Lecture 4: Information Gathering

Part II

BKACAD's Security Training

Table of Content

Foot-printing Concept

Passive Information Gathering

Active Information Gathering

Active Information Gathering

Concept

Beyond passive information gathering and explore techniques that involve direct interaction with target services.

Main focus on DNS service.

DNS Enumeration

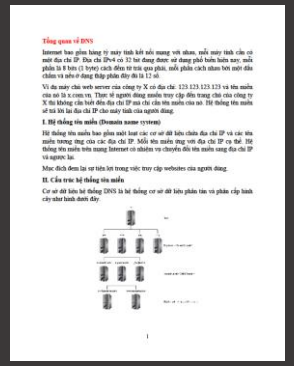
Concept

The Domain Name System (DNS) is one of the most critical systems on the Internet and is a distributed database responsible for translating user-friendly domain names into IP addresses.

This is facilitated by a hierarchical structure that is divided into several zones, starting with the top-level root zone.

www.megacorpone.com

For more information, check DNS-note.pdf file:



DNS Enumeration

DNS process

Process:

1. A hostname of Megacorp-One is entered into a browser or other application.
2. The browser passes the hostname to the operating system's DNS client and the operating system then forwards the request to the external DNS server it is configured to use. This first server in the chain is known as the DNS recursor and is responsible for interacting with the DNS infrastructure and returning the results to the DNS client. The DNS recursor contacts one of the servers in the DNS root zone. The root server then responds with the address of the server responsible for the zone containing the Top Level Domain (TLD), in this case, the .com TLD.
3. Once the DNS recursor receives the address of the TLD DNS server, it queries it for the address of the authoritative name-server for the megacorpone.com domain. The authoritative name-server is the final step in the DNS lookup process and contains the DNS records in a local database known as the zone file. It typically hosts two zones for each domain, the forward lookup zone that is used to find the IP address of a specific hostname and the reverse lookup zone (if configured by the administrator), which is used to find the hostname of a specific IP address.
4. Once the DNS recursor provides the DNS client with the IP address for www.megacorpone.com, the browser can contact the correct web server at its IP address and load the webpage.

DNS Enumeration

Interacting with a DNS Server

Each domain can use different types of DNS records. Some of the most common types of DNS records include:

- NS – Name-server records contain the name of the authoritative servers hosting the DNS records for a domain.
- A - Also known as a host record, the "a record" contains the IP address of a hostname (such as www.megacorpone.com).
- MX - Mail Exchange records contain the names of the servers responsible for handling email for the domain. A domain can contain multiple MX records.
- PTR - Pointer Records are used in reverse lookup zones and are used to find the records associated with an IP address.
- CNAME - Canonical Name Records are used to create aliases for other host records.
- TXT - Text records can contain any arbitrary data and can be used for various purposes, such as domain ownership verification.

```
kali@kali:~$ host -t ns megacorpone.com
```

```
C:\Users\administrator> nslookup
```

DNS Enumeration

Automating Lookups

Try Python with socket module

```
kali@kali:~$ python
>>> import socket
>>> socket.gethostbyname("notExistDomain.bkacad")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
socket.gaierror: [Errno -5] No address associated with hostname
>>> socket.gethostbyname("www.megacorpone.com")
'3.220.87.155'
>>>
```

DNS Enumeration

Exercise

Write a small script to brute-force DNS domain from Megacorp-One. List of domain need to brute-force as: www, ftp, mail, owa, proxy, router.

Hint: `python, import socket, for, try except, List...`

DNS Enumeration

DNS Zone Transfer

A zone transfer is basically a database replication between related DNS servers in which the zone file is copied from a master DNS server to a slave server. The zone file contains a list of all the DNS names configured for that zone. Zone transfers should only be allowed to authorized slave DNS servers but many administrators misconfigure their DNS servers, and in these cases, anyone asking for a copy of the DNS server zone will usually receive one.

```
kali@kali:~$ host -l <domain name> <dns server address>
```

DNS Enumeration

DNSRecon

DNSRecon is an advanced, modern DNS enumeration script written in Python. Running `dnsrecon` against megacorpone.com using the `-d` option to specify a domain name, and `-t` to specify the type of enumeration to perform (in this case a zone transfer)

```
kali@kali:~$ dnsrecon -d megacorpone.com -t axfr
```

DNS Enumeration

DNSRecon

To begin the brute force attempt, we will use the `-d` option to specify a domain name, `-D` to specify a file name containing potential subdomain strings, and `-t` to specify the type of enumeration to perform (in this case `brt` for brute force)

```
kali@kali:~$ dnsrecon -d megacorpone.com -D ~/list.txt
-t brt
```

DNS Enumeration

Exercise

1. Recreate the example above and use `dnsrecon` to attempt a zone transfer from megacorpone.com.
2. **[Very-Hard]** Write a small script to attempt a zone transfer from megacorpone.com using a higher-level scripting language such as Python.

Hint: `import dns, import argparse, import socket, for, try except, List…`

DNS Enumeration

Exercise – Walkthrough

```python
#!/usr/bin/env python

import argparse
import dns.zone
import dns.resolver
import socket

def main(address):
    soa_answer = dns.resolver.query(address, 'SOA')
    master_answer = dns.resolver.query(soa_answer[0].mname, 'A')
    try:
        z = dns.zone.from_xfr(dns.query.xfr(master_answer[0].address, address))
        names = z.nodes.keys()
        names.sort()
        for n in names:
            print(z[n].to_text(n))
    except socket.error as e:
        print('Failed to perform zone transfer:', e)
    except dns.exception.FormError as e:
        print('Failed to perform zone transfer:', e)


if __name__ == '__main__':
    parser = argparse.ArgumentParser(description='DNS Python')
    parser.add_argument('--address', action="store", dest="address",  default='dnspython.org')
    given_args = parser.parse_args()
    address = given_args.address
    main (address)
```