

Username: Palm Beach State College IP Holder **Book:** Kali Linux – Assuring Security by Penetration Testing. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

Chapter 4. Information Gathering

In this chapter, we will discuss the information gathering phase of penetration testing. We will describe the definition and purpose of information gathering. We will also describe several tools in Kali Linux that can be used for information gathering. After reading this chapter, we hope that the reader will have a better understanding of the information gathering phase and will be able to do information gathering during penetration testing.

Information gathering is the second phase in our penetration testing process (Kali Linux testing process) as explained in the *Kali Linux testing methodology* section in [Chapter 2, Penetration Testing Methodology](#). In this phase, we try to collect as much information as we can about the target, for example, information about the **Domain Name System (DNS)** hostnames, IP addresses, technologies and configuration used, username's organization, documents, application code, password reset information, contact information, and so on. During information gathering, every piece of information gathered is considered important.

Information gathering can be categorized in two ways based on the method used: **active** information gathering and **passive** information gathering. In the active information gathering method, we collect information by introducing network traffic to the target network. While, in the passive information gathering method, we gather information about a target network by utilizing a third-party's services, such as the Google search engine. We will cover this later on.

Note

Remember that no method is better in comparison to the other; each has its own advantage. In passive scanning, you gather less information but your action will be stealthy; while, in active scanning, you get more information but some devices may catch your action. During a penetration testing project, this phase may be done several times for the completeness of information collected. You may also discuss with your pen-testing customer, which method they want.

For this chapter, we will utilize the passive and active methods of information gathering to get a better picture of the target.

We will discuss the following topics in this chapter:

- Public websites that can be used to collect information about the target domain
- Domain registration information
- DNS analysis
- Route information
- Search engine utilization

Using public resources

On the Internet, there are several public resources that can be used to collect information regarding a target domain. The benefit of using these resources is that your network traffic is not sent to the target domain directly, so our activities are not recorded in the target domain logfiles.

The following are the resources that can be used:

No.	Resource URL	Description
1	http://www.archive.org	This contains an archive of websites.
2	http://www.domaintools.com/	This contains domain name intelligence.
3	http://www.alexa.com/	This contains the database of information about websites.
4	http://serversniff.net/	This is the free "Swiss Army Knife" for networking, server checks, and routing.
5	http://centralops.net/	This contains free online network utilities such as domain, e-mail, browser, ping, traceroute, and Whois.
6	http://www.robtex.com	This allows you to search for domain and network information.
7	http://www.pipl.com/	This allows you to search for people on the Internet by their first and last names, city, state, and country.

No.	Resource URL	Description
8	http://yname.com	This allows you to search for people across social networking sites and blogs.
9	http://wink.com/	This is a free search engine that allows you to find people by their name, phone number, e-mail, website, photo, and so on.
10	http://www.isearch.com/	This is a free search engine that allows you to find people by their name, phone number, and e-mail address.
11	http://www.tineye.com	TinEye is a reverse image search engine. We can use TinEye to find out where the image came from, how it is being used, whether modified versions of the image exist, or to find higher resolution versions.
12	http://www.sec.gov/edgar.shtml	This can be used to search for information regarding public listed companies in the Securities and Exchange Commission.

Due to the ease of use, you only need an Internet connection and a web browser, we suggest that you utilize these public resources first before using the tools provided with Kali Linux.

Note

To protect a domain from being abused, we have changed the domain name that we used in our examples. We are going to use several domain names, such as [example.com](#) from IANA and a dummy domain name [example.com](#) as well for illustrative purposes.

Username: Palm Beach State College IP Holder **Book:** Kali Linux – Assuring Security by Penetration Testing. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

Querying the domain registration information

After you know the target domain name, the first thing you would want to do is query the **Whois** database about that domain to look for the domain registration information. The **Whois** database will give information about the DNS server and the contact information of a domain.

WHOIS is a protocol for searching Internet registrations, databases for registered domain names, IPs, and autonomous systems. This protocol is specified in RFC 3912 (<https://www.ietf.org/rfc/rfc3912.txt>).

By default, Kali Linux already comes with a **whois** client. To find out the **Whois** information for a domain, just type the following command:

```
# whois example.com
```

The following is the abridged result of the **Whois** information:

```
Whois Server Version 2.0
```

```
Domain names in the .com and .net domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.
```

```
Domain Name: EXAMPLE.COM
Registrar: REGISTRAR.COM
Whois Server: whois.registrar.com
Referral URL: http://registrar.com
Name Server: NS.HOSTING.COM
Name Server: NS2.HOSTING.COM
Status: clientDeleteProhibited
Status: clientRenewProhibited
Status: clientTransferProhibited
Status: clientUpdateProhibited
Updated Date: 08-Apr-2012
Creation Date: 08-Apr-2012
Expiration Date: 08-Apr-2015
```

```
>>> Last update of whois database: Wed, 25 Jul 2012 02:15:41 UTC <<<
```

```
Please note: the registrant of the domain name is specified
in the "registrant" field. In most cases, registrar.com
is not the registrant of domain names listed in this database.
```

```
The Registrant:
```

```
Jalan Sudirman No. 1
DKI Jakarta
Indonesia 12345
```

```
Domain Name: EXAMPLE.COM
Created on: 08-Apr-12
Expires on: 08-Apr-15
Last Updated on: 08-Apr-12
```

Administrative Contact:

The Registrant
Jalan Sudirman No. 1
DKI Jakarta
Indonesia 12345
62 2112345678

Technical Contact:

The Registrant registrant@example.com
Jalan Sudirman No. 1
DKI Jakarta
Indonesia 12345
62 2112345678

Domain servers in listed order:

NS.HOSTING.COM
NS2.HOSTING.COM

From the preceding **Whois** result, we can get the information of the DNS server and the contact person of a domain. This information will be useful at the later stages of penetration testing.

Besides using the command-line **whois** client, the **Whois** information can also be collected via the following websites, which provide the **whois** client:

- www.whois.net
- www.internic.net/whois.html

Or, you can also go to the top-level domain registrar for the corresponding domain:

- America: www.arin.net/whois/
- Europe: www.db.ripe.net/whois
- Asia-Pacific: www.apnic.net/apnic-info/whois_search2

Note

Beware, that to use the top-level domain registrar **whois**, the domain needs to be registered through their own system. For example, if you use ARIN **WHOIS**, it only searches in the **ARIN WHOIS** database and will not search in the **RIPE** and **APNIC Whois** databases.

After getting information from the **Whois** database, next we want to gather information about the DNS entries of the target domain.

Username: Palm Beach State College IP Holder **Book:** Kali Linux – Assuring Security by Penetration Testing. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

Analyzing the DNS records

The goal of using the tools in the DNS records category is to collect information about the DNS servers and the corresponding records of a target domain.

The following are several common DNS record types:

No.	Record type	Description
1	SOA	This is the start of authority record.
2	NS	This is the name server record.
3	A	This is the IPv4 address record.
4	MX	This is the mail exchange record.
5	PTR	This is the pointer record.
6	AAAA	This is the IPv6 address record.
7	CNAME	This is the abbreviation for canonical name. It is used as an alias name for another canonical domain name.

For example, in a penetration test engagement, the customer may ask you to find out all of the hosts and IP addresses available for their domain. The only information you have is the organization's domain name. We will look at several common tools that can help you if you encounter this situation.

host

After we get the DNS server information, the next step is to find out the IP address of a hostname. To help us out on this matter, we can use the following

`host` command-line tool to lookup the IP address of a host from a DNS server:

```
# host www.example.com
```

The following is the command's result:

```
www.example.com has address 192.0.43.10
www.example.com has IPv6 address 2001:500:88:200::10
```

Looking at the result, we know the IPv4 and IPv6 addresses of the host `www.example.com`.

By default, the `host` command will look for the `A`, `AAAA`, and `MX` records of a domain. To query for any records, just give the `-a` option to the command.

```
# host -a example.com
Trying "example.com"
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 25153
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 2

;; QUESTION SECTION:
```

```

;example.com.                IN      ANY
;; ANSWER SECTION:
example.com.                 3201   IN      SOA      dns1.icann.org.
hostmaster.icann.org. 2012080782 7200 3600 1209600 3600
example.com.                 46840  IN      NS       a.iana-servers.net.
example.com.                 46840  IN      NS       b.iana-servers.net.

;; ADDITIONAL SECTION:
b.iana-servers.net.         1401   IN      A         199.43.133.53
a.iana-servers.net.         1401   IN      A         199.43.132.53

```

Received 170 bytes from 202.152.165.39#53 in 563 ms

The **host** command looks for these records by querying the DNS servers listed in the **/etc/resolv.conf** file of your Kali Linux system. If you want to use other DNS servers, just give the DNS server address as the last command-line option.

Note

If you give the domain name as the command-line option in **host**, the method is called forward lookup, but if you give an IP address as the command-line option to the **host** command, the method is called reverse lookup.

Try to do a reverse lookup of the following IP address:

```
host 23.23.144.81
```

What information can you get from this command?

The **host** tool can also be used to do a DNS zone transfer. With this mechanism, we can collect information about the available hostnames in a domain.

Note

A DNS zone transfer is a mechanism used to replicate a DNS database from a master DNS server to another DNS server, usually called a slave DNS server. Without this mechanism, the administrators have to update each DNS server separately. The DNS zone transfer query must be issued to an authoritative DNS server of a domain.

Due to the nature of information that can be gathered by a DNS zone transfer, nowadays, it is very rare to find a DNS server that allows zone transfer to an arbitrary zone transfer request.

If you find a DNS server that allows zone transfer without limiting who is able to do it, this means that the DNS server has been configured incorrectly.

The following is an example of performing DNS zone transfer for a domain via a misconfigured DNS server:

```
# host -l example.com ns4.isp.com
```

The following is the DNS zone transfer result:

```

Using domain server:
Name: ns4.isp.com
Address: 172.16.176.22#53
Aliases:

example.com name server ns1.isp.com.
example.com name server ns2.isp.com.
example.com has address 192.168.1.1
smtp.example.com has address 192.168.1.2
mail.example.com has address 192.168.1.3
webmail.example.com has address 192.168.1.3
www.example.com has address 192.168.1.4

```

The `host` command will return information about the NS, PTR, and address records of a domain. In this case, the misconfigured DNS server is `ns4.isp.com`.

dig

Besides the `host` command, you can also use the `dig` command to do DNS interrogation. The advantages of `dig` compared to `host` are its flexibility and clarity of output. With `dig`, you can ask the system to process a list of lookup requests from a file.

Let's use `dig` to interrogate the `example.com` domain:

```
root@kali:~# dig example.com

; <<>> DiG 9.8.4-rpz2+rl005.12-P1 <<>> example.com
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 3786
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
;example.com.                IN      A

;; ANSWER SECTION:
example.com.                 41023   IN      A      192.0.43.10

;; Query time: 14 msec
;; SERVER: 10.17.3.245#53(10.17.3.245)
;; WHEN: Mon May 20 08:53:09 2013
;; MSG SIZE rcvd: 45
```

Without giving any options besides the domain name, the `dig` command will only return the `A` record of a domain. To request for any other DNS record type, we can give the `type` option in the command line:

```
# dig example.com any

; <<>> DiG 9.7.0-P1 <<>> example.com any
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 40971
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 2

;; QUESTION SECTION:
;example.com.                IN      ANY

;; ANSWER SECTION:
example.com.                 3565   IN      SOA      dns1.icann.org.
hostmaster.icann.org. 2012080782 7200 3600 1209600 3600
example.com.                 83186  IN      AAAA     2001:500:88:200::10
example.com.                 48296  IN      NS       b.iana-servers.net.
example.com.                 48296  IN      NS       a.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net.         182    IN      A        199.43.132.53
b.iana-servers.net.         182    IN      A        199.43.133.53
```

```
;; Query time: 327 msec
;; SERVER: 202.152.165.39#53(202.152.165.39)
;; WHEN: Sat Aug 18 10:46:09 2012
;; MSG SIZE rcvd: 198
```

From the result, we can see that the `dig` output now returns the DNS records of `SOA` , `NS` , `A` , and `AAAA` .

To do zone transfer using `dig` , we must set the authoritative DNS server for that domain and set `axfr` as the type:

```
# dig @ns4.isp.com example.com axfr
```

Following is the abridged result of the preceding command:

```
; <<>> DiG 9.7.0-P1 <<>> @ns4.isp.com example.com axfr
; (1 server found)
;; global options: +cmd
example.com.          3600    IN      SOA     ns1.isp.com.
hostmaster.isp.com. 2011020409 900 600 86400 3600
example.com.          3600    IN      NS      ns1.isp.com.
example.com.          3600    IN      NS      ns4.isp.com.
example.com.          3600    IN      A       192.168.1.1
example.com.          3600    IN      MX      192.168.1.3
mail.example.com.     3600    IN      A       192.168.1.3
webmail.example.com. 3600    IN      A       192.168.1.3
www.example.com.      3600    IN      A       192.168.1.4
example.com.          3600    IN      SOA     ns1.isp.com.
hostmaster.isp.com. 2011020409 900 600 86400 3600
;; Query time: 855 msec
;; SERVER: 172.16.176.22#53 (172.16.176.22)
;; WHEN: Sat Aug 18 10:59:11 2012
;; XFR size: 9 records
```

We can see in the preceding result that the DNS records are similar to those of the `host` command. Based on this, we can be confident about the DNS records collected.

dnsenum

To collect information from a DNS server, we can utilize `dnsenum` . The DNS information that can be gathered is as follows:

- The host IP addresses
- The DNS server of a domain
- The `MX` record of a domain

Note

In this chapter, you may see that we used several tools that generate similar results, this is because we need to validate the information collected. If the information is found in more than one tool, we can be more confident with the information.

Besides being used to get DNS information, `dnsenum` also has the following features:

- Get additional names and subdomains utilizing the Google search engine.
- Find out subdomain names by brute forcing the names from the text files. The `dnsenum` tool included in Kali Linux comes with a `dns.txt` dictionary file that contains 1,480 subdomain names and a `dns-big.txt` file, which contains 266,930 subdomain names.

- Carry out `Whois` queries on C-class domain network ranges and calculate its network ranges.
- Carry out reverse lookup on network ranges.
- Use threads to process different queries.

To access `dnsenum` , go to the console and type the following command:

```
# dnsenum
```

This will display the usage instruction on your screen.

As an example of the `dnsenum` tool usage, we will use `dnsenum` to get DNS information from a target domain. The command to do this is as follows:

```
# dnsenum example.com
```

The following is the abridged result of that command:

```
dnsenum.pl example.com
dnsenum.pl VERSION:1.2.2
```

```
----- example.com -----
```

Host's addresses:

Name Servers:

```
ns1.isp.com      10771      IN      A        172.168.1.2
ns0.isp.com      7141       IN      A        172.168.1.1
```

Mail (MX) Servers:

```
hermes1.example.com 86400      IN      A        192.168.10.3
hermes.example.com  3600       IN      A        192.168.10.2
```

Trying Zone Transfers and getting Bind Versions:

```
Trying Zone Transfer for example.com on ns0.isp.com ...
AXFR record query failed: NOERROR
```

ns0.isp.com Bind Version:

DNS server

```
Trying Zone Transfer for example.com on ns1.isp.com ...
```

```
example.com      86400      IN      SOA
example.com      86400      IN      NS
```

```

example.com          86400    IN      MX
example.com          86400    IN      TXT
admin.example.com    3600     IN      NS
blogs.example.com    3600     IN      NS
ftp.example.com      3600     IN      A      192.168.10.4
hermes.example.com   3600     IN      A      192.168.10.2
hermes.example.com   86400    IN      TXT
hermes.example.com   86400    IN      SPF
hermes1.example.com  86400    IN      A      192.168.10.2
www.example.com      3600     IN      NS

```

ns1.isp.com Bind Version:
DNS server

brute force file not specified, bay.

Using the default options of `dnsenum`, we can get information about the host address, name servers, and the mail server's IP address. Fortunately, the `ns1.isp.com` DNS server allows us to do zone transfer for the `example.com` domain.

In the case that the zone transfer is not successful, we can do brute forcing of the lookups to find the subdomains from a wordlist. For example, if we want to brute force the subdomain using the provided text file wordlist (`dns.txt`), the following is the appropriate command:

```
dnsenum -f dns.txt example.com
```

The following is the result of the brute forcing process:

Brute forcing with
`dns.txt`:

```

apps.example.com    86400    IN      A
192.168.10.152
mail.example.com    86400    IN      A      192.168.10.107
portal.example.com  86400    IN      A      192.168.10.249

```

Beware that brute forcing the DNS lookups will take some time to finish.

Luckily for us, the target domain uses common subdomain names. So we are able to find several subdomains (`apps`, `mail`, and `portal`) in the target domain based on the dictionary file we have.

Another technique that can be used to find the subdomain is by using Google. This will be useful if the DNS zone transfer is disabled. To use Google, just add the options `-p` for the number of Google pages to be processed or `-S` to define the number of subdomains to be collected. You may also want to set the number of threads to do the queries (`--threads`) in order to speed up the process.

dnsdict6

Up until now, we only talked about the DNS tools to enumerate the subdomains in IP Version 4. If you want to enumerate the IP Version 6 subdomains, you can use `dnsdict6` from the **The Hacker's Choice (THC)** group.

To access `dnsdict6` in Kali Linux, you can use the console and type the following command:

```
# dnsdict6
```

It will display the `dsndict6` help page.

Without giving any options, **dnsdict6** will use the built-in wordlist and eight threads.

Let's enumerate the subdomains available in the **example.com** domain using the following command line:

```
# dnsdict6 example.com
```

The following screenshot shows the result of this command:

```
root@kali:~# dnsdict6 example.com
Starting DNS enumeration work on example.com. ...
Starting enumerating example.com. - creating 8 threads for 798 words...
Estimated time to completion: 1 to 2 minutes
www.example.com. => 2001:500:88:200::10

Found 1 domain name and 1 unique ipv6 address for example.com.
```

After brute forcing the subdomain using the **dnsdict6** built-in wordlist (containing 798 words), we know that there is only one subdomain (**www**) available in the **example.com** domain that has an IP Version 6 address.

Note

We found that the number of words displayed by **dnsdict6** is incorrect. We tested this using a file containing three words; the **dnsdict6** command informed us that the number of words is four.

Also, the **dnsdict6** tool can be used to find the subdomain on IP Version 4 using the **-4** option, and it can also collect information about the DNS and NS of a domain by using the **-d** option. Let's use these options to check the **example.com** domain:

```
root@kali:~# dnsdict6 -d -4 example.com
Starting DNS enumeration work on example.com. ...
Gathering NS and MX information...
NS of example.com. is b.iana-servers.net. => 199.43.133.53
NS of example.com. is b.iana-servers.net. => 2001:500:8d::53
NS of example.com. is a.iana-servers.net. => 199.43.132.53
NS of example.com. is a.iana-servers.net. => 2001:500:8c::53
Warning: no mail sever (MX) information found

Starting enumerating example.com. - creating 8 threads for 798 words...
Estimated time to completion: 1 to 2 minutes
Warning: wildcard domain configured
*.example.com. -> 124.81.172.106
Warning: wildcard domain configured (2nd test)
www.example.com. => 192.0.43.10
www.example.com. => 2001:500:88:200::10

Found 1 domain name, 2 unique ipv4 and 1 unique ipv6 addresses for example.com.
```

fierce

The **fierce** tool is a DNS enumeration tool that uses several techniques to find all of the IP addresses and hostnames of a target. It works by first querying your system's DNS server for the target DNS server; next, it uses the target DNS server. It also supports the wordlist supplied by the user to find subdomain names. It does this recursively until all of the wordlist items are tested. The main feature of **fierce** is that it can be used to locate noncontiguous IP space and hostnames against specified domains.

To access **fierce** in Kali Linux, you can use the console and type the following command:

```
# fierce -h
```

This will display the usage instructions on your screen.

As an example, let's use **fierce** to find information about a domain:

```
# fierce -dns example.com -threads 3
```

The following is the abridged result:

```
DNS Servers for targetdomain.com:
```

```
ns4.example.com
ns1.example.com
ns2.example.com
ns3.example.com
```

```
Trying zone transfer first...
```

```
Testing ns4.example.com
    Request timed out or transfer not allowed.
Testing ns1.example.com
    Request timed out or transfer not allowed.
Testing ns2.example.com
    Request timed out or transfer not allowed.
Testing ns3.example.com
    Request timed out or transfer not allowed.
```

```
Unsuccessful in zone transfer (it was worth a shot)
```

```
Okay, trying the good old fashioned way... brute force
```

```
Checking for wildcard DNS...
```

```
Nope. Good.
```

```
Now performing 1895 test(s)...
```

```
192.168.116.3    voips.example.com
192.168.116.7    ns.example.com
192.168.116.19  streaming.example.com
192.168.117.50  dev.example.com
192.168.117.16  mx1.example.com
192.168.117.17  mx2.example.com
192.168.117.18  mx3.example.com
192.168.117.16  imap.example.com
192.168.117.5   www.example.com
192.168.117.6   intra.example.com
192.168.117.17  mail.example.com
192.168.117.5   web.example.com
192.168.117.16  webmail.example.com
```

```
Subnets found (may want to probe here using nmap or unicornscan):
```

```
192.168.73.0-255 : 2 hostnames found.
192.168.46.0-255 : 1 hostnames found.
192.168.116.0-255 : 34 hostnames found.
192.168.117.0-255 : 25 hostnames found.
```

```
Done with Fierce scan: http://ha.ckers.org/fierce/
```

```
Found 62 entries.
```

```
Have a nice day.
```

It may take some time to finish the DNS enumeration using **fierce** .

Note

In this section, we talked a lot about finding hostnames for a domain; you may ask what are the purposes of these hostnames. In a penetration testing project, one of the authors found a web meeting session after getting the hostnames' result from the DNS analysis phase. That host allowed the author to join the ongoing web meeting session.

DMitry

DMitry (Deepmagic Information Gathering Tool) is an all-in-one information gathering tool. It can be used to gather the following information:

- The **Whois** record of a host by using the IP address or domain name
- Host information from Netcraft.com
- Subdomains in the target domain
- The e-mail address of the target domain
- Open, filtered, or closed port lists on the target machine by performing a port scan

Even though this information can be obtained using several Kali Linux tools, it is very handy to gather all of the information using a single tool and to save the report to one file.

Note

We thought this tool is more suitable to be categorized under DNS analysis instead of the Route analysis section because the capabilities are more about DNS analysis rather than in routing analysis.

To access **DMitry** from the Kali Linux menu, navigate to **Applications | Kali Linux | Information Gathering | OSINT Analysis | dmitry** or you can use the console and type the following command:

```
# dmitry
```

As an example, let's do the following to a target host:

- Perform a **Whois** lookup
- Get information from Netcraft.com
- Search for all the possible subdomains
- Search for all the possible e-mail addresses

The command for performing the mentioned actions is as follows:

```
# dmitry -iwnse targethost
```

The following is the abridged result of the preceding command:

```
Deepmagic Information Gathering Tool
"There be some deep magic going on"

HostIP:192.168.xx.xx
HostName:targethost
Gathered Netcraft information for targethost
-----
Retrieving Netcraft.com information for targethost
No uptime reports available for host: targethost

Gathered Subdomain information for targethost
-----
```

```

Searching Google.com:80...
HostName:targethost
HostIP:192.168.xx.xx
HostName:www.ecom.targethost
HostIP:192.168.xx.xx
HostName:blogs.targethost
HostIP:192.168.xx.xx
HostName:static.targethost
HostIP:192.168.xx.xx
HostName:webmail.targethost
HostIP:192.168.xx.xx
...
Gathered E-Mail information for targethost
-----
Found 0 E-Mail(s) for host targethost, Searched 0 pages containing 0
results

```

We can also use **DMitry** to perform a simple port scan by giving the following command:

```
# dmitry -p targethost -f -b
```

The result of the preceding command is as follows:

```

Deepmagic Information Gathering Tool
"There be some deep magic going on"

HostIP:192.168.xx.xx
HostName:targethost

Gathered TCP Port information for 192.168.xx.xx
  Port          State
...
80/tcp          open
...
135/tcp         filtered
136/tcp         filtered
137/tcp         filtered
138/tcp         filtered
139/tcp         filtered

Portscan Finished: Scanned 150 ports, 138 ports were in state closed

```

From the preceding command, we find that the targethost is using a device to do packet filtering. It only allows incoming connections to port **80**, which is commonly used for a web server.

Maltego

Maltego is an open source intelligence and forensics application. It allows you to mine and gather information and represent the information in a meaningful way. The word open source in Maltego means that it gathers information from the open source resources. After gathering the information, Maltego allows you to identify the key relationship between the information gathered.

Maltego is a tool that can graphically display the links between data, so it will make it easier to see the common aspects between pieces of information.

Maltego allows you to enumerate the following Internet infrastructure information:

- Domain names
- DNS names

- **Whois** information
- Network blocks
- IP addresses

It can also be used to gather the following information about people:

- Companies and organizations related to the person
- E-mail addresses related to the person
- Websites related to the person
- Social networks related to the person
- Phone numbers related to the person

Kali Linux, by default, comes with Maltego 3.3.0 Kali Linux edition. The following are the limitations of the community version (<http://www.paterva.com/web5/client/community.php>):

- Not for commercial use
- A maximum of 12 results per transform
- You need to register yourself on our website to use the client
- API keys expire every couple of days
- Runs on a (slower) server that is shared with all community users
- Communication between client and server is not encrypted
- Not updated until the next major version
- No end user support
- No updates of transforms on server side

There are more than 70 transforms available in Maltego. The word transform refers to the information gathering phase of Maltego. One transform means that Maltego will only do one phase of information gathering.

To access Maltego from the Kali Linux menu, navigate to **Kali Linux | Information Gathering | OSINT Analysis | maltego** or you can use the console and type the following command:

```
# maltego
```

You will see the Maltego welcome screen. After several seconds, you will see the following Maltego start-up wizard that will help you set up the Maltego client for the first time:

Welcome to Maltego!

Steps

1. **Welcome**
2. Login
3. Login result
4. Select transform seeds
5. Update transforms



Startup wizard - Welcome (1 of 5)

Welcome to Maltego!

This wizard will guide you through the steps of setting up your Maltego Client for first use.

We hope that you enjoy using our product as much as we enjoy building it!

Please note that the Community Edition is intended for non-commercial use only!

< Back
Next >
Finish
Cancel
Help

Click on **Next** to continue to the next window as shown in the following screenshot:

Welcome to Maltego!

Steps

1. Welcome
2. **Login**
3. Login result
4. Select transform seeds
5. Update transforms



Startup wizard - Login (2 of 5)

Enter your details below to log in to the Maltego Community Server

Or if you have not done so yet, [register here](#)

Login

* Email Address

Password

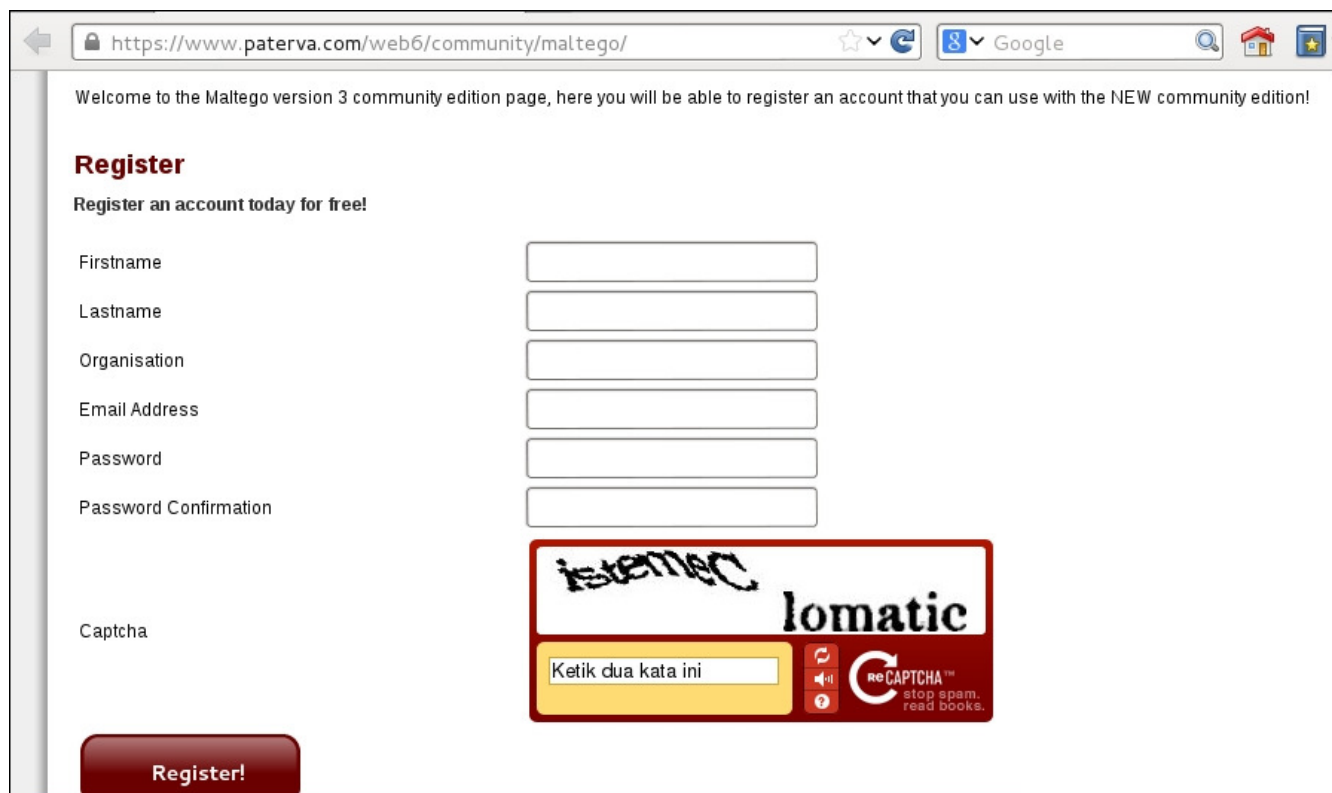


* Solve captcha

< Back
Next >
Finish
Cancel
Help

In this window, you need to enter your login information to the Maltego community server. If you don't have the login information, you need to register yourself first by clicking on the **register here** link.

The following screenshot shows the **Register** page:



Welcome to the Maltego version 3 community edition page, here you will be able to register an account that you can use with the NEW community edition!

Register

Register an account today for free!

Firstname

Lastname

Organisation

Email Address

Password

Password Confirmation

Captcha



Ketik dua kata ini

Register!

You need to fill in your details into the corresponding fields provided, and click on the **Register!** button to register.

If you already have the login details, you can enter them in the fields provided. When the login information is correct, the following information will be displayed:



Welcome to Maltego!

Startup wizard - Login result (3 of 5)

Hello Tedi, welcome to Maltego Community Edition!

Personal details

First name


Surname

Email address

Your API key is valid until May 22, 2013 at 12:00:00 AM WIT

Steps

1. Welcome
2. Login
3. **Login result**
4. Select transform seeds
5. Update transforms



MALTEGO RADIUM CE
KALI LINUX


< Back Next > Finish Cancel Help

You will then need to select the transform seeds as shown in the following screenshot:

Welcome to Maltego!

Steps

1. Welcome
2. Login
3. Login result
4. **Select transform seeds**
5. Update transforms



MALTEGO RADIUM CE
KALI LINUX

Startup wizard - Select transform seeds (4 of 5)

Discover transforms from:

☒ Maltego public servers

☐ Local TAS (Transform Application Server)

Hostname/IP:

Note: The transform seed settings can be changed later through Manage->Discover Transforms.


< Back
Next >
Finish
Cancel
Help

The Maltego client will connect to the Maltego servers in order to get the transforms. If Maltego has been initialized successfully, you will see the following screenshot:

Welcome to Maltego!

Steps

1. Welcome
2. Login
3. Login result
4. Select transform seeds
5. **Update transforms**



MALTEGO RADIUM CE
KALI LINUX

Startup wizard - Update transforms (5 of 5)

Ready...Set...GO!

Your new Maltego client has been initialized successfully!

2 new application server(s) were found
48 new transforms were found
51 new entities were installed

You are now ready to use Maltego!

☒ Run a machine (NEW!!)

☐ Open a blank graph and let me play around

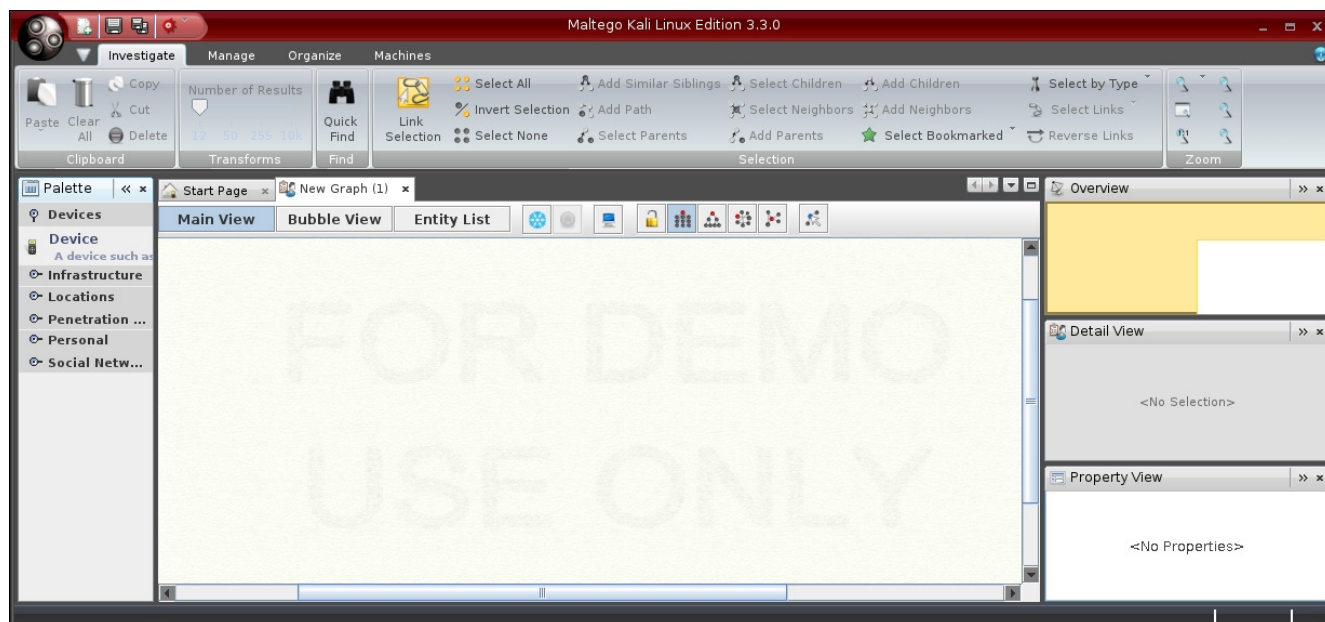
☐ Open an example graph

☐ Go away, I have done this before!

< Back
Next >
Finish
Cancel
Help

This means that your Maltego client initialization has been done successfully. Now you can use the Maltego client.

Before we use the Maltego client, let's first see the Maltego interface:



On the top-left side of the preceding screenshot, you will see the **Palette** window. In the **Palette** window, you can choose the entity type for which you want to gather the information. Maltego divides the entities into six groups as follows:

- **Devices** such as phone or camera
- **Infrastructure** such as AS, DNS name, domain, IPv4 address, MX record, NS record, netblock, URL, and website
- **Locations** on Earth
- **Penetration testing** such as built with technology
- **Personal** such as alias, document, e-mail address, image, person, phone number, and phrase
- **Social Network** such as Facebook object, Twitter entity, Facebook affiliation, and Twitter affiliation

In the top-middle of the preceding screenshot, you will see the different views: **Main View**, **Bubble View**, and **Entity List**. Views are used to extract information that is not obvious from large graphs—where the analyst cannot see clear relationships by manual inspection of data. **Main View** is where you work most of the time. In **Bubble View**, the nodes are displayed as bubbles, while in the **Entity List** tab, the nodes are simply listed in text format.

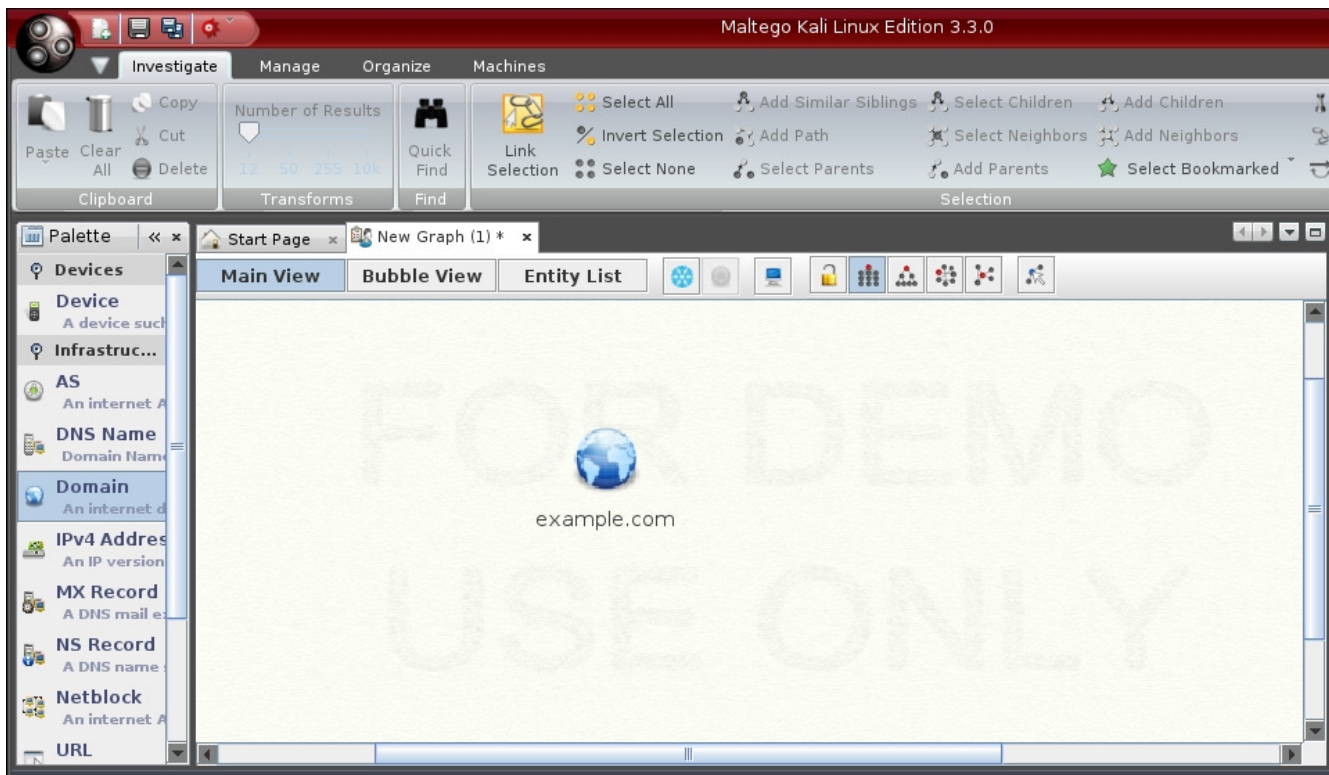
Next to the views, you will see different layout algorithms. Maltego supports the following four layout algorithms:

- **Block layout**: This is the default layout and is used during mining
- **Hierarchical layout**: Think of this as a tree-based layout, such as a file manager
- **Centrality layout**: Nodes that are the most central to the graph (for example, most incoming links) appear in the middle, with the other nodes scattered around it
- **Organic layout**: Nodes are packed together tightly in such a way that the distance between each node and all the other nodes is minimized

After a brief description of the Maltego client user interface, it's time for the action.

Let's suppose you want to gather information about a domain. We will use the domain example.com for this example. We will explore how to do this in the following sections.

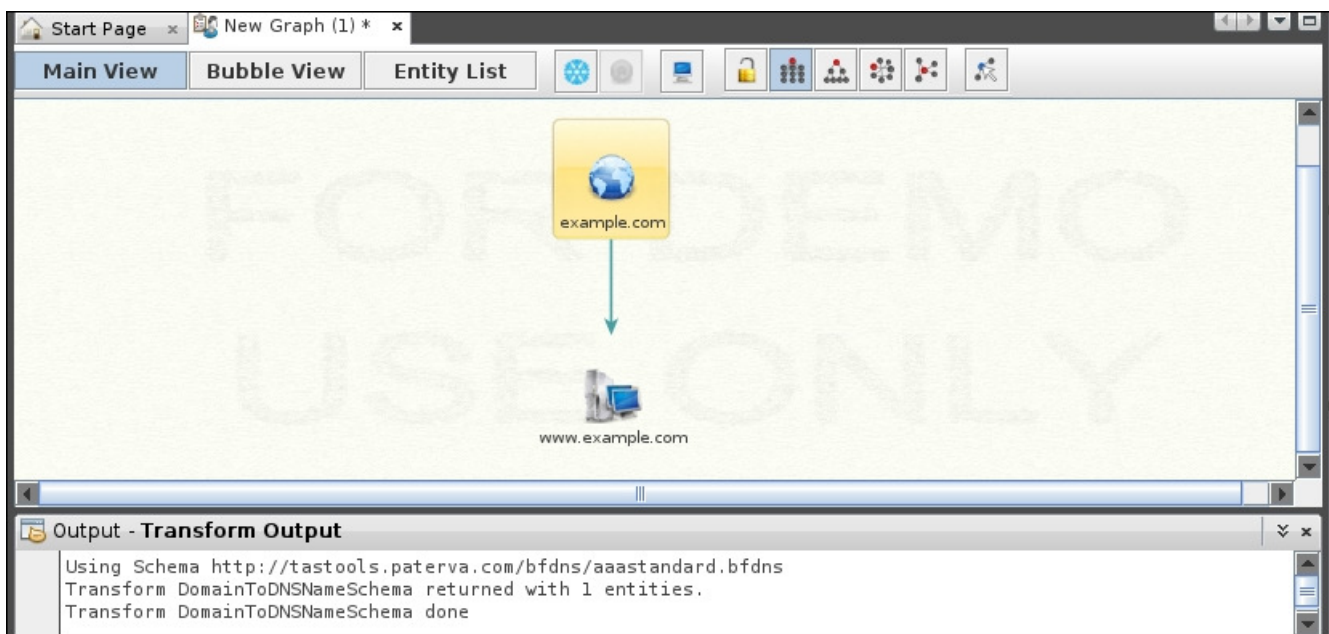
Create a new graph (*Ctrl + T*), go to the **Palette** tab, select **Infrastructure**, and click on **Domain**. Drag it to the main window. If successful, you will see a domain called **paterva.com** in the main window. Double-click on the name and change it to your target domain, such as example.com, as shown in the following screenshot:



If you right-click on the domain name, you will see all of the transforms that can be done to the domain name:

- DNS from domain
- Domain owner's details
- E-mail addresses from domain
- Files and documents from domain
- Other transforms, such as To Person, To Phone numbers, and To Website
- All transforms

Let's choose **DomainToDNSNameSchema** from domain transforms (**Run Transform** | **Other Transforms** | **DomainToDNSNameSchema**). The following screenshot shows the result:



After the **DNS from Domain** transform, we got information on the website address (**www.example.com**) related to the **example.com** domain.

You can run other transforms to the target domain.

If you want to change the domain, you need to save the current graph first. To save the graph, click on the Maltego icon, and then select **Save**. The graph will be saved in the Maltego graph file format (**.mtgx**). To change the domain, just double-click on the existing domain and change the domain name.

Next, we will describe several tools that can be used for getting route information.

Username: Palm Beach State College IP Holder **Book:** Kali Linux – Assuring Security by Penetration Testing. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

Getting network routing information

The tools in this category can be used to get the network routing information of a target. We will describe several tools that are commonly used for this purpose. Knowledge of the network routing information will allow the penetration tester to understand the network of the target machine, such as which path is taken by the packets sent from the penetration tester machine to the target machine. The routing information will also give a clue as to whether the particular target is protected by firewall.

Let us see the several tools that can help you get routing information.

tcptraceroute

The `tcptraceroute` tool can be used as a complement to the `traceroute` command. The `traceroute` command sends a UDP or ICMP echo request packet with a **Time To Live (TTL)** of one and increments the TTL until the packet reaches the target, while the `tcptraceroute` tool uses TCP SYN to send out the packet to the target.

The advantage of using `tcptraceroute` is that, nowadays, it is common to find a firewall device filtered `traceroute` packet, so it will not be possible to trace the network path to the target completely. However, this firewall still allows a packet to reach a particular TCP port in the target machine. By using `tcptraceroute`, we will be able to find the network path to the target, even though there is a firewall in front of it.

The `tcptraceroute` tool will receive a SYNACK packet if the port is open and a RST packet if the port is closed.

To access `tcptraceroute`, you can use the console and type the following command:

```
# tcptraceroute
```

This will display usage information on your screen.

Let's go for some action.

We run the `traceroute` command to trace our network route to the `example.com` domain as follows:

```
# traceroute www.example.com
```

The redacted result for this command is as follows:

```
traceroute to www. example .com (192.168.10.100), 30 hops max, 40 byte
packets
 1 192.168.1.1 (192.168.1.1)  8.382 ms  12.681 ms  24.169 ms
 2 1.static.192.168.xx.xx.isp (192.168.2.1)  47.276 ms  61.215 ms  61.057
ms
 3  * * *
 4 74.subnet192.168.xx.xx.isp (192.168.4.1)  68.794 ms  76.895 ms  94.154
ms
 5 isp2 (192.168.5.1)  122.919 ms  124.968 ms  132.380 ms
...
15 * * *
...
30 * * *
```

After route number `15`, we are no longer able to get the route information. Usually, this is because the `traceroute` packets are blocked by a filtering device.

We will try again using `tcptraceroute`, and we know that the target host has an open TCP port for the web server (`80`). We can use the following command:

```
# tcptraceroute www.example.com
```


The result for this command is as follows:

```

Selected device eth0, address 192.168.1.107, port 41884 for outgoing
packets
Tracing the path to www. example .com (192.168.10.100) on TCP port 80
(www),          30 hops max
 1  192.168.1.1  55.332 ms  6.087 ms  3.256 ms
 2  1.static.192.168.xx.xx.isp (192.168.2.1)    66.497 ms
50.436          ms  85.326 ms
 3  * * *
 4  74.subnet192.168.xx.xx.isp (192.168.4.1)  56.252 ms  28.041 ms  34.607
ms
 5  isp2 (192.168.5.1)  51.160 ms  54.382 ms  150.168 ms
 6  192.168.6.1  106.216 ms  105.319 ms  130.462 ms
 7  192.168.7.1  140.752 ms  254.555 ms  106.610 ms
...
14  192.168.14.1  453.829 ms  404.907 ms  420.745 ms
15  192.168.15.1  615.886 ms  474.649 ms  432.609 ms
16  192.168.16.1 [open]  521.673 ms  474.778 ms  820.607 ms

```

This time, our packet is able to reach the targethost, and it gives us all the route information from our machine to the targethost.

tctrace

Another tool that can be used to do route analysis is **tctrace** . It works by sending a TCP SYN packet to the target.

To access **tctrace** , you can use the console and type the following command:

```
# tctrace -i<device> -d<targethost>
```

In the preceding command, **-i** is the network interface to the target and **-d** is the target.

To run **tctrace** to a target, the following command is used:

```
# tctrace -i eth0 -d www.example.com
```

The following result is obtained:

```

1(1)  [192.168.1.1]
2(1)  [192.168.2.1]
3(all) Timeout
4(3)  [192.168.4.1]
5(1)  [192.168.5.1]
6(1)  [192.168.6.1]
7(1)  [192.168.7.1]
...
14(1) [192.168.14.1]
15(1) [192.168.15.1]
16(1) [192.168.16.1] (reached; open)

```

Username: Palm Beach State College IP Holder **Book:** Kali Linux – Assuring Security by Penetration Testing. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

Utilizing the search engine

The Kali Linux tools grouped in this category can be used to collect domain, e-mail address, and document metadata information from the target. These tools use a search engine to do their actions. The advantage of these tools is that they use search engine sites. So, you don't access the target website yourself, instead the search engine site will do that for you. As a result, the target website will not know about your action.

Let us explore several of these tools.

theharvester

The **theharvester** tool is an e-mail accounts, username, and hostname/subdomains gathering tool. It collects information from various public sources. As of Version 2.2, the public sources that are supported are as follows:

- Google
- Google profiles
- Bing
- PGP
- LinkedIn
- Yandex
- People123
- Jigsaw
- Shodan

To access **theharvester** in Kali Linux, you can use the console and type the following command:

```
# theharvester
```

This will display the usage information and example on your screen.

If we want to find the e-mail addresses and hostnames for a target domain using Google and limit the result to 100, the following is the appropriate command:

```
# theharvester -d example.com -l 100 -b google
```

The following e-mail addresses and hostnames are found:

```
[-] Searching in Google:
    Searching 0 results...
```

```
[+] Emails found:
```

```
-----
info@ example .com
user1@ example .com
user2@ example .com
user3@ example .com
```

```
[+] Hosts found in search engines:
```

```
-----
192.168.118.14:sd1. example .com
192.168.118.14:sd2. example .com
192.168.118.14:event. example .com
192.168.118.14:test. example .com
```



```
203.34.118.7:nms. example .com
```

From the preceding result, we notice that we are able to get several e-mail addresses and hostnames from the Google search engine.

If we want to gather more information, let's say we want to collect the username from the target, we can use linkedin.com to do this. The following is the command for that:

```
# theharvester -d example.com -l 100 -b linkedin
```

The following is the result:

```
[ - ] Searching in Linkedin..
      Searching 100 results..
Users from Linkedin:
```

```
user1
user2
user3
user4
user5
user6
```

```
Total results: 6
```

The preceding list of usernames collected from LinkedIn will be useful in a penetration testing step later if we want to do an attack, such as a social engineering attack.

Metagoofil

Metagoofil is a tool that utilizes the Google search engine to get metadata from the documents available in the target domain. Currently, it supports the following document types:

- Word document (`.docx` , `.doc`)
- Spreadsheet document (`.xlsx` , `.xls` , `.ods`)
- Presentation file (`.pptx` , `.ppt` , `.odp`)
- PDF file (`.pdf`)

Metagoofil works by performing the following actions:

- Searching for all of the preceding file types in the target domain using the Google search engine
- Downloading all of the documents found and saving them to the local disk
- Extracting the metadata from the downloaded documents
- Saving the result in an HTML file

The metadata that can be found are as follows:

- Usernames
- Software versions
- Server or machine names

This information can be used later on to help in the penetration testing phase.

To access **Metagoofil** , go to the console and execute the following command:

```
# metagoofil
```

This will display a simple usage instruction and example on your screen.

As an example of **Metagoofil** usage, we will collect all the DOC and PDF documents (**-t .doc , .pdf**) from a target domain (**-d example.com**) and save them to a directory named **test** (**-o test**). We limit the search for each file type to 20 files (**-l 20**) and only download five files (**-n 5**). The report generated will be saved to **test.html** (**-f test.html**). We give the following command:

```
# metagoofil -d example.com -l 20 -t doc,pdf -n 5 -f test.html -o test
```

The redacted result of this command is as follows:

```
[ - ] Starting online search...
[ - ] Searching for doc files, with a limit of 200
      Searching 100 results...
      Searching 200 results...
Results: 191 files found
Starting to download 5 of them:
-----

[1/5] /support/websearch/bin/answer.py?answer=186645&%20form=bb&
amp;hl=en
Error downloading /support/websearch/bin/answer.py?answer=186645&
amp;%20form=bb&hl=en
[2/5] http://www. example .com/documents/customerevidence
/27402_Cakewalk_final.doc
[3/5] http:// www. example .com/documents/customerevidence
/5588_marksspencer.doc
[4/5] http:// www. example .com/documents/uk/Ladbrokes.doc
[5/5] http:// www. example .com/~Gray/papers/PITAC_Interim_Report_8_98.doc

[ - ] Searching for pdf files, with a limit of 200
      Searching 100 results...
      Searching 200 results...
Results: 202 files found
Starting to download 5 of them:
-----

[1/5] /support/websearch/bin/answer.py?answer=186645&%20form=bb&
amp;hl=en
Error downloading /support/websearch/bin/answer.py?answer=186645&
amp;%20form=bb&hl=en
[2/5] http:// www. example .com/pubs/77954/sl021801.pdf
[3/5] http:// www. example .com/pubs/152133/deepconvexnetwork-
interspeech2011-pub.pdf
[x] Error in the parsing process
[4/5] http:// www. example .com/en-us/collaboration/papers/uruguay.pdf
[5/5] http:// www. example .com/pubs/63611/2002-droppo-icslpb.pdf

[+] List of users found:
-----
Benjamin Van Houten
Marketing
IT
```

May Yee
sarah condon
clarel
Jim Gray

[+] List of software found:

Microsoft Office Word
Microsoft Word 10.0
Microsoft Word 9.0
Microsoft Word 8.0
Acrobat Distiller 5.0.5 (Windows)
Adobe PDF Library 8.0
Adobe InDesign CS3 (5.0.2)

[+] List of paths and servers found:

'Macintosh HD:Temporary Items:AutoRecovery save of Congressio'
'NCO Server:Staff (NCO Staff):Yolanda Comedy:IR22July:IR10Aug'
'C:\jim\HPCC\PACIT_Report_8_98.doc'

[+] List of e-mails found:

gzweig@mail. example .com

You can see from the preceding result that we get a lot of information from the documents we have collected, such as the usernames and path information. We can use the obtained usernames to look for patterns in the username and for launching a brute force password attack on the usernames. But, be aware that doing a brute force password attack on an account may have the risk of locking the user accounts. The path information can be used to guess the operating system that is used by the target. We got all of this information without going to the domain website ourselves.

Metagoofil is also able to generate information in a report format. The following screenshot shows the generated report in HTML:



In the report generated, we get information about usernames, software version, e-mail address, and server information from the target domain.

Username: Palm Beach State College IP Holder **Book:** Kali Linux – Assuring Security by Penetration Testing. No part of any chapter or book may be reproduced or transmitted in any form by any means without the prior written permission for reprints and excerpts from the publisher of the book or chapter. Redistribution or other use that violates the fair use privilege under U.S. copyright laws (see 17 USC107) or that otherwise violates these Terms of Service is strictly prohibited. Violators will be prosecuted to the full extent of U.S. Federal and Massachusetts laws.

Summary

This chapter introduced you to the information gathering phase. It is usually the first phase that is done during the penetration testing process. In this phase, you collect as much information as you can about the target organization. By knowing the target organization, it will be easier when we want to attack the target. There is a Chinese proverb which says:

Know yourself, know your enemy, and you shall win a hundred battles without loss.

This saying can't be more true than in penetration testing.

We described several tools included in Kali Linux that can be used for information gathering. We started by listing several public websites that can be used to gather information about the target organization. Next, we described how to use tools to collect domain registration information. Then, we described tools that can be used to get DNS information. Later on, we explored tools for collecting routing information. In the final part of the chapter, we described tools that utilize search engine capabilities.

In the next chapter, we will discuss how to discover a target.