Purple team bootcamp

Nmap guide

Prepared By:
Kazim Ali Obad

Supervisor:
Anmar Mohammed

Mohammed baqer

2026/2/14

Contents

Main Concepts and Terminology:

Active Scanning vs. Passive Scanning

1. Active scanning sends requests and actively interacts with the target network to gather information
2. Passive scanning involves monitoring network traffic without sending probes. This is like observing the body from a distance - gathering information without direct interaction. While both have value, active scanning is more effective for detecting issues in real-time.

الفحص النشط يرسل استفسارات ويتفاعل مباشرة مع شبكة الهدف لجمع المعلومات.

الفحص السلبي يتضمن مراقبة حركة المرور الشبكية دون إرسال استفسارات. هذا مثل مراقبة الجسم من مسافة - جمع المعلومات دون التفاعل المباشر. كلاهما له قيمته، ولكن الفحص النشط أكثر فعالية لاكتشاف المشاكل في الوقت الفعلي.

2. Nmap (Network Mapper)
used for **network discovery** and **security auditing** . Nmap helps us identify which devices are on a network and what services they are running,. It's essential for penetration testing and network management, giving us insight into where vulnerabilities might lie.

- **Nmap** is a tool used for network discovery and vulnerability scanning.
- It helps identify devices and services on a network
- Essential for **penetration testing** and improving network security.

Scanning Process:

identifying which **ports** on a device are open and may allow traffic in or out. Think of **port scanning** as checking the entrances and exits of a building. **Service discovery** tells us what services are available on these open ports, similar to checking what organs or functions are operating in the body. Finally, **operating system fingerprinting** allows us to determine the "health" of a device by

identifying the operating system**Port scanning** identifies which ports on a device are open and can allow communication.

- **Service discovery** determines what services are running on those open ports (e.g., web services on port 80).
- **Operating system fingerprinting** helps detect the OS of a device, similar to diagnosing a patient's condition.

حيث يحدد المنافذ التي هي مفتوحة على الجهاز وقد تسمح بمرور البيانات الداخلة والخارجة. فكر في فحص المنافذ كفحص المداخل والمخارج لمبنى .اكتشاف الخدمات يخبرنا ما هي الخدمات المتاحة على هذه المنافذ المفتوحة التعرف على نظام التشغيل يسمح لنا بتحديد "صحة" الجهاز من خلال معرفة نظام التشغيلفحص المنافذ يحدد المنافذ التي هي مفتوحة على الجهاز ويمكنها السماح بالتواصل.

- اكتشاف الخدمات يحدد ما هي الخدمات التي تعمل على هذه المنافذ المفتوحة (مثل خدمات الويب على المنفذ 80).
- التعرف على نظام التشغيل يساعد في الكشف عن نظام التشغيل للجهاز، مثلما يتم تشخيص حالة المريض.

Key Techniques:

1. **Network Discovery with Nmap:**
   Imagine you are hired by a company to conduct a **penetration test**. The first step is always **network discovery**. You use Nmap to scan the network and identify which devices are online, and what services they provide. This step is crucial before trying to access any network resources,

- **Network discovery** is the first step in penetration testing.
- Nmap helps identify which devices and services are active on a network, similar to an initial diagnostic check.

اكتشاف الشبكة باستخدام:Nmap
تخيل أنك تم توظيفك من قبل شركة لإجراء اختبار اختراق .الخطوة الأولى دائمًا هي اكتشاف الشبكة، تستخدم Nmap لفحص الشبكة وتحديد الأجهزة المتصلة، وما هي الخدمات التي تقدمها. هذه الخطوة حاسمة قبل محاولة الوصول إلى أي موارد شبكية، اكتشاف الشبكة هو الخطوة الأولى في اختبار الاختراق.

- يساعد Nmap في تحديد الأجهزة والخدمات النشطة على الشبكة

2. Open Ports:

Ports are like doors to a server. If a door is open, someone can enter. A **port scan** is like checking the doors to ensure they are secure. For example, a server may have port 80 open for HTTP traffic or port 21 for FTP. Knowing which doors are open helps determine how a server can be accessed. A closed or filtered port, like a locked door, prevents unauthorized access.

- **Open ports** are entry points for communication with a server or device.
- A **port scan** helps identify these open doors and understand the potential access points for attackers.
- **Closed or filtered ports** prevent communication, like locked doors to secure areas.

فهم المنافذ المفتوحة:

المنافذ مثل الأبواب لخادم. إذا كان الباب مفتوحًا، يمكن للمرء الدخول. فحص المنافذ مثل التحقق من الأبواب للتأكد من أنها آمنة. على سبيل المثال، قد يحتوي الخادم على منفذ 80 مفتوحًا لمرور حركة HTTPأو المنفذ 21 لـ FTP. معرفة الأبواب المفتوحة يساعد في تحديد كيفية الوصول إلى الخادم. المنفذ المغلق أو المفلتر، مثل الباب المقفل، يمنع الوصول غير المصرح به.

- **المنافذ المفتوحة هي نقاط الدخول للتواصل مع الخادم أو الجهاز.**
- **يساعد فحص المنافذ في تحديد هذه الأبواب المفتوحة وفهم نقاط الوصول المحتملة للمهاجمين.**
- **المنافذ المغلقة أو المفلترة تمنع الاتصال، مثل الأبواب المقفلة للمناطق المحمية.**

3. Service and OS Identification:

After identifying open ports, Nmap can provide information about the services running on these ports. Nmap also helps to detect the operating system (OS) of the device Identifying these services and OS types helps in understanding what might be vulnerable to exploitation.

- **Service identification** reveals which applications or services are running on open ports (e.g., HTTP, FTP).
- **Operating system identification** helps in recognizing potential vulnerabilities based on the OS and its version.
- Knowing the services and OS gives insights into possible **exploits** based on known vulnerabilities in those services.
-

التعرف على الخدمات ونظام التشغيل:
بعد تحديد المنافذ المفتوحة، يمكن لـ Nmap تقديم معلومات حول الخدمات التي تعمل على هذه المنافذ.
يساعد Nmap أيضًا في الكشف عن نظام التشغيل (OS) للجهازيساعد تحديد هذه الخدمات وأنواع أنظمة التشغيل في فهم ما قد يكون عرضة للاستغلال.

- **التعرف على الخدمات يكشف ما هي التطبيقات أو الخدمات التي تعمل على المنافذ المفتوحة) مثل HTTP، FTP).**
- **التعرف على نظام التشغيل يساعد في التعرف على الثغرات المحتملة بناءً على نظام التشغيل وإصداره.**
- **معرفة الخدمات ونظام التشغيل يعطي رؤى حول الثغرات المحتملة استنادًا إلى الثغرات المعروفة في هذه الخدمات.**

Key Insights:

1. **Why Scanning Is Critical:** a network engineer must understand the network's structure and vulnerabilities before attempting to secure it. Scanning helps you **map the network**, identify vulnerable services, and understand the attack surface. This step is essential in creating a defense strategy.

- **Scanning** is essential to understand the network and its weaknesses before any defense measures are implemented.
- It helps in **mapping the network**, identifying services that could be exploited, and preparing defense strategies.

1. لماذا يعد الفحص أمرًا بالغ الأهمية:
يجب على مهندس الشبكة فهم هيكل الشبكة والثغرات قبل محاولة تأمينها. يساعد الفحص في رسم خريطة الشبكة، وتحديد الخدمات المعرضة للهجوم، وفهم مساحة الهجوم. هذه الخطوة أساسية في إنشاء استراتيجية دفاعية.

- **الفحص أمر حيوي لفهم الشبكة وضعفها قبل تطبيق أي تدابير دفاعية.**
- **يساعد في رسم خريطة الشبكة، وتحديد الخدمات التي قد يتم استغلالها، وتحضير استراتيجيات الدفاع.**

2. Best Practices for Scanning:
Use a **combination of scans** (SYN, TCP, UDP) for a comprehensive analysis. Always **verify results** to ensure the accuracy of your findings. Be aware of **firewalls and filters** that may block your scans

- Using a **combination of scans** gives a fuller picture of the network's status.
- Always verify **scan results** to ensure they are correct
- Be mindful of **firewalls** and **network filters** that could block scanning efforts.

2.أفضل الممارسات للفحص:
استخدم مزيجًا من الفحوصات (SYN، TCP، UDP) للحصول على تحليل شامل تحقق دائمًا من النتائج لضمان دقة النتائج التي حصلت عليها. كن على دراية بـ جدران الحماية والفلاتر التي قد تمنع الفحوصات

- **استخدام مزيج من الفحوصات يوفر صورة أكمل عن حالة الشبكة.**
- **تحقق دائمًا من نتائج الفحص لضمان دقتها**
- **كن حذرًا من جدران الحماية و الفلاتر الشبكية التي قد تمنع محاولات الفحص.**
-

Advanced Techniques in Port Scanning and Service Identification

1. **Understanding Port Scanning and Connection Issues:**
   When you begin scanning, you first input the **IP address**, **port**, **username**, and **password**. For instance, if you use **FortiClient** for a connection, you need to specify these details:
   - **Host**, **port** (e.g., 443), **username** (e.g., "Ali @ Google"), and **password**.
   - After entering the details correctly, you can initiate the connection. If you enter the wrong **port number**, it will lead to a failed connection. For example, using **port 8** instead of **port 443** would not work because the service is associated with the wrong port.
   - Correctly entering the **port number** is crucial for establishing the right connection.
   - If you use the wrong port, the connection attempt will fail

Port Scanning Basics:

Port scanning is one of the **initial techniques** we use to determine open ports and the services running on them. Think of this as checking the **entry points** to a building.

- The **port scanning** process checks which **ports** are open, allowing access to services.
- The scan results show the **status** of the ports: **open**, **closed**, or **filtered**.
  - **Open** means the port is accessible and can allow traffic.
  - **Closed** means the port is not accepting connections.
  - **Filtered** means the port is being blocked by a firewall or security measure.

Example:
If **port 80** is scanned and marked as **open**, you know a **web server** is likely running on that port. If it's marked **closed**, no service is active. If it's **filtered**, a firewall might be blocking access.

- **Port scanning** helps in identifying which services might be vulnerable to attack by showing open or closed ports.
- A **filtered port** indicates that firewall or network protections are in place.

Understanding Ports

Port State	Description	Analogy
Open	Accessible and accepting connections	Unlocked door
Closed	Not accepting connections	Locked door
Filtered	Blocked by firewall/security measure	Door with security guard

Difference Between TCP and SYN Scans:

A **TCP scan** completes the full handshake (SYN → SYN-ACK → ACK), which involves fully establishing a connection between your system and the target. This method can be easily detected by the server, as the connection is fully logged.

- A **SYN scan**, however, only sends the initial **SYN** packet and waits for a **SYN-ACK** response. This doesn't complete the connection, making it stealthier and harder to detect. It's like testing the door without actually opening it.

Example:

- **TCP Scan:** Sends a SYN, receives a SYN-ACK, then sends an ACK to complete the connection. This leaves logs in the server.
- **SYN Scan:** Sends a SYN, waits for a SYN-ACK, but stops before completing the handshake. It minimizes the risk of detection.
- **SYN scans** are faster and more stealthy than full **TCP scans**.
- Full **TCP scans** give more detailed information but are easier to detect by the server.

UDP Scanning:

UDP ports are different from **TCP** because **UDP** is a connectionless protocol. This means that **UDP scanning** doesn't need to establish a connection before sending data, making it trickier to detect and more useful in scanning **UDP ports**.

Example:

Many services run on **UDP** but are often overlooked in **TCP scans**. For instance, **DNS** typically runs on UDP (port 53), and **VoIP** systems may use UDP for voice traffic.

- **UDP scanning** is useful for discovering vulnerabilities in **connectionless services**.
- It's more difficult to identify open **UDP ports** compared to **TCP ports**, as they don't use handshakes.

Service and Version Identification:

Once you know which ports are open, identifying the **services** running on them is crucial. This process allows you to assess which software or applications are available on the network.

- By using **service version scanning**, you can identify the **version of software** running on each open port (e.g., Apache 2.8, FTP 3.2). This

is important because older versions of services may have known vulnerabilities.

Example:
If **port 80** is open and running **Apache 2.8**, you now know the **version of Apache** running on the web server. This allows you to check for specific vulnerabilities associated with that version.

- Identifying the **version** of a service helps in determining if it has any **known vulnerabilities**.
- **Service version scanning** helps in narrowing down the specific exploits you may need to defend against.

1. **فهم فحص المنافذ ومشاكل الاتصال:**

عندما تبدأ في الفحص، أول شيء تقوم به هو إدخال عنوان IP و المنفذ و اسم المستخدم و كلمة المرور .على سبيل المثال، إذا كنت تستخدم FortiClientللاتصال، تحتاج إلى تحديد هذه التفاصيل:

- **المضيف و المنفذ (مثل 443) و اسم المستخدم (مثل "علي @ جوجل") و كلمة المرور.**
- **بعد إدخال التفاصيل بشكل صحيح، يمكنك بدء الاتصال.**

إذا أدخلت رقم المنفذ خاطئًا، فسينتج عن ذلك فشل في الاتصال. على سبيل المثال، استخدام المنفذ 8 بدلاً من المنفذ 443 لن يعمل لأن الخدمة مرتبطة بالمنفذ الخطأ.

- **إدخال رقم المنفذ بشكل صحيح أمر حاسم لإتمام الاتصال الصحيح.**
- **إذا استخدمت المنفذ الخطأ، ستفشل المحاولة**

2. **أساسيات فحص المنافذ:**

فحص المنافذ هو أحد أول التقنيات التي نستخدمها لتحديد المنافذ المفتوحة والخدمات التي تعمل عليها. فكر في هذا مثل التحقق من نقاط الدخول إلى المبنى.

- **عملية فحص المنافذ تتحقق من المنافذ المفتوحة التي تسمح بالوصول إلى الخدمات.**
- **تظهر نتائج الفحص حالة كل منفذ :مفتوح أو مغلق أو مفلتر.**
  - **مفتوح يعني أن المنفذ نشط ويمكنه قبول الاتصالات.**
  - **مغلق يعني أن المنفذ غير نشط ولا يقبل الاتصالات.**
  - **مفلتر يعني أن المنفذ يتم حظره بواسطة جدار حماية أو إجراء أمني.**

مثال:
إذا تم فحص المنفذ 80 وأظهر "مفتوح"، فأنت تعرف أن خدمة الويب تعمل. إذا أظهر "مغلق"، لا توجد خدمة نشطة. إذا كان "مفلترًا"، قد يقوم جدار الحماية بحظر الوصول.

- فحص المنافذ يساعد في تحديد الخدمات التي قد تكون عرضة للهجوم من خلال إظهار المنافذ المفتوحة أو المغلقة.
- المنافذ المفلترة تشير إلى أن جدار الحماية أو الإجراءات الأمنية قد تكون في مكانها.

3. الفرق بين فحص TCP وفحص:SYN

فحص sT TCP يكمل المصافحة الكاملة(SYN → SYN-ACK → ACK) ، مما ينطوي على إتمام الاتصال بين جهازك والهدف. يمكن اكتشاف هذه الطريقة بسهولة من قبل السيرفر لأن الاتصال يتم تسجيله بالكامل.

- فحص sS SYN ، من ناحية أخرى، يرسل فقط حزمة SYNوينتظر رد SYN-ACK. لكنه لا يكمل الاتصال، مما يجعله أكثر خفاءً وأصعب في الكشف. يشبه الأمر اختبار الباب لمعرفة ما إذا كان مفتوحًا دون دخوله.

مثال:

- فحص TCP: يرسلSYN ، يتلقىSYN-ACK ، ثم يرسل ACK لإتمام الاتصال. يترك هذا أثرًا في سجل السيرفر.
- فحص SYN: يرسل SYN فقط، ويتوقف بعد تلقي SYN-ACK. لا يكمل المصافحة، مما يقلل من فرص الكشف.
- فحص SYN أسرع وأكثر خفاءً من فحص.TCP
- فحص TCP يقدم معلومات أكثر تفصيلًا ولكن يسهل اكتشافه من السيرفر.

4. فحص:UDP

sU UDPهو بروتوكول غير معتمد على الاتصال، مما يجعله مختلفًا عن TCP. هذا يجعل فحص UDP أكثر تعقيدًا لأنه لا يحتاج إلى إتمام الاتصال قبل إرسال البيانات.

مثال:

العديد من الخدمات تعمل على UDPوغالبًا ما يتم تجاهلها في فحص TCP. على سبيل المثال، يعمل DNSعادة على UDP (المنفذ 53)، وقد تستخدم أنظمة VoIP UDPلحركة الصوت.

- فحص UDP مفيد لاكتشاف الثغرات في الخدمات غير المعتمدة على الاتصال.
- من الصعب تحديد المنافذ المفتوحة في UDP مقارنةً بـ TCP، لأنها لا تستخدم المصافحة.

Understanding TCP and SYN Scanning:

When performing **TCP scans**, the process involves completing the full handshake, meaning that it sends a **SYN** packet, receives a **SYN-ACK** from the target, and then sends an **ACK** back to complete the connection. This

leaves logs in the server, which means the server will know that an attempt was made **Problem with TCP Scan:** The connection is **logged** in the server, leaving a trace.

- **Solution: SYN scans** are useful here as they only initiate the handshake (send SYN) and stop at that point, avoiding connection completion and minimizing detection.
- **SYN Scans** are **stealthier** than **TCP scans** because they don't complete the handshake, making them harder to detect.

Host Discovery Commands

Command	Purpose	When to Use
nmap -sn <target>	Ping scan only	Quick host discovery
nmap -Pn <target>	Skip ping, assume host up	When ICMP is blocked
nmap -PB <target>	Standard host discovery	Default behavior

Identifying Open Ports and Services:

When you scan for **open ports**, you're checking whether a server is allowing incoming connections on certain ports. For example, when scanning port 80, you're likely checking if a **web service** is running.

- **Open Port:** The port is accessible and services can be connected to.
- **Closed Port:** The port is not accepting connections.
- **Filtered Port:** The port might be blocked by a **firewall** or security measure.
- **Open ports** reveal active services, while **filtered** ports can indicate that protective mechanisms like firewalls are in place.
- Understanding the status of each port helps in identifying vulnerabilities in the system.

Port Scanning Commands

Command	Description	Example
nmap -p <port>	Scan specific port	nmap -p 80 192.168.1.1
nmap -p <range>	Scan port range	nmap -p 1-1000 192.168.1.1
nmap -p-	Scan all 65535 ports	nmap -p- 192.168.1.1
nmap -F	Fast scan (top 100 ports)	nmap -F 192.168.1.1

Service Version and OS Fingerprinting:

Once you have identified the open ports, the next critical step is to determine the **service** running on each port and its **version**.

- **Service Version sV :** If you find a service running on port 80 (HTTP), you can determine which **version** of the service is running, e.g., Apache 2.8. Knowing the version allows you to check if there are **known vulnerabilities** associated with it.
- **OS Fingerprinting sO :** This helps in identifying the **operating system** of the server. For example, if you detect that the server is running **Linux 2.6**, you can research its vulnerabilities.
- Identifying the **service version** is crucial for **security assessments** and helps in pinpointing vulnerabilities in outdated services.
- **OS fingerprinting** is valuable for knowing which operating system is running, allowing for targeted attacks or defensive measures.

Service and OS Detection

Command	Purpose	Information Gathered
nmap -sV <target>	Service version detection	Service name and version
nmap -sV --version-all <target>	Aggressive version detection	All possible version info
nmap -O <target>	OS fingerprinting	Operating system details
nmap -sSV <target>	Combined service + OS detection	Both service and OS info

UDP Scanning:

UDP scans are used to detect open **UDP ports**, which are commonly overlooked during **TCP scans**. Unlike TCP, UDP doesn't require establishing a connection before sending data.

- For example, services like **DNS** and **VoIP** typically use **UDP**. Scanning for these open **UDP ports** helps in identifying vulnerabilities associated with these services.
- **UDP scanning** is essential for uncovering services running over **UDP**, which are not easily detected by regular TCP scans.
- It is more challenging because **UDP** is a connectionless protocol and doesn't involve a handshake.

Combining Scans for Better Results:

The most effective way to scan a network is to **combine multiple scan techniques**. For example:

- **SYN Scan** for stealthiness.
- **TCP Scan** for thoroughness.
- **UDP Scan** for connectionless services.
- **Combining scans** gives you a more complete picture of the network's health and vulnerabilities.
- This multi-faceted approach ensures you don't miss any potential security risks.

1. فهم فحص TCP و:SYN

عند إجراء فحصTCP ، تشمل العملية إتمام المصافحة بالكامل، مما يعني إرسال حزمة SYN، ثم تلقي SYN-ACKمن الهدف، ثم إرسال ACKلإتمام الاتصال. هذا يترك سجلات في السيرفر، مما يعني أن السيرفر سيعرف أنه تم إجراء محاولة اتصالالمشكلة في فحص :TCP الاتصال يتم تسجيله في السيرفر، مما يترك أثراً.

- **الحل :يمكن استخدام فحص SYN هنا لأنه يبدأ المصافحة فقط) يرسل (SYN ويتوقف في تلك المرحلة، مما يتجنب إتمام الاتصال ويقلل من فرص الكشف.**
- **فحص SYN أكثر خفة من فحص TCP لأنه لا يكمل المصافحة، مما يجعله أصعب في الكشف.**

2. التعرف على المنافذ المفتوحة والخدمات:

عندما تقوم بفحص المنافذ المفتوحة، فإنك تتحقق مما إذا كان السيرفر يقبل الاتصالات الواردة على المنافذ المعينة. على سبيل المثال، عندما تفحص المنفذ 80، فإنك تتحقق مما إذا كانت خدمة الويب تعمل.

- المنفذ المفتوح :المنفذ مفتوح ويمكن للاتصالات الوصول إليه.
- المنفذ المغلق :المنفذ غير نشط ولا يقبل الاتصالات.
- المنفذ المفلتر :قد يتم حظر المنفذ بواسطة جدار حماية أو إجراء أمني.
- المنافذ المفتوحة تكشف عن الخدمات النشطة، بينما قد تشير المنافذ المفلترة إلى أن جدار الحماية أو الإجراءات الأمنية موجودة.
- معرفة حالة كل منفذ يساعد في تحديد الثغرات المحتملة في النظام.

3. إصدار الخدمة و التعرف على نظام التشغيل:

بمجرد أن تحدد المنافذ المفتوحة، فإن الخطوة المهمة التالية هي تحديد الخدمة التي تعمل على كل منفذ وإصدارها.

- إصدار الخدمة :إذا وجدت خدمة تعمل على المنفذ 80(HTTP) ، يمكنك تحديد إصدار الخدمة مثل Apache 2.8. معرفة الإصدار يسمح لك بالتحقق مما إذا كانت هناك ثغرات معروفة مرتبطة به.
- التعرف على نظام التشغيل :يساعد هذا في تحديد نظام التشغيل الخاص بالسيرفر. على سبيل المثال، إذا اكتشفت أن السيرفر يعمل على لينكس 2.6، يمكنك البحث عن الثغرات المعروفة لذلك النظام.

- التعرف على إصدار الخدمة أمر بالغ الأهمية لتقييم المخاطر المتعلقة بالثغرات في البرمجيات المحددة.
- التعرف على نظام التشغيل له قيمة كبيرة في تحديد ما إذا كان هناك ثغرات محتملة في نظام التشغيل.

4. فحص:UDP

يستخدم فحص UDP لاكتشاف المنافذ المفتوحة فيUDP ، والتي غالبًا ما يتم تجاهلها أثناء فحص .TCP على عكسTCP ، UDPلا يتطلب إنشاء اتصال قبل إرسال البيانات.

- على سبيل المثال، غالبًا ما تستخدم خدمات DNS و VoIPبروتوكول .UDPيساعد فحص هذه المنافذ المفتوحة في UDP في تحديد الثغرات المرتبطة بهذه الخدمات.

- فحص UDP أمر أساسي لاكتشاف الخدمات التي تعمل عبر UDPوالتي قد تكون مخفية في الفحوصات التقليدية.
- يصعب اكتشاف المنافذ المفتوحة في UDP مقارنة بـ TCPلأنها لا تستخدم المصافحة.

Best Practices:

The key to an effective security strategy is a holistic approach to scanning and understanding the network's vulnerabilities. By combining SYN scans, TCP scans, and UDP scans, you ensure that you identify all potential vulnerabilities, no matter how obscure. Additionally, service version scanning and OS fingerprinting allow for a more accurate assessment of the target's security.

1. Ping Scan (PB)

Purpose:

A Ping Scan is used to check if a host is reachable. It sends an **ICMP echo request** (ping) and waits for a response. If there's no response, the host is considered down.

Command:

nmap -PB <target IP or range>

- **-PB** skips the port scan and only performs **host discovery** to check if the host is up.

Usage:

- This scan is quick and useful when you only need to know if a target is reachable.

الغرض:

يتم استخدام فحص الـ **Ping**للتحقق من إمكانية الوصول إلى جهاز مضيف. يرسل **طلب صدى ICMP** (ping)وينتظر الرد. إذا لم يتم الحصول على رد، يُعتبر الجهاز المضيف غير متصل.

الاستخدام:

- هذا الفحص سريع ومفيد عندما تحتاج فقط لمعرفة إذا كان الجهاز الهدف قابلًا للوصول.

2. Ping Scan (PN)

Purpose:

A No Ping Scan skips the ICMP ping check and assumes the host is up, proceeding directly to the port scan. This is useful when ICMP (ping) responses are blocked by a firewall or other security mechanisms.

Command:
nmap -Pn <target IP or range>

- **-Pn** skips the ping check and proceeds directly to **port scanning**.

Usage:

- This scan is effective when ping is blocked but you are certain the host is up.

الغرض:
يتجاوز فحص No Pingفحص الـ ICMP (ping) ويفترض أن الجهاز الهدف يعمل، ثم ينتقل مباشرة إلى فحص المنافذ. يكون هذا مفيدًا عندما يتم حظر استجابات الـ ICMP (ping) بواسطة جدار ناري أو آليات أمان أخرى.

الاستخدام:

- **هذا الفحص فعال عندما يكون الـ ping محظورًا ولكنك متأكد من أن الجهاز الهدف يعمل.**

3. TCP Connect Scan (PP)

Purpose:

The **TCP Connect Scan** attempts to establish a full **TCP connection** by completing the **TCP handshake** (SYN, SYN-ACK, ACK). It's the most reliable method but is easily detectable since it fully connects.

Command:
nmap -p <port range> <target IP>

- **-p** specifies the port(s) to scan.

Usage:

- This scan is used for thorough and reliable scanning but leaves a trace in the server logs, making it easier to detect.

الغرض:
محاولة فحص TCP Connect لإتمام اتصال TCP كامل عن طريق استكمال مصافحة TCP (SYN، SYN-ACK، ACK). إنها الطريقة الأكثر موثوقية ولكنها قابلة للكشف بسهولة لأنها تكمل الاتصال بالكامل.

الاستخدام:

- **يتم استخدام هذا الفحص للفحص الشامل والموثوق، لكنه يترك أثرًا في سجلات الخادم، مما يجعل من السهل اكتشافه.**

4. SYN Scan (PM)

Purpose:

A **SYN Scan** sends a **SYN packet** and waits for a **SYN-ACK response** without completing the full handshake. It's faster and stealthier than the TCP connect scan, as it doesn't establish a full connection and leaves fewer traces.

Command:
nmap -sS <target IP>

- **-sS** tells Nmap to perform a **SYN scan**.

Usage:

- This is a **stealthier** and **faster** method, useful when you want to avoid detection.

الغرض:
يُرسل SYN Scan حزمة SYN وينتظر استجابة SYN-ACK دون إتمام المصافحة الكاملة. إنه أسرع وأكثر سرية من فحص الاتصال الكامل TCP ، لأنه لا يُكمل الاتصال بالكامل ويترك آثارًا أقل.

الاستخدام:

- **هذه الطريقة أكثر سرية و أسرع، مفيدة عندما تريد تجنب الكشف.**

5.PE - Ping Scan with Timing

Purpose:

A **Ping Scan** is a host discovery technique where Nmap pings a range of IP addresses to check which hosts are **up** or **alive**. In some cases, you may want to set **specific timing** for your scans (like PE, though it's not standard Nmap shorthand).

Command:
nmap -sn <target IP or range>

- **-sn** skips the port scan and only performs a **ping** check to determine if the host is reachable.

الغرض:
يُعد فحص الـ Ping تقنية لاكتشاف الأجهزة حيث يقوم Nmapبإرسال طلبات ping لعدد من عناوين الـ IPللتحقق من الأجهزة التي هي حية أو مستجيبة .في بعض الحالات، قد ترغب في تعيين توقيتات محددة لفحصك مثلPE ، على الرغم من أنه ليس اختصارًا قياسيًا

sn-يتجاوز فحص المنافذ ويقوم فقط بإجراء فحص الـ ping لتحديد ما إذا كان الجهاز المضيف قابلًا للوصول.

Timing Options and Challenges

The **timing** of a scan is critical in ensuring that it runs efficiently and avoids detection. When you're scanning a network, the speed of your scan can either trigger an alert or help you bypass detection systems.

During **live target scans**, you might face issues such as **delays** in responses, especially when scanning **large networks**. This happens because the more hosts you scan, the more time it takes to process the data. Additionally, **firewalls** and **intrusion detection systems (IDS/IPS)** may be in place to prevent fast scans from completing.

تعتبر التوقيتات **أثناء الفحص أمرًا بالغ الأهمية لضمان تشغيل الفحص بشكل فعال وتجنب الاكتشاف. عند فحص الشبكة، يمكن أن يؤدي** سرعة الفحص **إلى تحفيز تنبيه أو مساعدتك على تجاوز أنظمة الكشف.**

أثناء الفحص للأهداف الحية، **قد تواجه مشكلات مثل** التأخيرات **في الاستجابات، خاصةً عندما تقوم بفحص** شبكات كبيرة **.يحدث ذلك لأنه كلما زادت الأجهزة التي تقوم بفحصها، زاد الوقت الذي يستغرقه معالجة البيانات. بالإضافة إلى ذلك، قد تكون** الجدران النارية **و** أنظمة الكشف عن التسلل (IDS/IPS) **موجودة لمنع اكتمال الفحص السريع.**

2. Scan Timing Adjustments:

Nmap provides several timing options to control the **speed** of the scan:

- **T5 (Maximum Speed):**
  - This option is **extremely fast** but can easily be detected by IDS/IPS systems. It's used when you want to gather information as quickly as possible, but it can lead to **false positives** and a higher risk of detection.
  - **Usage:** Avoid using in environments where stealth is important.
- **T4 (Aggressive):**
  - **Faster** than normal, but still **stable**. It's suitable when you need **speed** but don't want to take the extreme risk of detection. However, it can still be flagged by IDS/IPS systems if the scan is too rapid.
  - **Usage:** Ideal for environments where time is of the essence but some **detection** is acceptable.
- **T3 (Normal):**
  - The **default speed** for scans. It strikes a balance between **speed** and **stealth**, making it less likely to trigger security alarms while still completing the scan within a reasonable time.
  - **Usage:** Recommended for most regular scans where **stealth** is required.
- **T2 (Slow Down):**
  - This reduces the scan speed significantly, making it **less likely** to be detected by security systems, but it can significantly increase the time taken to complete the scan.
  - **Usage:** Useful in **sensitive environments** like **banks** or other critical infrastructure.
- **T1 (Very Slow):**
  - **Extremely slow**, but this makes the scan **more stealthy**. **IDS/IPS systems** are much less likely to notice or flag such scans.
  - **Usage:** Perfect when **maximum stealth** is necessary but don't mind waiting for the scan to finish.
- **T0 (Stealth Mode):**
  - The **slowest option** for maximum stealth. This ensures that even highly sensitive environments with advanced detection methods will have difficulty detecting the scan.
  - **Usage:** Best for when **complete stealth** is necessary.
  -

Template	Name	Speed	Stealth	Use Case
T0	Paranoid	Extremely Slow	Maximum	Maximum stealth, sensitive environments
T1	Sneaky	Very Slow	Very High	When detection must be avoided
T2	Polite	Slow	High	Sensitive environments like banks
T3	Normal	Medium	Medium	Default, balanced scanning
T4	Aggressive	Fast	Low	When speed matters, some detection acceptable
T5	Insane	Maximum	Very Low	Quick info gathering, high detection risk

Nmapيوفر العديد من خيارات التوقيت للتحكم في سرعة الفحص:

T5 (أقصى سرعة)

- **هذه الخيار سريع للغاية ولكن يمكن اكتشافه بسهولة من قبل أنظمة IDS/IPS. يتم استخدامه عندما ترغب في جمع المعلومات بأسرع ما يمكن، ولكن قد يؤدي إلى إيجابيات خاطئة وزيادة خطر الاكتشاف.**
- **الاستخدام :تجنب استخدامه في البيئات التي تعتبر فيها السرية مهمة.**

T4 (عدواني)

- **أسرع من الوضع الطبيعي ولكن لا يزال مستقرًا .مناسب عندما تحتاج إلى السرعة ولكن لا ترغب في تحمل خطر الاكتشاف الشديد. ومع ذلك، يمكن أن يتم الإشارة إليه بواسطة أنظمة IDS/IPS إذا كان الفحص سريعًا جدًا.**
- **الاستخدام :مثالي في البيئات التي يكون فيها الوقت مهمًا ولكن يمكن قبول بعض الاكتشاف.**

T3 (عادي)

- **السرعة افتراضية للفحوصات. يوازن بين السرعة و السرية، مما يجعله أقل عرضة لتنشيط التنبيهات الأمنية بينما يكتمل الفحص في وقت معقول.**
- **الاستخدام :موصى به لمعظم الفحوصات العادية حيث تكون السرية مطلوبة.**

T2 (تباطؤ)

- **يقلل هذا الخيار من سرعة الفحص بشكل كبير، مما يجعله أقل عرضة للكشف بواسطة الأنظمة الأمنية، ولكن قد يزيد بشكل كبير من الوقت المستغرق لإكمال الفحص.**
- **الاستخدام :مفيد في البيئات الحساسة مثل البنوك أو البنية التحتية الحرجة.**

T1 (بطيء جدًا)

- **بطيء للغاية، ولكن هذا يجعل الفحص أكثر سرية .من غير المحتمل أن تلاحظ أنظمة IDS/IPSأو ترفع علمًا بشأن هذا الفحص.**
- **الاستخدام :مثالي عندما يكون السرية القصوى ضرورية ولكن لا تمانع في انتظار اكتمال الفحص.**

T0 (وضع السرية)

- **أبطأ خيار لتحقيق أقصى درجات السرية. يضمن هذا أنه حتى في البيئات الحساسة للغاية ذات أساليب الكشف المتقدمة، سيكون من الصعب اكتشاف الفحص.**
- **الاستخدام :الأفضل عندما تكون السرية الكاملة ضرورية.**

3. Practical Use of Timing Options:

When conducting scans, it's crucial to adjust the **speed** depending on the situation. Here's an example:

- If you're scanning a **large network** (e.g., a **500 IP range**), you must manage the **bandwidth usage** to avoid network disruptions. If you use too much bandwidth or conduct the scan too quickly, the **IDS/IPS** might detect the scan and block it.
- In these cases, you might use the **minimum and maximum host grouping** (--min-hostgroup and --max-hostgroup) to control the number of hosts scanned simultaneously, which also helps in **avoiding detection**.

عند إجراء الفحوصات، من الضروري ضبط السرعة **وفقًا للموقف. إليك مثالاً:**

- **إذا كنت تقوم بفحص** شبكة كبيرة)**مثل** نطاق 500 عنوان(IP ، **يجب عليك إدارة** استخدام النطاق الترددي **لتجنب تعطل الشبكة. إذا استخدمت الكثير من النطاق الترددي أو أجريت الفحص بسرعة كبيرة، قد تقوم** أنظمة IDS/IPS **بالكشف عن الفحص ووقفه.**
- **في هذه الحالات، يمكنك استخدام** تجميع الأجهزة في مجموعات **min-hostgroup--)و -max--** **hostgroup)للتحكم في عدد الأجهزة التي يتم فحصها في وقت واحد، مما يساعد أيضًا في** تجنب الكشف.

4. Combination of Techniques for Effective Scanning:

You can combine **timing options** with other techniques to improve the stealth and efficiency of your scans:

- **Stealthy Scan (Stealth Mode - T0/T1)**
  - This is useful when you need to scan sensitive targets (e.g., **corporate networks** or systems with **high security**).
  - You would **combine the slow scan (T0 or T1)** with methods like **SYN scan** (-sS), which is stealthier than a **full TCP connection** scan.
- **Timing with Specific Ports:**
  - Sometimes you need to scan specific ports such as **22 (SSH)**, **80 (HTTP)**, or **443 (HTTPS)**, especially when targeting services known to be running on those ports.
  - Example:
- nmap -p 22,80,443 -T3 <target IP>
  - This helps **focus on critical ports** and ensure that the scan is **done efficiently**.

الفحص الخفي) وضع السرية(T0/T1 -

- **هذا مفيد عندما تحتاج إلى فحص أهداف حساسة (مثل الشبكات المؤسسية أو الأنظمة ذات الأمان العالي.(**

يمكنك دمج الفحص البطيء T0 أو (T1 مع طرق مثل فحص(-sS) SYN ، الذي هو أكثر سرية من الفحص الكامل لاتصال.TCP

- **التوقيت مع المنافذ المحددة:**
  - **في بعض الأحيان تحتاج إلى فحص منافذ معينة مثل (SSH) 22و (HTTP) 80و (HTTPS) 443، خاصة عند استهداف الخدمات المعروفة بأنها تعمل على هذه المنافذ.**
  - **مثال:**

nmap -p 22,80,443 -T3 <target IP>

- **يساعد هذا في التركيز على المنافذ الحيوية وضمان إجراء الفحص بكفاءة.**

1. Time and Rate Control for Nmap Scans:

During a penetration test, you might be tasked with scanning large networks with numerous hosts. The challenge is ensuring that the scan doesn't overwhelm the network or trigger security defenses like Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS). To do so, we rely on **rate limiting** and **timing adjustments**.

- **Minimum and Maximum Host Groups:**
  The flags --min-hostgroup and --max-hostgroup allow you to control how

many hosts Nmap should scan in parallel. For large networks, especially when dealing with hundreds of hosts, you might want to balance between scanning efficiency and network bandwidth usage.

Command Example:

nmap --min-hostgroup 10 --max-hostgroup 50 <target>

Use Case:
If you are scanning a network with 500 IPs, using a small host group size (e.g., 50 hosts per group) helps avoid overwhelming the network's bandwidth and reduces the likelihood of triggering IDS/IPS due to excessive traffic.

1.التحكم في الوقت والمعدل لفحوصات:Nmap

أثناء اختبار الاختراق، قد يُطلب منك مسح شبكات كبيرة بها العديد من الأجهزة. التحدي هو التأكد من أن الفحص لا يطغى على الشبكة أو يُنبه أنظمة الأمان مثل أنظمة الكشف (IDS) أو أنظمة الحماية .(IPS) للقيام بذلك، نعتمد على تحديد المعدل و تعديل الوقت.

- **أقل وأكبر مجموعات الأجهزة:**

الفلاج --min-hostgroup و--max-hostgroup يسمحان لك بالتحكم في عدد الأجهزة التي يجب فحصها بالتوازي. بالنسبة للشبكات الكبيرة، خاصة عند التعامل مع مئات الأجهزة، قد ترغب في تحقيق التوازن بين الكفاءة في الفحص واستخدام عرض النطاق الترددي للشبكة.

مثال على الأمر:

nmap --min-hostgroup 10 --max-hostgroup 50 <target>

حالة الاستخدام:
إذا كنت تقوم بفحص شبكة بها 500 عنوانIP ، فإن استخدام حجم مجموعة أجهزة صغير (مثل 50 جهازًا لكل مجموعة) يساعد في تجنب إرهاق عرض النطاق الترددي للشبكة ويقلل من احتمال تشغيل أنظمة IDS/IPS بسبب الحركة الزائدة.

2. Rate Limiting with Nmap:

By controlling the rate at which probes are sent, Nmap can be configured to make the scan more subtle, especially if security devices are likely to flag fast scans. The --min-rate and --max-rate flags allow you to control the rate at which packets are sent.

Command Example:

nmap --min-rate 100 --max-rate 500 <target>

Use Case:
If you're scanning a business network and want to minimize the impact of the scan on operational traffic, controlling the packet rate is essential. For instance, during peak hours, setting a slower rate can ensure the scan does not disrupt the network and avoid detection by security systems.

2.تحديد المعدل مع Nmap:

من خلال التحكم في المعدل الذي يتم فيه إرسال الاستعلامات، يمكن تكوين Nmap لجعل الفحص أكثر دقة، خاصة إذا كانت أجهزة الأمان من المحتمل أن تحدد الفحوصات السريعة. تسمح لك الفلاق --min-rate و--max-rate بالتحكم في المعدل الذي يتم فيه إرسال الحزم.

مثال على الأمر:

nmap --min-rate 100 --max-rate 500 <target>

حالة الاستخدام:
إذا كنت تقوم بفحص شبكة تجارية وترغب في تقليل تأثير الفحص على حركة المرور التشغيلية، فإن التحكم في معدل الحزم أمر ضروري. على سبيل المثال، أثناء ساعات الذروة، يمكن أن يضمن ضبط المعدل بشكل أبطأ أن الفحص لا يعطل الشبكة ويتجنب الاكتشاف من قبل أنظمة الأمان.

3. Bypassing IDS/IPS with Delayed and Stealthy Scans:

Security systems like IDS/IPS often flag rapid scans, especially when multiple probes are sent in quick succession. By manipulating the **scan delay** and **max-scan-delay**, we can avoid triggering these defenses.

- **Delay Scans:**
  By using --scan-delay and --max-scan-delay, we insert a delay between probe requests, making it appear like normal network traffic instead of an aggressive scan.

Command Example:

nmap --scan-delay 500ms --max-scan-delay 2s <target>

Use Case:
For example, if you're conducting a scan against a network protected by IDS/IPS, increasing the delay between probes can prevent the scan from being detected by these systems.

3.تجاوز IDS/IPS مع الفحوصات المؤجلة والسرية:

غالبًا ما تميز أنظمة الأمان مثل IDS/IPS الفحوصات السريعة، خاصة عندما يتم إرسال العديد من الاستعلامات بسرعة. من خلال تعديل تأخير الفحص و أقصى تأخير للفحص، يمكننا تجنب تحفيز هذه الدفاعات.

- **تأخير الفحوصات:**
باستخدام --scan-delay و--max-scan-delay، يمكننا إدخال تأخير بين استعلامات الفحص، مما يجعلها تبدو مثل حركة مرور الشبكة العادية بدلاً من الفحص العدواني.

مثال على الأمر:

nmap --scan-delay 500ms --max-scan-delay 2s <target>

حالة الاستخدام:
على سبيل المثال، إذا كنت تقوم بإجراء فحص ضد شبكة محمية بأجهزةIDS/IPS ، فإن زيادة التأخير بين الاستعلامات يمكن أن تمنع الفحص من أن يُكتشف بواسطة هذه الأنظمة.

4. Advanced Techniques for Evasion:

For more advanced evasion, techniques like **paralellism adjustments** (--min-parallelism, --max-parallelism) and **max retries** (--max-retries) are used to adjust how aggressively Nmap scans. Lowering parallelism or limiting retries can reduce the scan's detectability.

Command Example:

nmap --max-retries 3 --min-parallelism 5 --max-parallelism 20 <target>

Use Case:
By reducing retries and controlling parallelism, the scan behaves more like legitimate traffic, helping to avoid detection from network security devices.

4.تقنيات متقدمة للتخفي:

لاستخدام تقنيات أكثر تقدمًا في التخفي، نستخدم مثل تعديلات التوازي (--min-parallelism, --max-parallelism)و أقصى محاولات إعادة الفحص (--max-retries)لضبط كيفية مسح Nmap بشكل عدواني. يؤدي تقليل التوازي أو تحديد المحاولات إلى تقليل قابلية الاكتشاف للفحص.

مثال على الأمر:

nmap --max-retries 3 --min-parallelism 5 --max-parallelism 20 <target>

حالة الاستخدام:
من خلال تقليل المحاولات والتحكم في التوازي، يتصرف الفحص بشكل أكثر مشابهة لحركة المرور المشروعة، مما يساعد في تجنب اكتشاف أجهزة الأمان.

These concepts and commands help optimize the scan process while maintaining a level of stealth to avoid detection from network security systems. Make sure to use the appropriate flags for each scenario to balance between scan speed, stealth, and accuracy.

Function	Flags (English)	Explanation (English)	Explanation (Arabic)
Size of the group of hosts to be scanned concurrently	--min-hostgroup, --max-hostgroup	These flags control the minimum and maximum number of hosts that will be scanned concurrently.	هذه الفلات تتحكم في الحد الأدنى والحد الأقصى لعدد الأجهزة التي سيتم مسحها في نفس الوقت.
Number of scanning probes to be launched in parallel	--min-parallelism, --max-parallelism	These flags define the minimum and maximum number of probes (requests) sent out in parallel. More probes increase the speed but also the likelihood of detection.	هذه الفلات تحدد الحد الأدنى والحد الأقصى لعدد الاستفسارات (الطلبات) المرسلة في وقت واحد. المزيد من الاستفسارات يزيد من السرعة لكن يزيد من احتمالية الكشف.

Timeout values for probes	--min-rtt-timeout, --max-rtt-timeout, --initial-rtt-timeout	These flags control the minimum and maximum timeout values for the round-trip time (RTT) of probes. It determines how long to wait for a response before timing out.	هذه الفلات تتحكم في القيم الدنيا والعليا لوقت المهلة (الوقت المستغرق للذهاب والعودة) للاستفسارات. يحدد هذا الوقت الذي يجب الانتظار فيه قبل أن يتم اعتبار الاستفسار غير مجاب عليه.
Maximum number of probe retransmissions allowed	--max-retries	This flag sets the maximum number of times a probe will be retried if no response is received. More retries increase the scan's thoroughness but also its duration.	هذا الفلاج يحدد الحد الأقصى لعدد مرات إعادة إرسال الاستفسارات في حال عدم الحصول على رد. المزيد من المحاولات تزيد من دقة الفحص ولكنها تطيل الوقت.
Maximum time before giving up on an entire host	--host-timeout	This flag sets the timeout value for scanning a single host. If the scan does not finish in this time, it will be considered failed.	هذا الفلاج يحدد الوقت الأقصى للفحص على جهاز واحد. إذا لم يتم الفحص في هذا الوقت، سيتم اعتباره فشل.
Control the delay inserted between each probe against an individual host	--scan-delay, --max-scan-delay	These flags control the delay between each probe sent to a single host. This can help avoid detection by slowing down the scan.	هذه الفلات تتحكم في التأخير بين كل استفسار والاستفسار الآخر المرسل إلى جهاز واحد. يمكن أن تساعد في تجنب الكشف عن طريق تبطيء الفحص.
Rate of probe packets sent per second	--min-rate, --max-rate	These flags define the minimum and maximum rate at which probe packets are sent to the target. A higher rate can speed up the scan but might cause network congestion or detection.	هذه الفلات تحدد الحد الأدنى والحد الأقصى لمعدل إرسال الحزم الاستفسارية في الثانية إلى الهدف. معدل أعلى يمكن أن يسرع الفحص لكنه قد يسبب ازدحام في الشبكة أو الكشف.

Defeat RST packet response rate by target hosts	--defeat-rst-ratelimit	This flag is used to bypass the rate-limiting mechanism for RST (Reset) responses from the target. It can be helpful in scenarios where firewalls limit the rate of RST responses.	هذا الفلاج يستخدم لتجاوز آلية تحديد معدل (إعادة RST الردود تعيين) من الأجهزة المستهدفة. يمكن أن يكون مفيداً في السيناريوهات التي تقوم فيها الجدران النارية بتحديد معدل الردود RST.

Outputs in Nmap

Viewing the output in real-time can be useful, but parsing the information afterward and feeding it into other tools is 10 times more useful. Enter the different output options from Nmap, saving to a file of one sort or another.

Here are a few options to output, but mainly these are xml, .gnmap, and .nmap, and they have the flags: -oX, -oG, -oN. There's also an Easter egg output in 1337 speak, which is -oS.

Flag	Format	Use Case
-oN <file>	Normal output	Human-readable, real-time format
-oX <file>	XML output	Machine parsing, tool integration
-oG <file>	Greppable output	Searching with grep
-oS <file>	"1337" output	Fun/easter egg format
-oA <basename>	All formats (normal, XML, greppable)	Comprehensive saving

1. XML Output (-oX)

The -oX option instructs Nmap to give the output in XML format for parsing later. This is useful for saving scan results in a structured format that can be easily processed by other tools or automated systems.

Command Example:

nmap 10.0.0.1 -oX outputfile

خيار oX-يوجه Nmap لتقديم النتائج بتنسيقXML ، وهو مفيد لحفظ نتائج الفحص بتنسيق منظم يمكن معالجته بسهولة بواسطة أدوات أخرى أو أنظمة مؤتمتة.

2. Greppable Output (-oG)

The -oG option saves the results in a greppable format, allowing you to easily search and extract specific data using grep. For example, you can search for "HTTPS" in the results with the following command:

Command Example:

nmap 10.0.0.1 -oG outputfile
grep HTTPS outputfile

خيار oG-يحفظ النتائج بتنسيق قابل للبحث باستخدام أدوات مثل grep. على سبيل المثال، يمكن استخدام هذا الأمر للبحث عن "HTTPS" في النتائج:

3. Normal Output (-oN)

The -oN option produces output in the same format as when viewed in real-time. It's useful for quickly identifying open ports or getting the bigger picture of a target. This output is human-readable and can be easily interpreted.

Command Example:

nmap 10.0.0.1 -oN outputfile

خيار oN-ينتج النتائج بنفس التنسيق الذي يظهر عند عرضها في الوقت الفعلي. هذا مفيد لاكتشاف المنافذ المفتوحة بسرعة أو للحصول على الصورة الأكبر للهدف. هذا التنسيق سهل القراءة ويمكن فهمه بسهولة.

4. All Output Formats (-oA)

The -oA option is used to save the scan results in **three different formats**: .nmap, .gnmap, and .xml. It's a very convenient way to save the scan output in multiple

formats at once, so you can use whichever format is most appropriate for your analysis later.

Command Example:

nmap 10.0.0.1 -oA outputfile

This command will create:

- **outputfile.nmap** – Normal output.
- **outputfile.gnmap** – Greppable output.
- **outputfile.xml** – XML output.

خيار oA-يُستخدم لحفظ نتائج الفحص في **ثلاثة تنسيقات مختلفة**nmap. : .gnmap، و .xml.هذه طريقة مريحة لحفظ نتائج الفحص في عدة تنسيقات مرة واحدة، بحيث يمكنك استخدام أي تنسيق يكون الأنسب لتحليل البيانات لاحقًا.

1. **Command Breakdown:**

   Command:

   nmap -sSV -p- --min-parallelism 64 --min-hostgroup 16 --max-hostgroup 64 --max-retries 3 -Pn -n -iL input_hosts.txt -oA output --version-all --reason

   Explanation:

- **nmap:** The command to start the Nmap tool.
- **-sSV:** This tells Nmap to use service version detection, providing both the service name and version number.
- **-p-:** This specifies that Nmap should scan all 65535 ports.
- **--min-parallelism 64:** This option specifies the minimum number of parallel operations Nmap will use during the scan. A higher value means more simultaneous scans.
- **--min-hostgroup 16:** This option defines the minimum number of hosts to scan in a group, reducing network congestion by not overloading the network.
- **--max-hostgroup 64:** This sets the maximum number of hosts per group, balancing speed and resource consumption.

- **--max-retries 3:** The number of retries Nmap will attempt for unreachable hosts or ports.
- **-Pn:** Skips host discovery, assuming the host is up. Useful when you know the hosts are active and do not need ping tests.
- **-n:** Disables DNS resolution, preventing Nmap from resolving IPs to hostnames.
- **-iL** input_hosts.txt: This tells Nmap to take the list of IPs or hosts from the specified file input_hosts.txt.
- **-oA** output: Nmap will output the scan results in three formats: XML, Nmap, and greppable formats, saved as output.
- **--version-all:** Instructs Nmap to detect all possible service versions during the scan.
- **--reason:** Displays the reason why Nmap considered a port open or closed.

2. **Detailed Explanation:**
    - *Service Version Detection (**-sSV**):*
      Nmap tries to determine the version of the services running on the open ports. This helps in identifying vulnerabilities tied to specific software versions.
    - *Port Scan (**-p-**):*
      By scanning all ports, Nmap ensures no potential entry point is overlooked, which is essential in vulnerability assessments.
    - *Parallelism and Hostgroup Options:*
      Adjusting the number of simultaneous scans and grouping hosts helps optimize scan performance, reducing time taken by balancing speed with resource use.
    - *Retries (**--max-retries 3**):*
      If a port or host does not respond, Nmap will retry up to three times before marking it as unreachable. This ensures thorough testing under varying network conditions.
    - *Skipping Host Discovery (**-Pn**):*
      When the network is known to be secure and active, skipping host discovery can speed up the scan by directly scanning the provided hosts.
3. **How to Interpret Results:**
    - After execution, you'll receive an output file with detailed information about the open ports, services, and their versions.

- The output will also indicate if a port is open or closed, with the reason provided for each determination (whether it was based on a direct response or a heuristic).

4. **Best Practices:**
   - Use -oA to save the output in multiple formats for easier analysis and reporting.
   - Tuning parallelism and retries helps you balance speed and network load.
   - Using -Pn is suitable when scanning a known list of active hosts without needing to ping them first.

Nmap Command Analysis

Command:

sudo nmap -sSV --version-all -p 21,22 --min-parallelism 64 --script=vuln {} -Pn -n

Explanation:

1. ***sudo nmap:***
   This runs Nmap with elevated privileges, which might be necessary to access certain parts of the network or use certain scanning techniques.
2. ***-sSV:***
   This option enables both service version detection (-sV) and attempts to identify the version of the services running on the open ports.
3. ***--version-all:***
   This option forces Nmap to detect all possible versions of services, ensuring you get the most detailed information about the services running on the target.
4. ***-p 21,22:***
   This specifies that Nmap should only scan ports 21 (FTP) and 22 (SSH). These are common ports for services such as FTP servers and SSH servers.
5. ***--min-parallelism 64:***
   This option adjusts the minimum level of parallelism (number of simultaneous operations) Nmap will use during the scan. The higher the number, the faster the scan, but with more load on the network.
6. ***--script=vuln:***
   This tells Nmap to use the "vuln" script category, which runs vulnerability

detection scripts against the target. It helps identify common vulnerabilities in the services that are found.

7. *{}:*
   This placeholder typically represents the target IP address or list of target IPs. It would need to be replaced with the actual IP address or the file containing a list of target IPs.
8. ***-Pn**:*
   This option skips host discovery. Nmap will assume that the hosts are up and go directly into scanning them. It is useful when you know the target is active but don't want to waste time pinging it.
9. *-n:*
   This option tells Nmap to skip DNS resolution. It will work with IP addresses directly and will not attempt to resolve hostnames, saving time.

1. **Nmap Scripting Engine (NSE)**:
   - **NSE** is a framework within **Nmap** that allows you to run custom scripts written in the **Lua** programming language.
   - Nmap, like Metasploit, has scanning and vulnerability detection capabilities but **NSE** is not a replacement for Metasploit; it is simply an additional feature in Nmap to extend its scanning abilities.
   - Lua is a lightweight and fast programming language that is used for scripting in Nmap.
2. **Key Flags for Nmap Scripts**:
   - **-sc**:
     This flag tells Nmap to perform a **script scan** using the default set of scripts. It's essentially the same as using --script=default.
   - **-script=vuln**:
     This flag will run a **select set of scripts** that look for **vulnerable software** on the target system. These scripts help in identifying known vulnerabilities in the services running on the target. It's important to **review** the scripts to ensure they won't cause any unwanted disruptions or crashes on the target.
   - **-script=safe**:
     This flag instructs Nmap to only run **safe scripts** against the target. These are scripts that are intended for information gathering and **do not exploit** vulnerabilities but rather collect information about the target. This ensures no harm or disruption is caused to the target system.

3. **Additional Information**:
    - NSE provides a variety of scripts, and Nmap users can access a **full list of available scripts** on the Nmap official site.

Nmap Scripting Engine (NSE) and Vulnerability Scanning Explanation

1. **NSE and Vulnerability Detection**:
    - Nmap's **Scripting Engine (NSE)** is not a replacement for **Metasploit** but is used for vulnerability scanning. It allows the execution of scripts that search for vulnerabilities in services.
    - The ***-script=vuln*** flag runs a select set of scripts specifically looking for vulnerabilities in the target's software.
    - **Metasploit** and **NSE** are often used together, with Metasploit focusing on exploiting discovered vulnerabilities, and Nmap focusing on detecting those vulnerabilities.
2. **How NSE Scripts Work**:
    - ***-script=safe:*** This flag ensures that Nmap only runs **safe** scripts that will not exploit vulnerabilities but will gather information about the system.
    - The scripts can detect versions, configurations, and vulnerabilities in services such as FTP, SSH, and more.
3. **Use of Scripts**:
    - The ***-script*** option is followed by the **script name** (e.g., vuln, safe, or others) to specify which type of scan to run.
    - Nmap can **gather detailed service information**, such as the version of FTP or SSH servers, and even provide login details if the scripts are designed to detect such information.
4. **Vulnerability Assessment and Exploitation**:
    - After scanning, **NSE** provides detailed information about vulnerabilities (e.g., CVE identifiers) and associated services.
    - The scripts used can identify weak points such as open ports, exploitable services, and outdated software versions.

Key Flags:

- ***-sSV***: Enables service version detection.
- ***-script=vuln:*** Runs vulnerability detection scripts.
- ***-script=safe:*** Runs only safe scripts for information gathering.
- ***-6:*** Used for IPv6 scanning.
- ***-T4:*** Speeds up the scan while balancing accuracy.

Understanding Daemons and Services:

A daemon refers to a background service or process running on a system. For example, the FTP daemon or the SSH daemon. Each daemon listens on a specific port, such as SSH running on port 22 or FTP on port 21.

These services are often referred to as "daemons" or "services" because they are always listening for incoming connections. The corresponding port (e.g., SSH on port 22) is open and available for use by clients.

الديمون هو عملية أو خدمة تعمل في الخلفية على النظام. على سبيل المثال، ديمون FTP أو ديمون SSH.كل ديمون يستمع على منفذ معين، مثل SSH الذي يعمل على المنفذ 22 أو FTP على المنفذ 21.

يتم الإشارة إلى هذه الخدمات بـ "الديمونات" أو "الخدمات" لأنها تظل دائمًا في حالة الاستماع للاتصالات الواردة. المنفذ المقابل) مثل SSH على المنفذ 22 (مفتوح ومتاحة للاستخدام من قبل العملاء.

Safe Script Execution (-script=safe):

Safe scripts are Nmap scripts that gather information without exploiting vulnerabilities. These scripts are used to collect data about the target system, such as service versions and configurations, but they do not cause harm.

السكربتات الآمنة هي سكربتات Nmap التي تجمع المعلومات دون استغلال الثغرات. تُستخدم هذه السكربتات لجمع البيانات حول النظام المستهدف، مثل إصدارات الخدمات والتكوينات، ولكنها لا تسبب أي ضرر.

Evasion and Filtering with Nmap:

Evasion techniques are essential to bypass firewalls, Intrusion Detection Systems (IDS), and Intrusion Prevention Systems (IPS). One technique is ack scanning (-sA), which uses ACK packets instead of regular TCP packets to avoid detection by firewalls.

Firewall evasion with fragmentation involves splitting packets into smaller pieces, making them harder for firewalls or IDS systems to detect. This technique can be enabled using the -f flag in Nmap.

تقنيات التهرب أمر بالغ الأهمية لتجاوز الجدران النارية، أنظمة كشف التسلل (IDS) و أنظمة منع التسلل (IPS). إحدى التقنيات هي الفحص باستخدام ACK (-sA) ، الذي يستخدم حزم ACK بدلاً من حزم TCP العادية لتجنب اكتشاف الجدران النارية.

تجاوز الجدار الناري باستخدام التجزئة يتضمن تقسيم الحزم إلى قطع أصغر، مما يجعل من الصعب على الجدران النارية أو أنظمة IDS اكتشافها. يمكن تمكين هذه التقنية باستخدام العلم f- في Nmap.

Fragmentation and MTU (Maximum Transmission Unit) Manipulation:

Fragmentation divides the payload into smaller segments to evade firewalls and IDS systems. By splitting packets, firewalls find it harder to detect the attack.

MTU manipulation involves adjusting the size of the packets being sent, and this is particularly useful for evading network detection systems. The ***--mtu*** flag allows users to modify the MTU size.

التجزئة تقسم الحمولة إلى أجزاء أصغر لتجاوز الجدران النارية وأنظمة IDS. من خلال تقسيم الحزم، تجد الجدران النارية صعوبة في اكتشاف الهجوم.

تلاعب MTU يتضمن تعديل حجم الحزم المرسلة، وهذه التقنية مفيدة بشكل خاص لتجاوز أنظمة الكشف الشبكي. يتيح العلم mtu-- للمستخدمين تعديل حجم MTU.

MTU and its Role in Evasion:

MTU (Maximum Transmission Unit) defines the largest size of data packets that can be sent over a network. By adjusting the MTU, you can manipulate the way data is transmitted, which can help avoid detection.

When bypassing firewalls, you may change the MTU to evade detection by splitting large data packets into smaller ones.

MTU(أقصى وحدة نقل) يحدد أكبر حجم للبيانات التي يمكن إرسالها عبر الشبكة. من خلال تعديل MTU، يمكنك التلاعب بطريقة نقل البيانات، مما يساعد على تجنب الكشف.

عند تجاوز الجدران النارية، قد تقوم بتغيير MTU لتجاوز الكشف عن طريق تقسيم الحزم الكبيرة إلى أصغر.

Script Usage for Vulnerability Scanning *(-script=vuln):*

The -script=vuln option in Nmap runs specific vulnerability detection scripts to identify weaknesses and provide detailed information about exploitable services.

These scripts are useful for identifying open services with vulnerabilities, such as FTP servers, SSH servers, and other commonly used services.

خيار -script=vuln في Nmap يقوم بتشغيل سكربتات اكتشاف الثغرات المحددة لتحديد نقاط الضعف وتوفير معلومات مفصلة حول الخدمات القابلة للاستغلال.

هذه السكربتات مفيدة لتحديد الخدمات المفتوحة التي تحتوي على ثغرات، مثل خوادم FTP و SSH والخدمات الأخرى الشائعة الاستخدام.