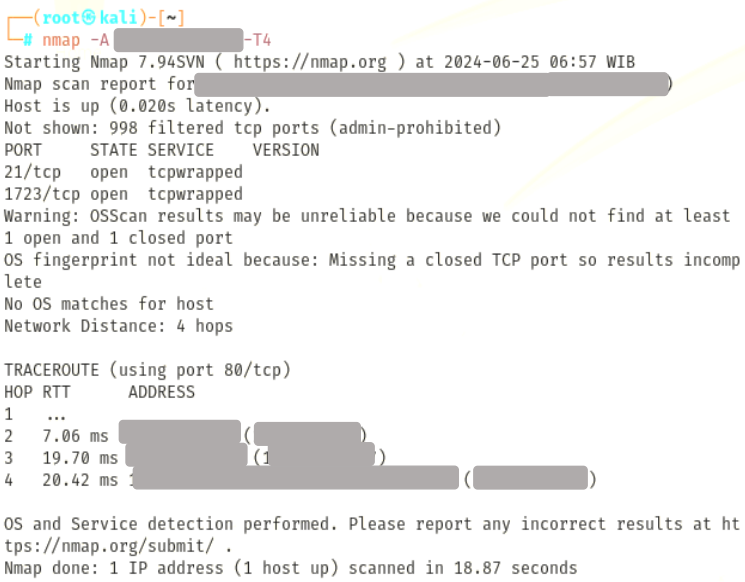
```
┌──(root㉿kali)-[~]
└─# nmap -sV [redacted] -f
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-25 03:46 WIB
Nmap scan report for [redacted] ([redacted])
Host is up (0.012s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE SERVICE    VERSION
21/tcp   open  tcpwrapped
80/tcp   open  tcpwrapped
443/tcp  open  tcpwrapped
1723/tcp open  tcpwrapped
3389/tcp open  tcpwrapped

Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 22.41 seconds
```

Gambar 7. Pemindaian Versi Sistem

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Gambar 7 menunjukkan hasil yang didapat setelah melakukan pemindaian versi *port* sistem. Pada gambar ditunjukkan bahwa sistem memiliki alat *filter* yang sensitif sehingga melakukan *filter* pada paket sehingga paket balasan yang dikirimkan tidak mengandung informasi yang diinginkan serta tidak didapatnya informasi mengenai layanan *port* meski pada tahapan sebelumnya, kita mendapat informasi layanan *port* tersebut.



Gambar 8. Pemindaian Agresif Sistem

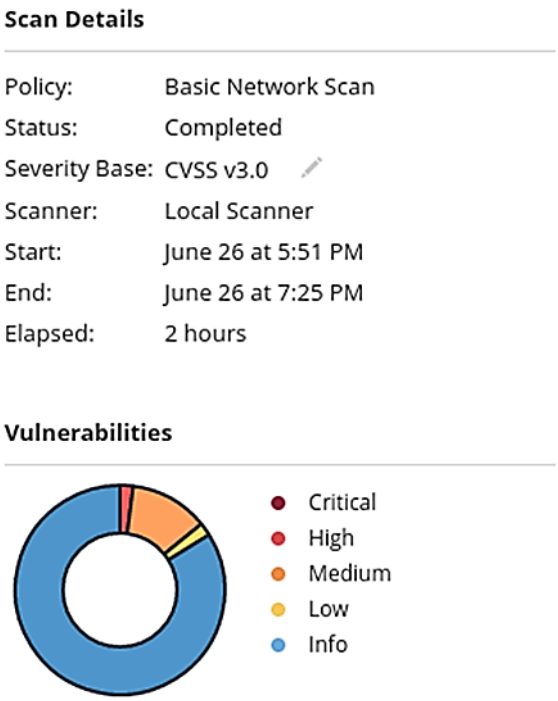
Gambar 8 menunjukkan hasil yang didapat dengan melakukan pemindaian agresif terhadap sistem. Pada gambar ditunjukkan bahwa terdapat beberapa informasi yang tidak dapat diterima karena ter*filter* dengan alat yang dimiliki oleh sistem. Beberapa informasi yang ter*filter* diantaranya adalah versi dan layanan yang ada pada *port* yang terbuka, sistem informasi yang digunakan oleh sistem. Tahapan ini memiliki hasil yang berbeda daripada tahapan-tahapan sebelumnya, akan tetapi tahapan ini tetap dilakukan karena untuk menunjang informasi tambahan yang bisa saja tidak didapatkan pada saat melakukan tahapan sebelumnya secara satu persatu.

3.2 Laporan Kerentanan

Dengan menggunakan *Nessus* untuk melakukan pemindaian kerentanan pada sistem X, maka didapatkan hasil pemindaian kerentanan yang diilustrasikan dengan diagram kerentanan serta rincian kerentanan-kerentanan yang dimiliki pada *firewall* sistem X. Pengujian keamanan menggunakan pemindaian dasar pada seluruh *port* serta pengujian kerentanan sistem secara menyeluruh dan kompleks pada *firewall* sistem X.

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)



Gambar 9. Hasil pemindaian kerentanan

Berdasarkan pada gambar 9, *nessus* menjabarkan lima kerentanan yang dapat ditemui di sistem X, yakni kritis, tinggi, sedang, rendah, dan informasi. Informasi yang ada pada diagram di atas bukan merupakan kerentanan, akan tetapi merupakan informasi yang perlu diperhatikan dan bisa menjadi catatan untuk pengembangan *website* ke depannya. Pada gambar 9 menunjukkan tingkat kerentanan yang dimiliki oleh sistem X. *Nessus* menjabarkan sistem X memiliki tingkat kerentanan tinggi, tingkat kerentanan sedang dan tingkat kerentanan rendah dimana sistem X memiliki banyak kerentanan tingkat sedang. Tabel 1 dapat digunakan untuk mengetahui rincian dari kerentanan-kerentanan yang dimiliki

Tabel 1. Kerentanan sistem X

Tingkat Kerentanan	CVSS V. 3.0.	Skor VPR	*Plugin*	Nama
Tinggi	7.5	5.1	42873	SSL Medium Strength Cipher *Suites* Supported (SWEET32)
Sedang	6.5	-	51192	SSL Certificate Cannot Be Trusted
Sedang	6.5	-	57582	SSL Self-Signed Certificate
Sedang	6.5	-	104743	TLS Version 1.0 Protocol Detection
Sedang	6.5	-	157288	TLS Version 1.1 Deprecated Detection
Sedang	5.9	4.4	658621	SSL RC4 Cipher *Suites* Supported (Bar Mitzvah)
Rendah	3.7	3.9	83875	SSL/TLS Diffie-Hellman Modulus <= 1024 (Logjam)
Informasi	N/A	-	45590	Common Platform Enumeration (CPE)

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Informasi	N/A	-	54615	Device Type
Informasi	N/A	-	84502	HSTS Missing From HTTPS Server
Informasi	N/A	-	69826	HTTP *Cookie* 'secure' Property Trans*port* Mismatch
Informasi	N/A	-	43111	HTTP Methods Allowed (per directory)
Informasi	N/A	-	10107	HTTP Server Type and Version
Informasi	N/A	-	12053	*Host* Fully Qualified Domain Name (FQDN) Resolution
Informasi	N/A	-	24260	HyperText Transfer Protocol (HTTP) Information
Informasi	N/A	-	46215	Inconsistent *Host*name and IP Address
Informasi	N/A	-	50344	Missing or Permissive Content-Security-Policy frame-ancestors HTTP Response Header
Informasi	N/A	-	50345	Missing or Permissive X-Frame-Options HTTP Response Header
Informasi	N/A	-	11219	*Nessus* SYN scanner
Informasi	N/A	-	19506	*Nessus* Scan Information
Informasi	N/A	-	11936	OS Identification
Informasi	N/A	-	10919	Open *Port* Re-check
Informasi	N/A	-	10940	Remote Desktop Protocol Service Detection
Informasi	N/A	-	56984	SSL / TLS Version Sup*port*ed
Informasi	N/A	-	45410	SSL Certificate 'commonName' Mismatch
Informasi	N/A	-	10863	SSL Certificate Information
Informasi	N/A	-	70544	SSL Cipher Block Chaining Cipher *Suites* Sup*port*ed
Informasi	N/A	-	21643	SSL Cipher *Suites* Sup*port*ed
Informasi	N/A	-	57041	SSL Perfect Forward Secrecy Cipher *Suites* Sup*port*ed
Informasi	N/A	-	94761	SSL *Root* Certification Authority Certificate Information
Informasi	N/A	-	51891	SSL Session Resume Sup*port*ed
Informasi	N/A	-	156899	SSL/TLS Recommended Cipher *Suites*
Informasi	N/A	-	22964	Service Detection
Informasi	N/A	-	25220	TCP/IP *Timestamp*s Sup*port*ed
Informasi	N/A	-	121919	TLS Version 1.1 Protocol Detection
Informasi	N/A	-	136318	TLS Version 1.2 Protocol Detection
Informasi	N/A	-	64814	Terminal Services Use SSL/TLS
Informasi	N/A	-	10287	Traceroute Information
Informasi	N/A	-	85601	Web Application *Cookie*s Not Marked HttpOnly

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Informasi	N/A	-	85602	Web Application *Cookies* Not Marked Secure
Informasi	N/A	-	91815	Web Application Sitemap
Informasi	N/A	-	11032	Web Server Directory Enumeration

Pada informasi kerentanan yang dapat dilihat pada tabel 1, terdapat empat informasi, yakni *Common Vulnerability Scoring System* (CVSS), nilai *Vulnerability Priority Rating* (VPR), *plugin*, serta nama kerentanan. CVSS dan VPR merupakan dua skor yang digunakan untuk menilai seberapa rentan suatu permasalahan yang timbul dari penilaian. *Plugin* merupakan program yang dibuat oleh *nessus* yang terdiri dari informasi kerentanan, perbaikan yang disederhanakan, serta algoritma untuk menguji kehadiran suatu permasalahan keamanan. Dengan menggunakan *plugin*, kita dapat mengetahui rincian kerentanan yang ada serta solusi yang ditawarkan. Tabel 1 menunjukkan bahwa sistem X memiliki satu kerentanan tingkat tinggi, lima kerentanan tingkat sedang, satu kerentanan tingkat rendah dan 35 informasi yang perlu diperhatikan. Dengan perincian kerentanan yang diberikan, maka pengujian dapat difokuskan pada kerentanan-kerentanan yang memiliki nilai kerentanan.

Tabel 2. Detail Kerentanan Sistem X

Tingkat Kerentanan	CVSS V. 3.0.	Nilai VPR	Plugin	Nama	Deskripsi
Tinggi	7.5	5.1	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)	Layanan jarak jauh mendukung penggunaan SSL chipers kekuatan sedang
Sedang	6.5	-	51192	SSL Certificate Cannot Be Trusted	Sertifikasi SSL pada layanan ini tidak dapat dipercaya
Sedang	6.5	-	57582	SSL Self-Signed Certificate	Rantai sertifikat untuk layanan ini berakhir pada sertifikat terta tangan sendiri yang tidak dikenali
Sedang	6.5	-	104743	TLS Version 1.0 Protocol Detection	Layanan jarak jauh yang terenkripsi pada lalu lintas menggunakan versi TLS yang lama
Sedang	6.5	-	157288	TLS Version 1.0 Deprecated Detection	Layanan jarak jauh yang terenkripsi pada lalu lintas menggunakan versi TLS yang lama
Sedang	5.9	4.4	658621	SSL RC4 Cipher Suites Supported (ar Mitzvah)	Layanan jarak jauh mendukung penggunaan sandi RC4
Rendah	3.7	3.9	83875	SSL/TLS Diffie-Hellman Modulus <= 1024 (Logjam)	*Host* jarak jauh memperbolehkan sambungan SSL/TLS dengan satu atau lebih modul Diffie-Hellman kurang dari atau sama dengan 1024 bit

Pada Tabel 2 ditunjukkan detail dari masing-masing kerentanan tingkat rendah, tingkat sedang, dan tingkat tinggi yang dimiliki pada sistem X yang terdiri dari tingkat kerentanan, nilai CVSS, nilai VPR, *plugin*, nama kerentanan dan deskripsi singkat dari kerentanan yang ada. Dengan menggunakan informasi yang ada pada tabel 2, maka pihak universitas dapat mengetahui kerentanan seperti apa yang dimiliki pada *firewall* sistem X dan apa maksud dari kerentanan yang ada.

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Tabel 3. Solusi Kerentanan Sistem X

Tingkat Kerentanan	CVSS V. 3.0.	Nilai VPR	Plugin	Nama	Solusi
Tinggi	7.5	5.1	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)	Konfigurasi ulang aplikasi yang terpengaruh dengan menggunakan sandi SSL dengan kekuatan sedang
Sedang	6.5	-	51192	SSL Certificate Cannot Be Trusted	Melakukan pembelian atau menghasilkan sertifikat SSL yang tepat untuk layanan ini
Sedang	6.5	-	57582	SSL Self-Signed Certificate	Melakukan pembelian atau menghasilkan sertifikat SSL yang tepat untuk layanan ini
Sedang	6.5	-	104743	TLS Version 1.0 Protocol Detection	Mengaktifkan layanan untuk TLS 1.2 dan 1.3 serta menonaktifkan layanan untuk TLS 1.0
Sedang	6.5	-	157288	TLS Version 1.1 Deprecated Detection	Mengaktifkan layanan untuk TLS 1.2 dan/atau 1.3 serta menonaktifkan layanan untuk TLS 1.0
Sedang	5.9	4.4	658621	SSL RC4 Cipher Suites Supported (ar Mitzvah)	Melakukan konfigurasi ulang aplikasi yang terpengaruh, dan jika bisa untuk menghindari penggunaan sandi RC4. Mempertimbangkan penggunaan TLS 1.2 dengan AES-CGM suites untuk layanan *browser* dan web-server
Rendah	3.7	3.9	83875	SSL/TLS Diffie-Hellman Modulus <= 1024 (Logjam)	Konfigurasi ulang layanan dengan penggunaan modul Diffie-Helman yang unik dengan 2048 bit atau lebih

Pada Tabel 3 ditunjukkan detail dari masing-masing kerentanan yang dimiliki pada sistem X yang terdiri dari tingkat kerentanan, nilai CVSS, nilai VPR, *plugin*, nama kerentanan dan solusi yang ditawarkan untuk kerentanan yang ada. Dengan menggunakan informasi yang ada pada tabel 2, pihak universitas dapat menerapkan tawaran solusi yang diberikan untuk memperbaiki kerentanan yang dimiliki pada *firewall* sistem X.

.

IV. KESIMPULAN

DOI: 10.52362/jisamar.v8i3.1563

Journal of Information System, Applied, Management, Accounting and Research.
http://journal.stmikjayakarta.ac.id/index.php/jisamar ,
jisamar@stmikjayakarta.ac.id , jisamar2017@gmail.com
e-ISSN: 2598-8719 (Online), p-ISSN: 2598-8700 (Printed) , Vol. 8 No.3 (August 2024)

Berdasarkan apa yang telah didapat dari bab sebelumnya, maka dapat disimpulkan pengujian keamanan *firewall* pada sistem X Universitas Z dapat dijalankan dengan cukup baik dimana didapatkan alamat IP beserta informasi-informasi yang ada pada jaringan *firewall* sistem X seperti *port* yang terbuka, layanan yang diberikan pada *port* yang terbuka, versi *port* yang terbuka, sistem operasi yang digunakan oleh sistem serta informasi tambahan yang diberikan melalui *scripting*. Akan tetapi, dikarenakan sistem X memiliki suatu alat sensitif yang dapat menyaring paket yang dikirimkan ke sistem dan mengembalikan paket secara parsial, maka informasi-informasi yang dibutuhkan untuk menunjang pengujian tidak dapat diperoleh secara maksimal meskipun hal ini tidak mengganggu jalannya pengujian. Selain itu, setelah dilakukan pemindaian kerentanan dengan menggunakan aplikasi *nessus*, maka didapatkan beberapa kerentanan yang dimiliki oleh *firewall* sistem X dengan rincian satu kerentanan tingkat tinggi, lima kerentanan tingkat sedang, satu kerentanan tingkat rendah, dan 35 informasi yang perlu diperhatikan beserta deskripsi kerentanan yang ada dengan penawaran solusi yang dapat dilakukan untuk memperbaiki kerentanan *firewall* sistem X. Dengan ditemukannya kerentanan-kerentanan yang ada pada *firewall* sistem X serta diberikannya solusi yang ditawarkan, diharapkan dapat menjadi bahan evaluasi kembali bagi pihak kampus serta diterapkannya beberapa solusi ditawarkan untuk meningkatkan keamanan *firewall* sistem X lebih baik lagi ke depannya.

REFERENASI

[1] Alhimni & Imam, Zuhri Yadi. Oktober. 2020. “Analisis Kinerja Virtual Router Redudancy Protocol (VRRP) di Mikrotik Router pada Dirjen Sumber Daya Air Balai Besar Wilayah Sungai Sumantera VIII”. Jurnal Pengembangan Sistem Informasi dan Informatika 4:233-243.
[2] Allen, L., Tedi H., Shakeel L., 2014. Kali Linux – Assuring Security by Penetration Testing. Livery Place: Packt Publishing Ltd.
[3] Ariyani, S. & Arta, W., 2017. “ATCS System Security Audit Using Nessus”. Journal of Information Engineering and Applications. 7, 3:25.
[4] Asnawi, Muhammad Fuat, M. Agung Nugroho. November. 2022. “Pengujian Keamanan Jaringan Menggunakan Metode Penetrasi Tes Pada Jaringan SMK Muhammadiyah 1 Wonosobo”. Jurnal Device 2:160-168.
[5] Bustami, Agustani & Syamsul Bahri. Agustus. 2020. “Ancaman, Serangan dan Tindakan Perlindungan pada Keamanan Jaringan atau Sistem Informasi: Systematic Review”. Jurnal Pendidikan dan Aplikasi Industri (UNISTEK) 2:60-70.
[6] Ganggupantulu, Rohit et al. Agustus. 2022. “Using Cyber Terrain in Reinforcement Learning for Penetration Testing”. IEEE
[7] Hasibuan, Marzuki & Andi Marwan Elhanafi. Desember. 2022.”Penetrartion Testing Sistem Jaringan Komputer Menggunakan Kali Linux Untuk Mengetahui Kerentanan Keamanan Server Dengan Metode Black Box Studi Kasus Web Server Diva Karaoke.co.id”. Jurnal Teknik Informatika 4:171-1777.
[8] Mastiara, Willy & Afriyudi. September. 2023. “Rancangan Blueprint Jaringan Komputer pada Hotel Amaris Palembang Menggunakan Metode Rekayasa Sistem jaringan Komputer (RSJK)”. Bina Darma Conference on Computer Science 2:448-452.
[9] Pramudita, Reza,Syifaul Fuada, Nuur Wachid Abdul Majid. April. 2020. “Studi Pustaka Tentang Kerentanan Keamanan E-Learning¬ dan Penanganannya”. Jurnal Media Informatika Budidarma 2:309-317
[10] Putra, Fauzan Prasetyo Eka et al. September. 2023. “Analisis Keamanan Jaringan dari Serangan Malware Menggunakan Firewall Filtering dengan Port Blocking”. Digital Transfomation Technology (Digitech) 2:857-863
[11] Rendro, Dwi Bayu, Ngatono, Wahyu Nugroho Aji. September. 2020. “Analisis Monitoring Sistem Keamanan Jaringan Komputer Menggunakan Software Nmap (Studi Kasus di SMK Negeri 1 Kota Serang)”. Jurnal PROSISKO 2:108-115.
[12] Rizal, Chairul et al. Maret. 2022. “Perancangan Server Kantor Desa Tomuan Holbung Berbasis Client Server”. Bulletin of Information Technology (BIT) 1:27-33.
[13] Ryan, Prima Posma. Avon Budiono, Ahmad Almaarif. 2019. “Implementasi dan Analisis Badusb Evilduino dengan Menggunakan Arduino Pro Micro pada Sistem Operasi Windows”. e-Proceeding of Engineering.
[14] Salama, Vivin & Alfhian Makmur. Januari. 2024. “Pembatasan Hak Akses Kinerja Jaringan WLAN Berbasis Linux Ubuntu pada SMK Kristen Pada Sappa”. Bandwith: Journal of Informatics and Computer Engineering 1:20-37.
[15] Santoso, Nugroho Adi, Khaediar Bagus Affandi, Rifki Dwi Kurniawan. Oktober. 2022. “Implementasi Keamanan Jaringan Menggunakan Port Knocking”. Jurnal Janitra Informatika dan Sistem Informasi 2:90-95.
[16] Saputra, A. A. G., 2019. Tugas Laporan Keamanan Jaringan Komputer, <URL: http://edocs.ilkom.unsri.ac.id/3872/1/09011181621004_KJK.pdf>

DOI: 10.52362/jisamar.v8i3.1563

[17] Saputro, Andik, Nanang Saputro, Hendro Wijayanto. November. 2020. "Metode Demilitarized Zone dan Port Knocking untuk Keamanan Jaringan Komputer". CyberSecurity dan Forensik Digital 2:22-27.
[18] Sari, Ayu Purnama, Sulistiyono, Naga Kemala. September. 2020. "Perancangan Jaringan Virtual Private Network Berbasis IP Security Menggunakan Router Mikrotik". Jurnal PROSISKO 2:150-164.
[19] Sutrisna, Cecep. Desember. 2021. "Aspek Hukum Perlindungan Data Pribadi dan Kondisi Darurat Kebocoran Atas Data Pribadi di Indonesia". Wacana Paramarta Jurnal Ilmu Hukum 5:1-10.
[20] Tudosi, Andrei-Daniel et al. Maret. 2023. "Research on Security Weakness Using Penetration Testing in a Distributed Firewall". Sensors 23:1-18.

DOI: 10.52362/jisamar.v8i3.1563