Decoy Technique (ديكوين)

Its concept lies in sending multiple requests from different sources to confuse security filters. For example, if you have a filter on a firewall or any other security device that blocks network traffic, this technique helps in hiding the actual source of the connection.

فكرته تكمن في إرسال طلبات متعددة من مصادر متعددة لخداع الفلاتر الأمنية. مثلا، إذا كان لديك فلتر على الجدار الناري أو أي جهاز أمني آخر يمنع حركة مرور الشبكة، فهذه التقنية تساعدك على إخفاء مصدر الاتصال الفعلي.

In this technique, we send packets from multiple IP addresses so that the system cannot identify the true source. This is used when there is a risk that an IP source may be blocked due to sending many requests in a short time.

في هذه التقنية، نقوم بإرسال حزم من عدة عناوين IP بحيث لا يتمكن النظام من تحديد المصدر الحقيقي. يُستخدم هذا عندما يكون هناك خطر من أن يتم حظر مصدر IP بسبب إرسال العديد من الطلبات في وقت قصير.

For example, we use a flag in commands such as:

nmap -sP --decoy 10 [target]

Here, the number "10" represents the number of decoys (i.e., the number of addresses that will be used to mask the source). The goal is to make security filters lose track among these multiple packets.

هنا، الرقم "10" يمثل عدد الديكويز (أي عدد العناوين التي ستُستخدم لتخفي المصدر). الهدف من ذلك هو جعل الفلاتر الأمنية تضيع بين هذه الحزم المتعددة.

Source Port Technique (سورس بورت)

Now, we move on to the "Source Port" technique. Here, the idea is to use "trusted ports" to bypass firewalls. As we know, some firewalls may be configured to allow traffic through certain ports like port 80 (HTTP) or port 443 (HTTPS).

For example, a firewall may be configured to accept network traffic only through port 80 or 443, as these ports are typically reserved for web traffic.

However, in some cases, the attacker might exploit this configuration and use port 80 to perform scans or other attacks without being easily detected by the firewall.

An example of this would be performing a scan through port 80 using a command like:

nmap -p 80 [target]

ننتقل الآن إلى "سورس بورت". هنا، الفكرة هي استخدام "بورتات" موثوقة لتجاوز الجدران النارية. كما نعلم، بعض الجدران النارية قد تكون معدة للسماح بحركة المرور من خلال بورتات معينة مثل بورت 80 (HTTP)أو بورت 443.(HTTPS)

على سبيل المثال، يمكن لجدار ناري أن يكون معدًا لقبول مرور الشبكة فقط عبر بورت 80 أو 443، لأن هذه البورتات عادة ما تكون مخصصة للترافيك الخاص بالويب.

ولكن في بعض الحالات، قد يستغل المهاجم هذا الإعداد ويستخدم بورت 80 لتنفيذ عمليات مسح (Scanning)أو هجوم آخر دون أن يتم اكتشافه بسهولة من قبل الفايرول.

Evasion and Detection Techniques

In modern networks, tools like IDS/IPS are used to detect suspicious activities, but by manipulating data size or packet order, these systems can be bypassed. We also use "randomized data" to change the size of packets, making it difficult for inspection systems to detect them.

One way to manipulate data is by changing the "length" of the data being sent, for example, by using the nmap tool to alter packet sizes.

For example, when manipulating the data size, inspections do not only check the data itself, but also the size. So, if you change the data size randomly, the firewall or IDS will not be able to detect it easily.

في الشبكات الحديثة، تستخدم أدوات مثل IDS/IPS للكشف عن النشاطات المشبوهة، ولكن عند التلاعب بحجم البيانات أو ترتيب الحزم، يمكن تجاوز هذه الأنظمة. نستخدم أيضًا "randomized data" لتغيير حجم الحزم بحيث يصعب على أنظمة الفحص التعرف عليها.

مثلاً، عند التلاعب بحجم البيانات، لا تقتصر الفحوصات على فحص البيانات فقط، بل يشمل أيضًا فحص الحجم. لذلك، إذا قمت بتغيير حجم البيانات بشكل عشوائي، لن يستطيع الفايرول أو IDS اكتشاف ذلك بسهولة.

إحدى طرق التلاعب بالبيانات هي من خلال تغيير "الـ length" الخاص بالبيانات التي يتم إرسالها، على سبيل المثال من خلال استخدام أداة nmap لتغيير حجم الحزم.

Data Manipulation and Evasion

When discussing data manipulation, we can use techniques such as "Data Length Manipulation" or "Changing Data Size" to bypass inspection. The idea here is that modern firewalls and inspection tools do not just inspect the data, but also the size of the data.

In this type of attack, we modify the size of the data being sent, making it harder for inspection tools to identify the data type or even its actual size. This technique helps bypass packet inspection in networks.

عند الحديث عن التلاعب بالبيانات، يمكننا استخدام تقنيات مثل "التلاعب في طول البيانات" أو "تغيير حجم البيانات" لتجاوز الفحص. الفكرة هنا هي أن الجدران النارية وأدوات الفحص الحديثة لا تكتفي بفحص البيانات فقط، بل تقوم أيضًا بفحص حجم البيانات.

في هذا النوع من الهجمات، نقوم بتعديل حجم البيانات المرسلة بحيث يصعب على أدوات الفحص تحديد نوع البيانات أو حتى حجمها الفعلي. هذه التقنية تساعد في تجاوز فحص الحزم في الشبكات.

Command Example (Nmap Data Length Manipulation):

nmap --data-length 100 [target]

Spoofing MAC Address

Sometimes, "Spoofing MAC Address" is used to bypass "Access Control" at the network layer. If there is a security policy that relies on MAC address for access control, we change the MAC address to make it appear as if it is from a different device.

في بعض الأحيان، يتم استخدام "Spoofing MAC Address" لتجاوز "Access Control" على مستوى الشبكة. إذا كان هناك سياسة أمان تعتمد على عنوان MAC للتحقق من الوصول إلى الشبكة، فإننا نقوم بتغيير عنوان MAC ليظهر كما لو كان جهاز آخر.

Command Example (Spoofing MAC Address using macchanger):

sudo macchanger -r eth0

This command randomly changes the MAC address of the eth0 network interface.

Source Port Manipulation

Sometimes, we use "Source Port Manipulation" to bypass filters. For example, we can use trusted ports like port 80 or 443 to send data without being detected.

أحيانًا نستخدم "Source Port Manipulation" لتجاوز الفلاتر. على سبيل المثال، يمكننا استخدام بورتات موثوقة مثل بورت 80 أو 443 لكي نتمكن من إرسال بيانات دون أن يتم اكتشافها.

Command Example (Using Nmap with Source Port 80):

nmap -p 80 --source-port 80 [target]

This command forces Nmap to use source port 80 when scanning the target.

Service Enumeration

Now we move to the topic of "Service Enumeration." After conducting a full network scan, we need to know the precise details about the protocols running on each port, such as FTP or SSH.

ننتقل الآن إلى موضوع "Service Enumeration". بعد إجراء مسح كامل للشبكة، نحتاج إلى معرفة تفاصيل دقيقة حول البروتوكولات التي تعمل على كل بورت، مثل FTP أو SSH.

Command Example (Service Version Detection with Nmap):

nmap -sV [target]

This command performs a service version detection scan and returns the versions of services running on open ports.

Spoofing and Bypass Control Policies

In networks, if there is a security policy that relies on checking MAC addresses or any other criteria to verify the device connected to the network, we might use "MAC Spoofing" to bypass these policies.

في الشبكات، إذا كانت هناك سياسة أمان تعتمد على فحص عناوين MAC أو أي معيار آخر للتحقق من الجهاز المتصل بالشبكة، فإننا قد نستخدم "MAC Spoofing" للتجاوز هذه السياسات.

Command Example (MAC Spoofing with ifconfig):

sudo ifconfig eth0 hw ether 00:11:22:33:44:55

This command manually sets the MAC address of eth0 to 00:11:22:33:44:55.

Service Enumeration Process

After performing a full scan of the network, we know the open ports like FTP and others. However, we need to gather additional details about each service running on these ports. This is crucial if we are planning a targeted attack against a specific service. This process involves collecting as much information as possible about each service.

بعد أن نقوم بإجراء مسح كامل (Full Scan) للشبكة، نعرف البورتات المفتوحة مثل FTP وغيرها. لكننا بحاجة لمعرفة تفاصيل إضافية حول كل خدمة تعمل على هذه البورتات. هذا أمر بالغ الأهمية إذا كنا نخطط لهجوم موجه ضد خدمة معينة. هذه العملية تتطلب جمع أكبر قدر ممكن من المعلومات حول كل خدمة.

Using Nmap for Service Enumeration

To perform a full service enumeration, we use a tool like Nmap. If we want to focus on the FTP service, we would use commands like -sV to detect the version that this service is running. However, in real-world scenarios, we might need additional information using Nmap's scripting capabilities.

لإجراء عملية التعداد الكامل للخدمات، نستخدم أداة مثل Nmap. إذا أردنا أن نركز على خدمةFTP ، فسنستخدم أوامر Nmap مثل -sV لاكتشاف الإصدار الذي تعمل به هذه الخدمة. ولكن في الحياة العملية، قد نحتاج إلى معلومات إضافية باستخدام السكربتات الخاصة بـNmap.

Command Example (Service Version Detection with Nmap):

nmap -sV [target]

This command detects the version of the service and provides details for each open port. However, if we want more information about an FTP service, we can use specific Nmap scripts like "banner grabbing" and "FTP/SSL."

هذا الأمر يقوم بالكشف عن نسخة الخدمة ويعرض التفاصيل المتعلقة بكل بورت مفتوح. ولكن إذا أردنا معرفة معلومات إضافية حول خدمةFTP ، يمكننا استخدام سكربتات Nmap المخصصة مثل سكربت "banner grabbing"و."FTP/SSL"

Using Banner Grabbing to Get Service Details

للحصول على تفاصيل حول الخدمة. باستخدام "Banner Grabbing" الخطوة التالية هي استخدام سكربت هذا السكربت، نستطيع استخراج معلومات عن الإصدار الحالي لأي خدمة تعمل على البورت. هذا السكربت يعمل على استخراج البيانات المخزنة في الحزم المرسلة من الخادم.

The next step is using the "Banner Grabbing" script to get details about the service. With this script, we can extract information about the current version of any service running on the port. This script works by extracting data stored in the packets sent by the server.

Command Example (Banner Grabbing with Nmap):

nmap -p 21 --script banner [target]

من خلال هذا الأمر، سنتمكن من الحصول على "البانر" الذي يحتوي على الإصدار الخاص بخدمة FTP أو أي خدمة أخرى تعمل على البورت المحدد.

With this command, we will be able to obtain the "banner" which contains the version of the FTP service or any other service running on the specified port.

Exploiting the Vulnerabilities Identified from Service Enumeration

Once we obtain version details using "Banner Grabbing," we move on to looking for vulnerabilities related to that version. For example, if we find that the FTP service is running an old version, we can search for vulnerabilities affecting that version.

If we discover that the version has a known vulnerability, such as a command execution vulnerability, we can exploit this vulnerability to compromise the system.

بمجرد أن نحصل على تفاصيل الإصدار باستخدام"Banner Grabbing" ، ننتقل إلى البحث عن الثغرات المتعلقة بهذا الإصدار. على سبيل المثال، إذا وجدنا أن خدمة FTP تعمل بإصدار قديم، يمكننا البحث عن الثغرات الأمنية التي تؤثر على هذا الإصدار.

إذا اكتشفنا أن الإصدار يحتوي على ثغرة معروفة، مثل ثغرة في تنفيذ الأوامر، يمكننا استخدام هذه الثغرة لاختراق النظام.

Using Password Cracking on FTP Services

In some cases, we might encounter an FTP service using "default username and password." In this case, we perform "Password Cracking" or "Brute Forcing" to break the password.

في بعض الحالات، قد نواجه خدمة FTP التي تستخدم "الاسم وكلمة المرور الافتراضية". في هذه الحالة، نقوم باستخدام "Password Cracking" أو "Brute Forcing" لاختراق كلمة المرور.

Command Example (Brute Forcing FTP Passwords with Hydra):

hydra -l user -P /path/to/passwordlist ftp://[target]

In this example, we are using the "Hydra" tool to attempt different passwords until we gain access to the service.

في هذا المثال، نستخدم أداة "Hydra" لمحاولة اختبار كلمات مرور مختلفة حتى نتمكن من الوصول إلى الخدمة.

FTP Service Enumeration

Let’s start with the FTP service. After performing the scan, if we find that the FTP service is open, we need to gather additional information about it, such as the version it is running on. One common method is using "banner grabbing" to get this information.

لنبدأ بخدمة FTP. بعد إجراء المسح، إذا اكتشفنا أن خدمة FTP مفتوحة، نحتاج إلى جمع معلومات إضافية عنها مثل الإصدار الذي يعمل عليه الخادم. أحد الأساليب الشائعة هو استخدام "banner grabbing" للحصول على هذه المعلومات.

Command Example (FTP Banner Grabbing using Nmap):

nmap -p 21 --script banner [target]

This method helps us know the version the server is running, which is crucial when attempting to exploit vulnerabilities associated with that version.

هذه الطريقة تساعدنا على معرفة الإصدار الذي يعمل عليه الخادم، وهو أمر بالغ الأهمية عند محاولة استغلال الثغرات المتعلقة بهذه النسخة.

Password Cracking with Hydra and Medusa

After gathering information about the service, we can move on to "Password Cracking." We can use tools like "Hydra" or "Medusa" to guess the password using a wordlist.

بعد جمع معلومات عن الخدمة، يمكننا الانتقال إلى مرحلة "Password Cracking". يمكننا استخدام أدوات مثل "Hydra" أو "Medusa" لتخمين كلمة المرور باستخدام قائمة كلمات المرور.(Wordlist)

Command Example (Using Hydra for FTP Password Cracking):

hydra -l user -P /path/to/wordlist ftp://[target]

In this example, we are using "Hydra" to test different passwords until we gain access to the service. This tool guesses the password using a list of potential words.

في هذا المثال، نستخدم أداة "Hydra" لاختبار كلمات مرور مختلفة حتى نتمكن من الوصول إلى الخدمة. هذه الأداة تقوم بتخمين كلمة المرور باستخدام قائمة من الكلمات المحتملة.

Using Telnet for Password Cracking

Another useful tool is "Telnet." If you have an open FTP or SSH service, you can use "Telnet" to manually connect to the service. Once connected, you can try different passwords to see if you can gain access.

أداة أخرى مفيدة هي "Telnet". إذا كان لديك خدمة FTP أو SSH مفتوحة، يمكنك استخدام "Telnet" للاتصال بالخدمة يدويًا. بمجرد الاتصال، يمكنك تجربة كلمات مرور مختلفة لمعرفة إذا كنت ستتمكن من الوصول.

Command Example (Using Telnet to connect to FTP):

telnet [target] 21

عند الاتصال بالخدمة عبر "Telnet" ، يمكنك البدء في تجربة كلمات مرور مختلفة. إذا كانت كلمات المرور صحيحة، ستتمكن من الوصول إلى النظام.

Once connected to the service via "Telnet," you can start trying different passwords. If the passwords are correct, you will gain access to the system.

SSH Service Enumeration

Now, let’s move to the SSH service. Like FTP, we can use "banner grabbing" to get information about the version the server is running. We can also use the same scanning techniques using Nmap.

الآن، دعونا ننتقل إلى خدمة SSH. مثلFTP ، يمكننا استخدام "banner grabbing" للحصول على معلومات حول الإصدار الذي يعمل عليه الخادم. يمكننا أيضًا استخدام نفس تقنيات المسح باستخدام.Nmap

Command Example (SSH Banner Grabbing using Nmap):

nmap -p 22 --script banner [target]

Once we know the version details, we can move on to exploiting the vulnerabilities associated with that version.

بمجرد أن نعرف تفاصيل النسخة، يمكننا الانتقال إلى مرحلة استغلال الثغرات الأمنية المرتبطة بهذا الإصدار.

SSH Password Cracking

For the SSH service, we can use the same tools like "Hydra" or "Medusa" to guess SSH passwords. We can also use "brute force" to crack the service.

بالنسبة لخدمةSSH ، يمكننا استخدام نفس الأدوات مثل "Hydra" أو "Medusa" لتخمين كلمات مرور SSH.نستطيع أيضًا استخدام "brute force" لاختراق الخدمة.

Command Example (Using Hydra for SSH Password Cracking):

hydra -l user -P /path/to/wordlist ssh://[target]

This tool will guess the password using a list of potential words until we gain access to the system.

هذه الأداة ستقوم بتخمين كلمة المرور باستخدام قائمة من الكلمات المحتملة حتى نتمكن من الوصول إلى النظام.

Exploiting FTP and SSH Vulnerabilities

Once we gain access to the service using the cracked passwords, we can perform "command execution" or even escalate privileges.

إذا تمكنا من الوصول إلى الخدمة باستخدام كلمات المرور المخترقة، يمكننا تنفيذ "command execution" أو حتى الحصول على صلاحيات المستخدم.

Once inside the system, we can use the appropriate commands to perform tasks like adding users or modifying files.

عند الوصول إلى النظام، يمكننا استخدام الأوامر المناسبة لتنفيذ مهام مثل إضافة مستخدمين أو تعديل الملفات.