# MODUL 3

# *VULNERABILITY SCANNING* & ASSESMENT

## 3.1 Tujuan Praktikum

1. Dapat mengenal dan memahami konsep dari Teknik *Scanning*
2. Dapat memahami cara kerja dari *Network Scanning*.
3. Dapat melakukan dasar *Network Scanning* menggunakan Nmap.

## 3.2 Alat dan Bahan

1. Laptop
2. VirtualBox
3. OS Kali Linux
4. Nmap
5. Nikto
6. Owasp ZAP
7. Dirsearch

## 3.3 Dasar Teori

### 3.3.1 Pengertian *Scanning*

*Scanning* adalah proses identifikasi port dan layanan yang terbuka pada sistem target. Teknik ini bertujuan untuk menemukan kerentanan dan celah keamanan yang mungkin ada pada sistem yang diincar. *Scanning* dilakukan setelah proses *Footprinting* untuk memperoleh informasi lebih detail tentang sistem jaringan target, sehingga dapat dikembangkan rencana serangan yang lebih efektif.

*Scanning* dapat dibedakan menjadi 3, yaitu:

1. *Port Scanning*, dilakukan untuk mengetahui service apa yang dijalankan oleh target berdasarkan well known ports.
2. *Network Scanning*, dilakukan untuk mengetahui aktifnya suatu host dan IP *Address* dari host tersebut.
3. *Vulnerability Scanning*, dilakukan untuk mengetahui sistem operasi, versi sistem operasi, maupun service pack yang digunakan.

### 3.3.2 Tujuan *Scanning*

1. Mengetahui port yang aktif pada port *scanner*
2. Mengetahui jalur *network* yang aktif menuju site tujuan
3. Mengetahui *vulnerability scanning* pada site yang terdeteksi

### 3.3.3 Jenis-Jenis *Scanning*

1. *Basic Scan*

*Basic Scan* dalam konteks pemindaian keamanan jaringan adalah jenis pemindaian yang dilakukan untuk mengidentifikasi port yang terbuka pada host atau perangkat dalam jaringan target. *Basic Scan* umumnya tidak mencoba mengidentifikasi sistem operasi atau layanan yang berjalan pada port-port tersebut, fokus utamanya adalah pada keterbukaan port. *Basic Scan* dapat digunakan sebagai langkah awal dalam pemindaian lebih lanjut diperlukan untuk mendapatkan pemahaman yang lebih mendalam tentang konfigurasi dan keamanan host yang bersangkutan.

Berikut adalah beberapa karakteristik dari *Basic Scan*:

- Pemindaian Port

  *Basic Scan* mencoba menghubungi host target pada berbagai port untuk melihat apakah port-port tersebut terbuka atau tertutup. Port terbuka menunjukkan bahwa ada potensi layanan atau aplikasi yang berjalan di port tersebut.
- Hasil yang terbatas

  Hasil dari *Basic Scan* mencoba menghubungi host target pada berbagai port untuk melihat apakah port-port tersebut terbuka atau tertutup. Port terbuka menunjukkan bahwa ada potensi layanan atau aplikasi yang berjalan di port tersebut.
- Kecepatan *Basic Scan* cenderung lebih cepat dibandingkan dengan pemindaian yang lebih mendalam seperti pemindaian OS (*Operating System*) atau pemindaian layanan. Hal ini karena *Basic Scan* hanya mencoba menghubungi port-port tanpa memeriksa respons lebih lanjut dari host.

- Kemungkinan lebih tidak terdeteksi

  Karena *Basic Scan* tidak mencoba mengidentifikasi sistem operasi atau layanan yang berjalan di balik port yang terbuka. Namun, ini juga berarti bahwa *Basic Scan* mungkin lebih sulit untuk mendeteksi jika ada layanan atau aplikasi yang berjalan di port tertentu yang tidak biasa atau tidak standar.

2. Syn *Scan*

   Syn *Scan* adalah teknik yang dapat digunakan oleh para *hacker* untuk menentukan status port komunikasi tanpa membuat koneksi penuh. Teknik ini terkadang digunakan untuk melakukan serangan *Denial of Service* (DoS). Syn *Scan* juga dikenal sebagai *half open scanning*.

   Cara kerja dari Syn *Scan* mirip dengan port *scan*, pelaku ancaman mencoba mengatur koneksi Transmisi Control Protocol/Internet Protocol (TCP/IP) dengan server di setiap port yang memungkinkan. Hal ini dilakukan dengan mengirimkan paket SYN (sinkronisasi), seolah-olah memulai jabat tangan tiga arah, ke setiap port di server. Jika server membalas dengan respons ACK (pengakuan) atau paket SYN/ACK (*synchronization acknowledge*) dari port tertentu, bnerarti port tersebut terbuka. Kemudian, klien yang bermusuhan mengirimkan paket RST (*reset*).

   Akibatnya, server berasumsi bahwa telah terjadi kesalahan komunikasi dan klien belum membuat sambungan. Dalam skenario ini, asumsi tersebut salah. Pelabuhan terbuka tetap terbuka dan rentan terhadap eksploitasi. Jika server merespons dengan paket RST dari port tertentu, ini menunjukkan bahwa port tersebut ditutup dan tidak dapat dieksploitasi. Ketika seorang peretas terus-menerus mengirimkan paket SYN dalam jumlah besar ke server, ia dapat menghabiskan sumber daya server. Hasilnya adalah hanya sedikit atau tidak ada komunikasi dari klien yang sah yang dapat dilakukan.

   Namun, Syn *Scan* memiliki beberapa keterbatasan dan potensi masalah, yaitu:

   - Deteksi oleh IDS/IPS: Syn *scan* dapat terdeteksi oleh Intrusion Detection Systems (IDS) atau Intrusion Prevention Systems (IPS) karena memicu proses koneksi yang tidak lengkap. Hal ini dapat menghasilkan tindakan pencegahan yang menghentikan pemindaian.

- Kebijakan Keamanan: Beberapa host atau jaringan dapat mengimplementasikan kebijakan keamanan yang membatasi atau memblokir permintaan SYN yang datang dari luar.
- False Positives: Syn *Scan* dapat menghasilkan hasil yang tidak akurat jika respons yang diterima tidak jelas, seperti ketika host target tidak merespons atau merespons dengan cara yang tidak diharapkan.

3. TCP *Scan*

TCP *Scan*, juga dikenal sebagai TCP connect *scan*, adalah salah satu teknik pemindaian keamanan jaringan yang digunakan untuk mengidentifikasi port yang terbuka pada host target dalam jaringan. Teknik ini termasuk dalam kategori pemindaian port dan merupakan salah satu metode yang paling umum digunakan dalam pemindaian keamanan.

Berikut adalah cara kerja TCP *scan*, yaitu:

- Pemindaian Koneksi: Dalam TCP *scan*, pemindai mencoba untuk membuat koneksi TCP lengkap ke host target pada setiap port yang ingin diperiksa. Ini dilakukan dengan mengirimkan permintaan TCP SYN (synchronize) ke port yang dituju.
- Respons Host: Jika port tersebut terbuka, host target akan merespons dengan sebuah paket TCP SYN-ACK (synchronize-acknowledgment), yang menandakan bahwa port tersebut dalam keadaan terbuka dan siap menerima koneksi. Jika port tersebut tertutup, host akan merespons dengan sebuah paket TCP RST (reset) sebagai tanda bahwa port tersebut tidak aktif.
- Penutupan Koneksi: Setelah mendapatkan respons SYN-ACK atau RST, pemindai akan menutup koneksi dengan mengirimkan paket TCP RST, sehingga tidak ada koneksi yang sebenarnya terjalin dengan host target.

Keuntungan dari TCP *scan* adalah kemampuannya untuk memberikan informasi yang akurat tentang keadaan keterbukaan port pada host target. Hasil pemindaian ini biasanya lebih dapat diandalkan dibandingkan dengan teknik pemindaian lainnya. Namun, ada beberapa keterbatasan dan masalah yang perlu diperhatikan:

- Kecepatan: TCP *scan* dapat memakan waktu lebih lama daripada teknik pemindaian lain yang lebih cepat, seperti Syn *Scan*, karena melibatkan proses pembentukan koneksi lengkap.
- Deteksi oleh IDS/IPS: Koneksi lengkap yang dibentuk oleh TCP *scan* dapat lebih mudah dideteksi oleh Intrusion Detection Systems (IDS) atau Intrusion Prevention Systems (IPS).
- Jejak Log: Karena melakukan koneksi lengkap, TCP *scan* dapat meninggalkan jejak log di host target, yang dapat terdeteksi oleh administrator jaringan.

4. UDP *Scan*

UDP *scan* adalah salah satu teknik dalam pemindaian keamanan jaringan yang digunakan untuk mengidentifikasi port yang terbuka pada sebuah host yang menjalankan layanan menggunakan protokol UDP (*User Datagram Protocol*). UDP adalah protokol transport yang berbeda dari TCP (*Transmission Control Protocol*) yang lebih umum digunakan. Perbedaan utama antara UDP dan TCP adalah bahwa UDP tidak memiliki mekanisme koneksi atau konfirmasi pengiriman data seperti yang dimiliki oleh TCP.

Dalam UDP *scan*, seorang penyerang akan mencoba mengirimkan paket UDP ke berbagai port pada host target dan mengamati responsenya. Respons yang berbeda dapat mengindikasikan apakah port tertentu terbuka atau tertutup.

Berikut adalah beberapa hasil yang mungkin dalam UDP *scan*:

- Port Terbuka: Jika host merespons dengan paket UDP yang valid atau respons khusus yang menunjukkan bahwa port tersebut aktif, maka port tersebut dianggap terbuka. Port terbuka dapat menunjukkan adanya layanan yang berjalan di port tersebut.
- Port Tertutup: Jika host merespons dengan paket ICMP (Internet Control Message Protocol) yang menunjukkan bahwa port tersebut tidak dapat dijangkau atau jika tidak ada respons sama sekali, maka port tersebut dianggap tertutup. Ini berarti tidak ada layanan yang berjalan di port tersebut.
- Port Filt: Dalam beberapa kasus, firewall atau perangkat keamanan dapat menghasilkan respons yang berbeda untuk port yang terbuka dan port yang tertutup. Ini dapat menunjukkan bahwa port tersebut dilindungi oleh firewall atau perangkat keamanan yang memblokir akses dari luar.

5. OS *Scan*

OS *Scan*, singkatan dari "*Operating System Scan*," adalah teknik pemindaian keamanan jaringan yang digunakan untuk mencoba mengidentifikasi sistem operasi yang digunakan oleh host atau perangkat dalam jaringan target. Tujuan dari OS *Scan* adalah untuk mengumpulkan informasi tambahan tentang host target, yang dapat membantu peneliti keamanan atau administrator jaringan dalam memahami lingkungan jaringan mereka dan mengidentifikasi potensi kerentanan.

OS *scan* dapat melibatkan berbagai metode, termasuk:

- Fingerprinting: Metode ini mencoba mengidentifikasi sistem operasi host target berdasarkan respons yang dihasilkan oleh host ketika menerima permintaan dari pemindaian. Respons ini mungkin mencakup berbagai parameter, seperti perilaku TCP/IP yang tidak standar, penggunaan opsi TCP/IP tertentu, atau respons terhadap paket tertentu.
- Analisis Banner: Beberapa layanan jaringan mengirimkan banner atau informasi identifikasi yang menyertakan informasi tentang sistem operasi yang digunakan. OS *scan* dapat mencoba mengambil informasi ini dari banner yang dikirimkan oleh layanan.
- Analisis Pola Paket: OS *scan* juga dapat melibatkan analisis pola paket yang dikirimkan oleh host target. Sistem operasi berbeda dapat memiliki karakteristik pola paket yang berbeda yang dapat digunakan untuk mengidentifikasinya.
- Analisis Respons Paket: Metode ini melibatkan pengamatan terhadap respons paket dari host target terhadap permintaan yang dikirimkan selama pemindaian. Respons ini dapat mengandung tanda-tanda yang mengungkapkan sistem operasi yang digunakan.

6. *Service Version Scan*

Merupakan teknik pemindaian keamanan jaringan yang digunakan untuk mengidentifikasi versi atau rincian lebih lanjut tentang layanan atau aplikasi yang berjalan di port-port tertentu pada host target dalam jaringan. Tujuan dari teknik ini adalah untuk mengumpulkan informasi yang lebih mendalam tentang layanan yang berjalan pada port-port terbuka, termasuk versi perangkat lunak yang digunakan. Dengan mengetahui versi perangkat lunak yang spesifik, peneliti keamanan atau

administrator jaringan dapat menilai potensi kerentanan yang mungkin ada pada layanan tersebut.

Cara kerja *service version scan* adalah sebagai berikut:

- Pemindaian Port Terbuka: Sebelum melakukan service version *scan*, biasanya dilakukan pemindaian port terbuka menggunakan teknik pemindaian port lainnya, seperti TCP connect *scan* atau Syn *Scan*, untuk mengidentifikasi port-port yang terbuka pada host target.
- Permintaan Informasi Versi: Setelah mengetahui port-port yang terbuka, teknik service version *scan* akan mengirimkan permintaan khusus yang dirancang untuk mengidentifikasi versi atau rincian lebih lanjut tentang layanan yang berjalan pada port tersebut.
- Analisis Respons: Setelah mengirimkan permintaan, teknik ini akan menganalisis respons yang diterima dari host target. Respons tersebut mungkin mengandung informasi seperti nama perangkat lunak, versi, atau rincian konfigurasi lainnya yang dapat membantu dalam mengidentifikasi layanan tersebut.
- Pemantauan dan Pelaporan: Hasil dari service version *scan* akan dicatat dan dapat digunakan untuk mengidentifikasi potensi kerentanan yang mungkin ada pada layanan yang berjalan pada host target. Informasi ini juga dapat digunakan untuk pemantauan dan pemeliharaan keamanan jaringan.

### *3.3.4* Cara Kerja *Vulnerability Scanning* & *Assesment*

*Vulnerability Assessment* atau evaluasi kerentanan, adalah proses untuk mengidentifikasi, mengevaluasi, dan mengukur kerentanan atau celah keamanan dalam infrastruktur IT, sistem jaringan, perangkat lunak, dan aplikasi perusahaan. Tujuan dari *Vulnerability Assessment* atau uji kerentanan adalah untuk mengidentifikasi potensi titik lemah atau kerentanan yang dapat dieksploitasi oleh pihak yang tidak bertanggung jawab, seperti penyerang atau peretas, guna merusak, mencuri, atau mengganggu operasional perusahaan.

Berikut adalah tahapan umum dari cara kerja *Vulnerability Assessment*:

- Perencanaan dan Persiapan

- Tentukan tujuan dan cakupan *assessment*, termasuk infrastruktur, aplikasi, dan sistem yang akan dievaluasi.
- Identifikasi sumber daya yang akan digunakan, termasuk perangkat lunak, alat, dan sistem analisis keamanan yang sesuai.
- Pastikan adanya izin dan otorisasi yang diperlukan sebelum melakukan asesmen untuk menghindari potensi masalah hukum atau privasi.

➢ Pengumpulan Informasi
- Kumpulkan informasi tentang infrastruktur IT perusahaan yang akan dinilai, seperti alamat IP, sistem operasi, aplikasi, dan konfigurasi jaringan.
- Identifikasi semua host (perangkat) dan layanan yang berjalan dalam jaringan.

➢ Pengenalan dan Pendekatan Pemindaian
- Lakukan pemindaian awal untuk mengenali semua host yang aktif dalam jaringan.
- Pilih pendekatan pemindaian yang sesuai, seperti pemindaian berbasis agen, pemindaian jaringan, atau pemindaian kode sumber (untuk aplikasi web).

➢ Pemindaian Kerentanan
- Gunakan alat keamanan khusus untuk memindai host dan layanan dalam jaringan. Alat ini akan mencari kerentanan yang telah diketahui atau celah umum yang dapat dieksploitasi oleh penyerang.
- Pemindaian ini mencakup pengujian keamanan yang mencari kerentanan seperti kelemahan sistem operasi, aplikasi yang tidak diperbarui, konfigurasi yang tidak aman, serta celah keamanan di tingkat jaringan dan aplikasi.

➢ Analisis Hasil Pemindaian
- Tinjau hasil pemindaian untuk mengidentifikasi dan memahami kerentanan yang ditemukan.
- Nilai tingkat risiko dari masing-masing kerentanan berdasarkan potensi dampak dan probabilitas eksploitasi.

➢ Pelaporan
- Buat laporan yang jelas dan terperinci tentang kerentanan yang ditemukan beserta rekomendasi tindakan korektif.

- Laporan harus mencakup langkah-langkah yang dapat diambil untuk mengatasi atau mengurangi risiko yang terkait dengan setiap kerentanan.

➢ Tindakan Korektif

- Implementasikan tindakan korektif yang direkomendasikan sesuai dengan prioritas dan tingkat urgensi.
- Pastikan bahwa semua kerentanan yang signifikan ditangani dan diselesaikan.

➢ Monitoring dan Evaluasi Lanjutan

- Pantau Keamanan infrastruktur secara berkala dengan Vilnerability Assessment yang berulang
- Evaluasi efektivitas langkag-langkah korektif yang diimplementasikan dan pastikan bahwa sistem tetap terlindungi dari kerentanan baru yang muncul

### 3.3.5 Tahapan aktivasi *hacking* yang didefinisikan dalam sertifikasi CEH

➢ *Reconnaissance*

*Reconnaissance* adalah tahap mengumpulkan data di mana hacker akan mengumpulkan semua data sebanyak-banyaknya mengenai target.

➢ *Scanning*

*Scanning* merupakan tanda dari dimulainya sebuah serangan *hacker* (*preattack*). Melalui *scanning* ini, hacker akan mencari berbagai kemungkinan yang bisa diguankan hacker untuk mengambil alih komputer korban.

➢ *Gaining Access*

Melalui sebuah informasi yang didapatkan, hacker akan mulai menyerang computer korban untuk menyerang computer korban untuk menguasainya.

➢ *Maintaining Access*

Setelah mendapatkan akses ke computer korban, hacker biasanya ingin tetap menguasai computer tersebut.

➢ *Covering Tracks*

Biasanya hacker akan berusaha menutup jejak mereka dengan cara menghapus log file serta menutup semua jejak yang mungkin ditinggalkan.

### 3.1.1 Tools *Vulnerability Scanning* & Assesment

- Nmap

Nmap (Network Mapper) merupakan suatu *opensource tools* yang biasanya digunakan untuk eksplorasi, information gathering, dan *vulnerability scanning* sebuah jaringan. Tools ini merupakan ciptaan seorang ahli *cyber security* Bernama Gordon Lyon pada tahun 1997 dan menjadi salah satu tools yang paling sering digunakan dalam *hacking* dan penetration testing karena kemampuannya memindai jaringan yang terhubung ke computer/*machine* target dan fitur-fitur lainnya yang membantu pengguna memahami detail pada sebuah jaringan. Nmap sendiri dapat digunakan secara gratis pada semua OS seperti Windows, Mac OS, Linux (semua distro), OpenBSD, FreeBSD, dan OS-OS lainnya. Perlu diketahui juga, Nmap menjadi tools bawaan pada beberapa OS tertentu. Salah satu OS yang sudah menyediakan Nmap secara default adalah Kali Linux.

- OwaspZAP

OWASP ZAP adalah alat pengujian penetrasi yang membantu pengembang dan profesional keamanan mendeteksi dan menemukan kerentanan dalam aplikasi web. ZAP adalah dikenal sebagai "*proxcy man-in-the-middle*." Itu berdiri di antara browser dan aplikasi web. Saat Anda menavigasi semua fitur situs web, itu menangkap semua tindakan. Kemudian menyerang situs web dengan teknik yang diketahui untuk menemukan kerentanan keamanan.

Saat ZAP spider aplikasi web, ia membangun peta halaman aplikasi web dan sumber daya yang digunakan untuk merender halaman tersebut. Kemudian merekam permintaan dan respons yang dikirim ke setiap halaman dan membuat peringatan jika ada sesuatu yang berpotensi salah dengan permintaan atau respons. OWASP ZAP melakukan beberapa fungsi keamanan termasuk:

- Memindai permintaan web secara pasif
- Menggunakan daftar kamus untuk mencari file dan folder di server web
- Menggunakan *crawler* untuk mengidentifikasi struktur situs dan mengambil semua link dan URL
- Mencegat, menampilkan, memodifikasi, dan meneruskan permintaan web antara browser dan aplikasi web

OWASP ZAP dapat mengidentifikasi kerentanan dalam aplikasi web termasuk otentikasi yang disusupi, paparan data sensitif, kesalahan konfigurasi keamanan, injeksi SQL, skrip lintas situs (XSS), deserialisasi yang tidak aman, dan komponen dengan kerentanan yang diketahui.

- Nikto

Nikto adalah alat pemindai / tools *scanning* kerentanan, dibuat dengan bahasa pemrograman Perl dan awalnya dirilis pada akhir 2001, yang menyediakan pemindaian / *scanning* kerentanan tambahan khusus untuk server web, artinya Nikto termasuk dalam tools / alat CGI scanner.

Nikto melakukan pemindaian pada website tertarget dengan mengirim perintah *request* sederhana yang akan melakukan uji coba atau pencarian informasi terkait link, file, *software*, log, dan informasi server. Dengan mencocokan versi pada target ke database yang berisi daftar versi yang memiliki celah keamanan dan menampilkannya sebagai informasi yang bisa dimanfaatkan pengguna nikto.

Fitur-fitur Nikto:

- Laporan output dalam HTML atau teks biasa
- Pergantian otomatis versi HTTP yang tersedia
- Pemeriksaan Software server umum dan khusus
- Dukungan SSL
- Dukungan proxy
- Dukungan cookie
- Menemukan sub-domain
- Memberikan rincian perangkat lunak (software) yang diinstal
- Mengambil file Nmap sebagai input untuk memindai port di server web.
- Mampu melakukan dictionary attack.

- Dirsearch

Dirsearch adalah alat yang melakukan serangan *bruteforce* terhadap direktori dan file sensitif yang ditemukan di situs web. Dirsearch memberikan pengguna kesempatan untuk melakukan penemuan konten web yang kompleks, dengan banyak vektor untuk daftar kata, akurasi tinggi, kinerja yang mengesankan, pengaturan koneksi/permintaan tingkat lanjut, teknik brute-force modern, dan keluaran yang bagus.