**Footprinting** is a process that passively gathers information about a target from many diverse resources. The goal of footprinting is to learn about a target environment prior to launching an attack.

After the target has been analyzed and all relevant information organized, **port scanning** can take place. The goal of performing port scanning is to identify open and closed ports as well as the services running on a given system. Port scanning forms a critical step in the attack process because the hacker needs to identify what services are present and running on a target system prior to initiating an effective attack. Port scanning also helps to determine the course of action in future steps because once the nature of running services is identified, an attacker can select the best tools for the attack. Once a port scan has been thoroughly performed, the hacker can then move on to mapping the network and looking for vulnerabilities that can be exploited.

The first step in port scanning is **preparation** – specifically, the gathering of information about the range of IP addresses in use by the target. The port scanning process will be more accurate and effective because only the IP addresses on the intended victim will be scanned. **When getting information about the network ranges, you have 2 options:**

• With a manual registrar query, you simply go directly to the registration sites and query for information manually.

• With an automatic registrar query, you use web-based tools.

Once a valid network range has been determined, the next step is to identify active machines on the network. There are several ways that this task can be accomplished, including the following:

• **wardialing (legacy technique that is rarely used)**

• **wardriving and related activities**

• **pinging**

• **port scanning**

**Wardialing** is a technique that was popular in the 1980s and 1990s as a footprinting tool, which explains why the process involves the use of modems. Wardialing uses a modem to dial phone numbers to locate modems. An attacker who picked a town at random and dialed up a range of phone numbers in that town would likely turn up several computers with modems attached. Wardialing using modems is a legacy technique with extremely limited utility value today, but it does give rise to many similar techniques that essentially search for open communication ports.

**Wardriving** is another technique for uncovering access points into a network. Wardriving is the process of locating wireless access points and gaining information about the configuration of each. This “sniffing” was initially performed with a laptop computer, a car, and software designed to record the access points detected. Additionally, a Global Positioning System (GPS) was optionally included to go to the next step of mapping the physical location of the access points. Although there are a multitude of tools used to perform wardriving, other tools, including the following, are useful in defending against these attacks:

• **AirSnort** – wireless cracking tool

• **AirSnare** – an intrusion detection system to help you monitor your wireless networks. It can notify you as soon as an unapproved machine connects to your wireless network.

• **Kismet** – wireless network detector, sniffer, and intrusion detection system commonly found on Linux.

• **NetStumbler** – wireless network detector; also available for Mac and handheld devices.

Wardriving generally preys upon situations in which security is not considered or is poorly planned. Warchalking is used to mark the presence of access points with special symbols and glyphs used to inform others who might follow about the presence of a Wi-Fi network. **Warchalking** is a variation of traditional “hobo marks” or “hobo signs”, marks drawn in chalk on signs or buildings that told other hobos where they could get a meal or help during the Great Depression.

A technique that is useful at determining whether a system is present and active is a **ping sweep** of an IP address range. By default, a computer will respond to a ping request with a ping reply or echo. A **ping** is a network utility that sends an **Internet Control Message Protocol (ICMP)** message. With the use of a ping, it is possible to identify active machines and measure the speed at which packets are moved from one host to another as well as obtain details, such as the **time to live (TTL)**. A key advantage of ICMP scanning is that it can be performed rapidly because it runs scanning and analysis processes in parallel. In other words, it means multiple systems can be scanned simultaneously.

**Cons of pinging**:

• It is not uncommon for network administrators to specifically block ping ICMP messages at the firewall or even turn off ping messages completely on host devices.

• It is a safe bet that any **intrusion detection system (IDS)** or **intrusion prevention system (IPS)** that is in place will detect and alert network managers in the event a ping sweep occurs.

• Ping sweeps have no capability to detect systems that are attached to the network but powered down.

**Port Scanning** – The most straightforward technique is to scan ports for active services. Port scanning is designed to probe each port on a system in an effort to determine which ports are open. It is effective for gaining information about a host because the probes sent to a system have the capability to reveal more information than a ping sweep can. A successful port scan will return results that will give a clear picture of what services are running on a system. There are 65,535 TCP and 65,535 UDP ports on any given system.

**Common Port Numbers**

Port Service Protocol

20/21 FTP TCP

22 SSH TCP

23 Telnet TCP

25 SMTP TCP

53 DNS TCP/UDP

80 HTTP TCP

110 POP3 TCP

135 RPC TCP

161/162 SNMP UDP

1433/1434 MSSQL TCP

**Flags** are bits that are set in the header of a packet, each describing a specific behavior. A penetration tester or attacker with a good knowledge of these flags can use this knowledge to craft packets and tune scans to get the best results every time. **UDP is harder to use for port scans successfully**. The reason UDP is more difficult to use in port scanning is that as data is transmitted, there are no mechanisms designed to return feedback to the sender. One of the mechanisms that port scanning relies on is the use of flags. Flags are used in the TCP protocol to describe the status of a packet and the communication that goes with it. For example, a packet flagged with the FIN flag signals the end or clearing of a connection. The ACK flag is a signal used to indicate that a connection has been acknowledged. An XMAS scan is a packet that has the FIN, PSH, and URG flags active at once, in effect lighting it up “like a Christmas tree”, as the saying goes. Some of the more popular scans designed for TCP port scanning include:

• **TCP connect scan** – this type of scan is the most reliable but also the easiest to detect. This attack can be easily logged and detected because a full connection is established. Open ports reply with a SYN/ACK, whereas closed ports respond with an RST/ACK.

• **TCP SYN scan** – this type of scan is commonly referred to as half open because a full TCP connection is not established. This type of scan was originally developed to be stealthy and evade IDS systems, although most modern systems have adapted to detect it. Open ports reply with a SYN/ACK, whereas closed ports respond with an RST/ACK.

• **TCP FIN scan** – this scan attempts to detect a port by sending a request to close a nonexistent connection. This type of attack is enacted by sending a FIN packet to a target port; if the port responds with an RST, it signals a closed port. This technique is usually effective only on UNIX devices.

• **TCP NULL scan** – this attack is designed to send packets with no flags set. The goal is to elicit a response from a system to see how it responds and then use the results to determine the ports that are open and closed.

• **TCP ACK scan** – this scan attempts to determine access control list (ACL) rule sets or identify whether stateless inspection is being used. If an ICMP destination is unreachable, the port is considered to be filtered.

• **TCP XMAS tree scan** – this scan functions by sending packets to a target port with flags set in combinations that are illegal or illogical (e.g. FIN, PSH, and URG). The results are then monitored to see how a system responds. Closed ports return an RST.

Because there are several techniques that can be used to thwart port scanning, it is impossible to cover them all, but listed here are some countermeasures that can prevent an attacker from acquiring information from a port scan:

• **deny all** – an approach to access control designed to block all traffic to all ports unless such traffic has been explicitly approved

• **proper design** – an aspect of a carefully planned network, including security measures such as IDSs and firewalls

• **firewall testing** – a way to verify a firewall’s capability to detect and block undesirable traffic

• **port scanning** – a technique that utilizes the same tools that an attacker will use to attack a system with the goal of gaining a better understanding of the methods involved

• **security awareness training** – something every organization should strive to provide. Effective security awareness teaches personnel to know how to look for certain behaviors and maintain security. Security awareness should also be used to encourage that security policies and practices are being followed and help administrators determine whether adjustments need to be made.

**Mapping Open Ports** – No matter which tools are used, the activity at this point can be boiled down to determining whether a target is online and then port scanning the target. **Nmap (Network Mapper)** is one of the most widely used security tools, and a firm understanding of this utility is generally considered to be a basic requirement for security professionals. At its core, Nmap is a port scanner that has the capability to perform a number of different scan types. Nmap’s results can display the status of the port in one of three states:

* **open** – the target device is accepting connections on the port.
* **Closed** – a closed port is not listening or accepting connections.
* **Filtered** – a firewall, filter, or other network device is monitoring the port and preventing full probing to determining its status.

One of the more common types of scan is a full TCP connection scan (-sT) because it completes all 3 steps of the TCP handshake. Although a full connect scan is the most common, a stealth scan is seen as more covert because only two steps of the three-step handshake are performed. One of the techniques to perform a somewhat stealthy scan is a SYN scan, which performs only the first 2 steps. This type of scan is also known as half-open scanning because it does not complete the connection.

**Superscan** is a windows-based port scanner developed by Foundstone. This port scanner is designed to scan TCP and UDP ports, perform ping scans, run Whois queries, and use Traceroute.

**Scanrand** is a scanning tool that is designed to scan a single host up to large-scale networks quickly and then return results about the network. Scanrand is unique among network scanners because although most scanning tools scan one port at a time, Scanrand scans ports in parallel using what is known as stateless scanning. By using stateless scanning, Scanrand can perform scans much faster than other network scanners. Stateless scanning is an approach to scanning that splits scanning into 2 distinct processes. The 2 processes work together to complete the scanning process, with one process transmitted and the other listening for results.

**THC-Amap (The Hacker’s Choice – Another Mapper)** is a scanner that offers a different approach to scanning. SSL expects a handshake, but services might not return a banner. Amap handles this by storing a collection of normal responses that can be provided to ports to elicit a response. The tool also excels at allowing the security professional to find services that have been redirected from standard ports.

**The mere existence of an open port does not mean vulnerability exists.** The open ports that are discovered provide clues to what operating system the target is running. Determining the OS that is running on a specific target is the purpose of **OS fingerprinting**. Once an OS is identified, it is possible to better focus the attacks that come later. To identify an OS, there are 2 different methods that can be utilized: **active fingerprinting** or **passive fingerprinting**.

The process of active OS fingerprinting is accomplished by sending specially crafted packets to the targeted system. In practice, multiple probes or triggers are sent from the scanning system to the target. When the responses are received from a target system, an educated guess can be made as to the OS that is present. Though it may appear otherwise, **OS identification** is an accurate method of determining the system in place because the tools have become much more accurate than in the past.

**Xprobe2**, a commonly used active fingerprinting tool, relies on a unique method to identify an operating system known as *fuzzy* signature matching. This method consists of performing a series of tests against a certain target and collecting the results. Xprobe2 cannot say definitively which OS is running but instead uses the results to infer what system is running.

**Nmap** is effective at identifying the Oss of networked devices and generally can provide results that are highly accurate. Several Nmap options that can be used to fine-tune the scan include:

* **-sV Application version detection**
* **-O OS fingerprinting**
* **-A Both of the previous options**

**Passive fingerprinting**, by design, does not interact with the target system itself. It is a passive tool that monitors or captures network traffic. The traffic monitored is analyzed for patterns that would suggest which Oss are in use. Passive OS fingerprinting tools simply sniff network traffic and then match that traffic to specific OS signatures. One tool for performing passive OS fingerprinting is **pOf**, which can identify an OS using passive techniques. That means pOf can identify the target without placing any additional traffic on the network that can lead to detection. The tool makes attempts to fingerprint the system based on the incoming connections that are attempted.

Although passive OS fingerprinting generally does not yield results as quickly as active OS fingerprinting, there are still benefits. Passive OS fingerprinting allows an attacker to obtain information about a target without triggering network defensive measures, such as IDS or firewalls. Although the process may take longer than active fingerprinting, the benefit is that the victim has less chance of detecting and reacting to the impending attack. Remember: active fingerprinting contacts the host; passive does not.

**In general, there are several methods to create a network “map”. The common methods include:**

* manually creating a list of computers and devices. This method is most error prone and often results in devices and computers being overlooked, but it is easy.
* Using a software utility to scan your network to create a list of discovered computers and devices. This method is more comprehensive than the manual method, but it can detect only computers and devices that are on and connected to the network when the scan is run.
* Drawing or generating a visual map of your network. This method is really an extension of one of the previous methods.

Instead of just listing computers and devices, drawing a visual representation often helps clarify how your network is constructed and how it operates. You can use drawing utilities, such as Microsoft Visio or Network Notepad, to draw the map or use network mapping suites, such as SolarWinds or Spiceworks Map IT, to scan your network and generate the visuals for you.

Even without these tools, you should be able to manually map your findings. This information can be recorded in a notebook or a simple spreadsheet. This spreadsheet should contain domain name information, IP addresses, Domain Name System (DNS) servers, open ports, OS version, publicly available IP address ranges, wireless access points, modem lines, and application banner details you may have discovered.