**# Email foot printing**

- By monitoring the email delivery and inspecting the e-mail headers

- Information includes

- IP address of the recipient

- Geolocation of the recipient

- Delivery information

- Visited links

- Browser and OS information

- Reading time

- Can track emails using various \*\*email tracking tools\*\*

- E.g., notifies sender of the email being delivered and opened by the recipient

- Used by marketers, sellers etc.

**## Email header analysis**

- Helps to determine an e-mail contains something malicious or not

- Email-headers include

- Sender's name

- IP/Email address of the sender

- Mail server

- Mail server authentication system

- Send and delivery stamps

- Unique number of the message

**### Authentication protocol headers**

- Allows you to detect forged sender addresses.

- The goal is for sender to identify itself to the receiver.

- E-mail headers include information about their pass status

**#### SPF: Sender Policy Framework**

- E.g., `'PASS' with IP 209.85.220.69` or `'NEUTRAL' ...`

- Verifies if the domain of the e-mail owned by the sending server.

- If not passed, many e-mail providers just block it.

- Based on e-mail servers who publish records and says "here's the IP addresses we'll send e-mails"

**#### DKIM: DomainKeys Identified Mail**

- E.g., `'PASS' with domain accounts.google.com`

- Allows the receiver to verify that an email claimed to have come from a specific domain was authorized by the owner of that domain using a digital signature on the domain.

**#### DMARC: Domain-based Message Authentication, Reporting and Conformance**

- E.g., `PASS` or `FAIL`

- Combination of two protocols SPF + DKIM

- It builds on them and adds more policy

**## Verifying email legitimacy**

- Double check `FROM`

- Check the spelling in domain name so it is coming from the domain of the company

- If it is random e-mail check if it's from one of the biggest domain providers or if something legit.

- Check IP of the domain

- It can be someone’s computer (home router IP) or a private server

- Major mail service providers check to determine if domain of the e-mail is tied to the source IP of the e-mail (e.g., have a record)

- You can tie a public Wi-Fi (e.g. coffee shop) IP to domain and send the e-mails from there.

**## E-mail policies**

- Different e-mail service provider has different policies regarding to their SMTP

- Once hacker recognizes e-mail servers then then he/she can create accounts there, send e-mails back and further to figure out what the rules are.

- E.g., google does not allow you to see the IP address of the sender

- They proxy it behind one of their servers

- Workarounds are not so efficient.

- Each have own ruling list

- Determines e.g. what kind of files that can be send

**## Getting an IP address from an e-mail**

- You can then get IP and a lot from browser headers including

- browser information, OS info, device types

- Revealing your IP is not safe as even home routers have pretty static IP addresses

- Last usually 30 days up to 3 months

- You can still release DHCP lease in your home router settings to get a new IP from the ISP.

- You can send an image from a back-end server that you own

- Some e-mail providers request it and hide users IP

- You can send a direct link

- No e-mail provider can protect you from that

- Can be done through social engineering e.g.

- You know from social media that Bob was celebrating yesterday. You send an e-mail stating "Hi Bob, crew and I had a great time last night, you're never going to guess what Sam did in toilet, threw himself up, check out his pictures"

- E.g.

1. Install Apache ***“yum install httpd”***

2. Start Apache ***“systemctl start httpd”***

3. Create a file: ***“cd /var/www/html/”*** then ***“touch <RESOURCE\_NAME>;”***

4. Check logs live: ***“tail -f /var/log/httpd/access\_log***”

5. You'll get the IP address when the link ***(“<IP\_ADDRESS>/<RESOURCE\_NAME>”)*** is opened

- You can find out self IP address using ***“curl ifconfig.me”***

6. And you can look at the location of IP using ***“geoiplookup <IP\_ADDRESS>;”***

**# DNS Information Gathering**

DNS enumeration is one of the most popular reconnaissance tasks there is for building a profile of your target.

In plain english, it's the act of detecting and enumerating all possible DNS records from a domain name. This includes hostnames, DNS record names, DNS record types, TTLs, IP addresses, and a bit more, depending on how much information you're looking for.

With effective DNS enumeration, you can clone DNS zones manually, using scripts or by exploiting DNS zone transfer vulnerabilities, known as AXFR (Asynchronous Transfer Full Range) Transfer. This latter type of DNS transfer takes place when an attacker detects a misconfigured DNS server that is actually responding to AXFR requests.

**¶Impact**

Once DNS enumeration is completed, unauthenticated users may use this information to observe internal network records, grabbing useful DNS information that provides the attacker access to a full DNS map. This allows him to explore the attack surface area of any company, so he can later scan it, collect data, and—while he's at it—exploit it if there's an open opportunity.

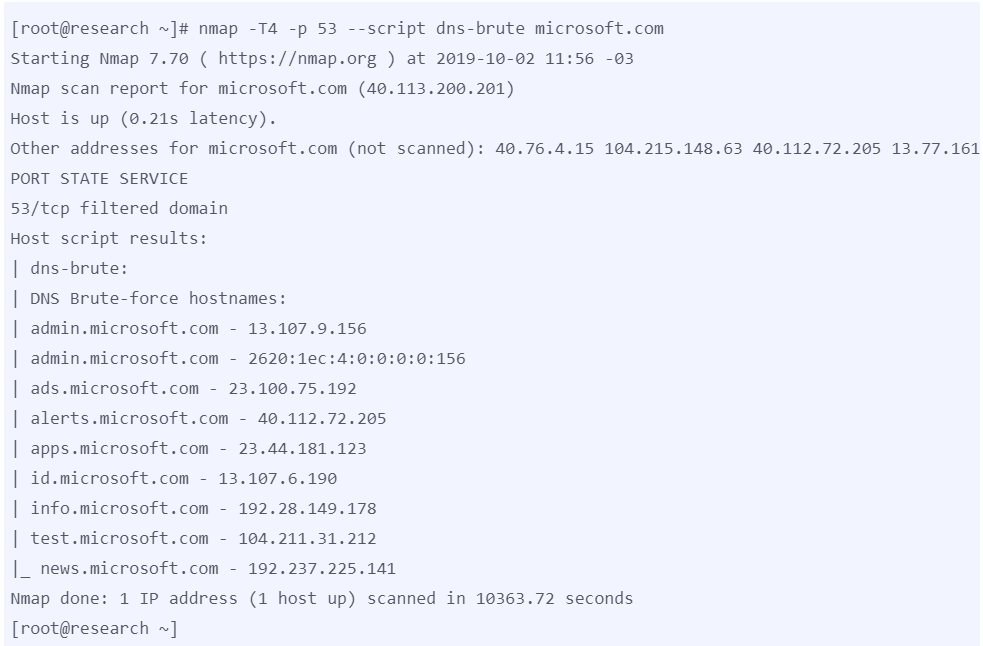
In the past we've seen a bit of DNS enumeration, such as in the How to Find Subdomains article. However, that was only focused on subdomains. Today we'll go one step forward and show you how to perform full DNS enumeration.

**DNS Information Gathering tools**

**Nmap**

Nmap was our 1st choice when we reviewed the best port scanners, but it's really more than that. This time it will help us reveal DNS information from a remote domain name.

By using the dns-brute script, Nmap will attempt to enumerate DNS hostnames by brute forcing popular subdomain names. In this case, we did it against microsoft.com and this was the result:



**DNSRecon**

DNSRecon is another great script that can help you discover DNS data from any given domain name.

It allows you to enumerate all types of DNS records, including A, AAAA, SPF, TXT, SOA, NS and MX, and also includes a brute force technique for grabbing subdomain and host A and AAAA records based on a wordlist.

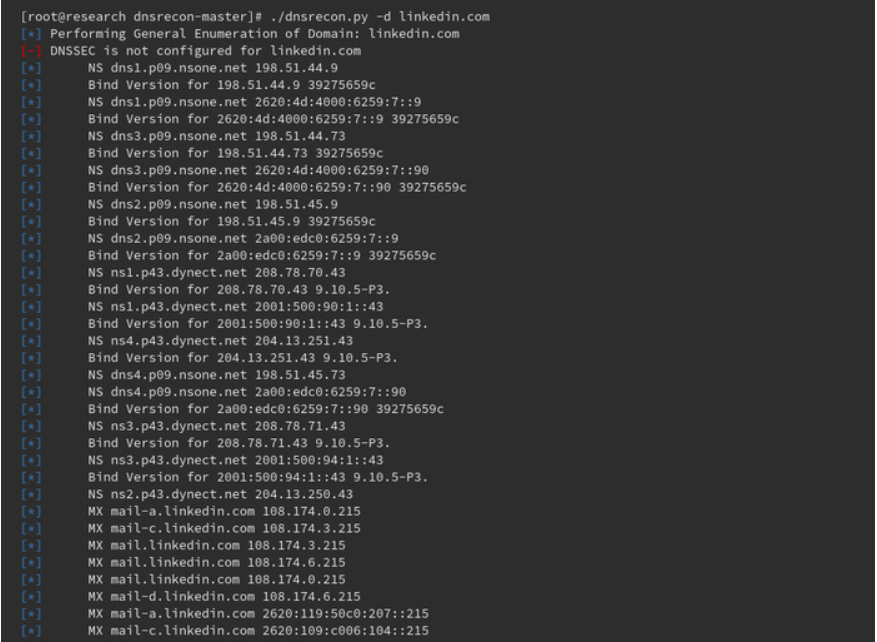
A cool thing we noticed is that it supports checking for cached A and AAAA DNS records on the DNS servers, as well as local DNS enumeration capabilities.

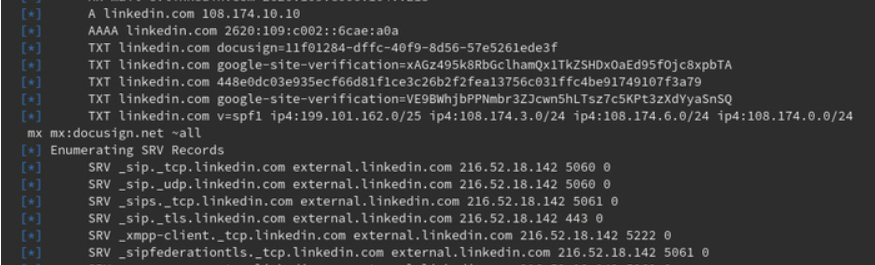
How can I perform DNS exploration with DNSRecon?

The easiest way is by using the -d parameter, as you see below:

****

Here we performed this dns enumeration against linkedin.com, and this was the result:

****

****

As shown, it was able not only to fetch multiple records (MX, A, AAAA, TXT, SRV and NS), but also to find some exposed bind versions from their DNS servers.

**# WHOIS Information Gathering**

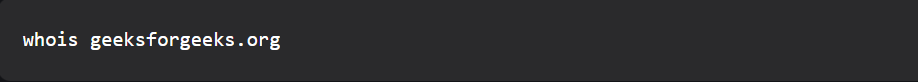
Whois is a widely used Internet record listing that identifies who owns a domain and how to get in contact with them. The Internet Corporation for Assigned Names and Numbers (ICANN) regulates domain name registration and ownership. Whois records have proven to be extremely useful and have developed into an essential resource for maintaining the integrity of the domain name registration and website ownership process.

whois is a database record of all the registered domains over the internet. It is used for many purposes, a few of them are listed below.

* It is used by Network Administrators in order to identify and fix DNS or domain-related issues.
* It is used to check the availability of domain names.
* It is used to identify trademark infringement.
* It could even be used to track down the registrants of the Fraud domain.

**Example:**

To use whois lookup, enter the following command in the terminal



**Expected output:**

