

Cyber Warriors: A Comprehensive Introduction to Cybersecurity Tools and Techniques

June 24-28, 2024

Murtuza Jadliwala

murtuza.jadliwala@utsa.edu



UTSA



Network Surveillance

Recap - Phase I: Reconnaissance

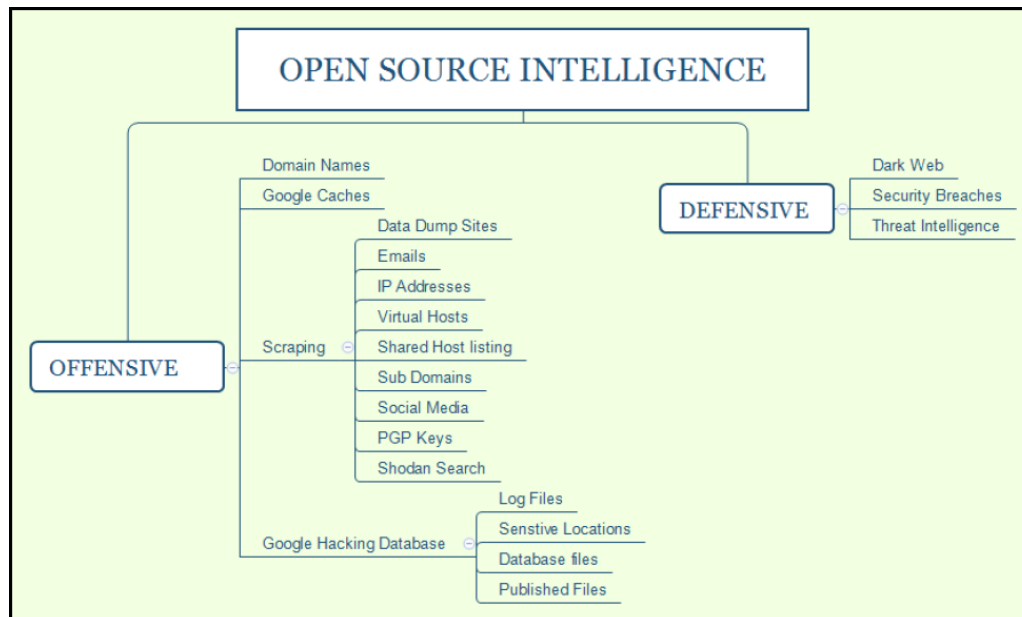
- Process of learning about of the target system, its users and the exploitable resources on that system.
- Most important phase of the attack kill chain
 - Reconnaissance is important to determine the scope of the attack, attack surface and post-exploitable actions.
- Two types of reconnaissance activities:
 - **Passive:** Does not directly interact with the target system. Could employ publicly-available or open source intelligence (OSINT). For example, web scraping. It could involve interactions that are publicly available!
 - **Active:** Collects intelligence by interacting (in a valid fashion) with the target system. For example, port scanning.
 - Passive reconnaissance is undetectable, while active reconnaissance can be detected by the target!
 - Reconnaissance phase is required for vulnerability assessment (or for determining the attack surface)!

Basic Principles of Passive Reconnaissance

- Difficult for target to differentiate passive reconnaissance from normal business activities.
 - Attacker's identity (e.g., source IP address) and activities are not logged or determined.
- Passive reconnaissance can be categorized into two types:
 - **Direct:** Normal (non-malicious) interactions with the target system/network.
 - **Indirect:** No interactions with the target system/network.

Open Source Intelligence (OSINT)

- OSINT is a form of *indirect passive reconnaissance* using only public information sources, such as Internet!
- OSINT can be categorized into two types:
 - **Offensive:** OSINT required for an attack on a target.
 - **Defensive:** OSINT of previous attacks relevant to a target for defensive purposes.



How to Collect Offensive OSINT?

A publicly (Internet) visible target's domain (list of all available sub-domain) and DNS information (routes from the attacker to the target)

- Tool 1: Sublist3r

```
root@kali:~/Sublist3r# ./sublist3r.py -d cyberhia.com

Sublist3r

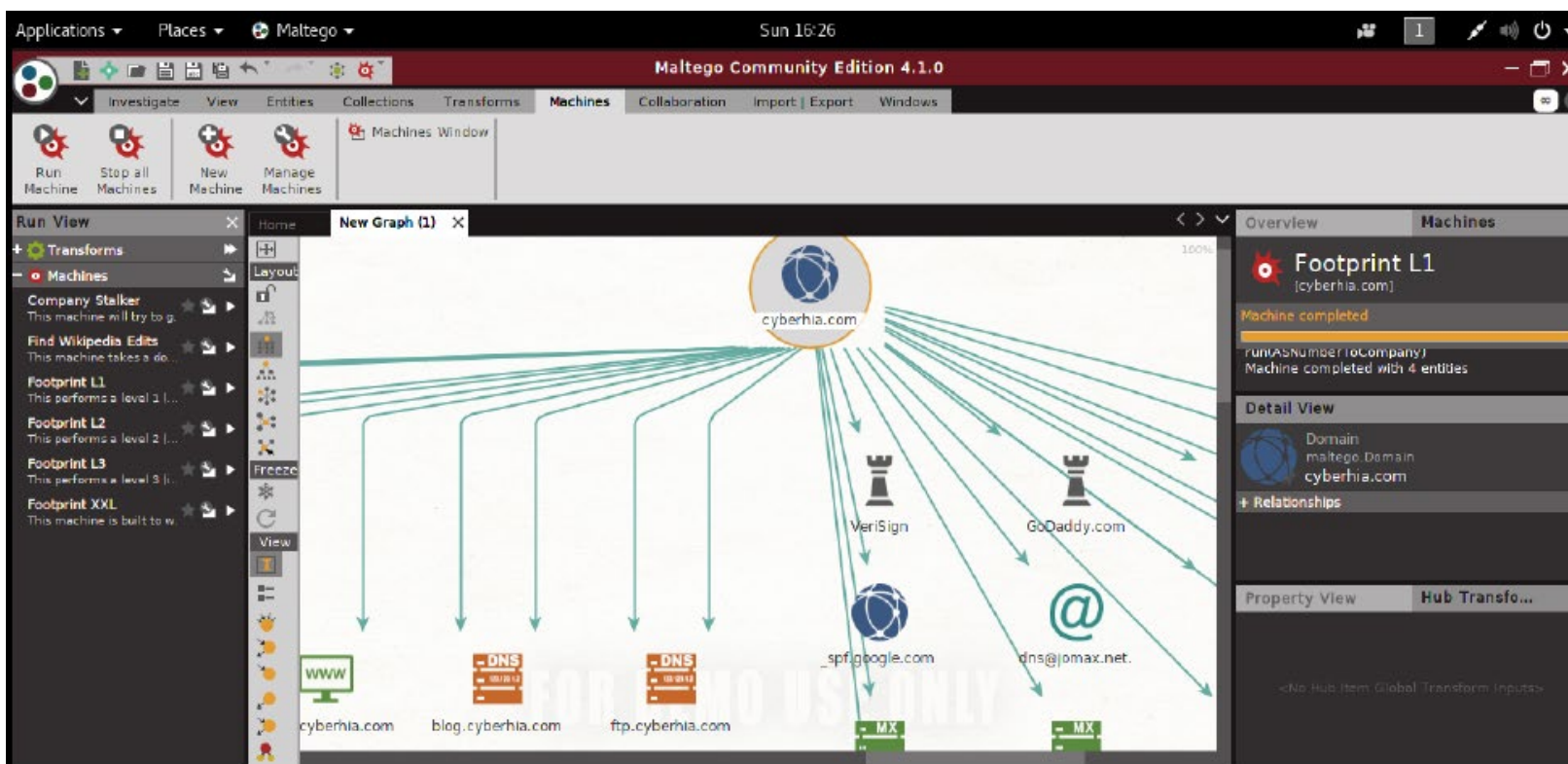
# Coded By Ahmed Aboul-Ela - @aboul31a

[-] Enumerating subdomains now for cyberhia.com
[-] Searching now in Baidu..
[-] Searching now in Yahoo..
[-] Searching now in Google..
[-] Searching now in Bing..
[-] Searching now in Ask..
[-] Searching now in Netcraft..
[-] Searching now in DNSdumpster..
[-] Searching now in Virustotal..
[-] Searching now in ThreatCrowd..
[-] Searching now in SSL Certificates..
[-] Searching now in PassiveDNS..
[-] Total Unique Subdomains Found: 3
www.cyberhia.com
blog.cyberhia.com
demo.cyberhia.com
```


How to Collect Offensive OSINT?

A publicly (Internet) visible target's domain (list of all available sub-domain) and DNS information (routes from the attacker to the target)

- Tool 2: Maltego



How to Collect Offensive OSINT?

Searching for keywords in URLs and social networking sites (Twitter, Facebook, etc.)

- Tool: OSRFramework (usufy module to search URLs, searchfy module to search social networking sites)

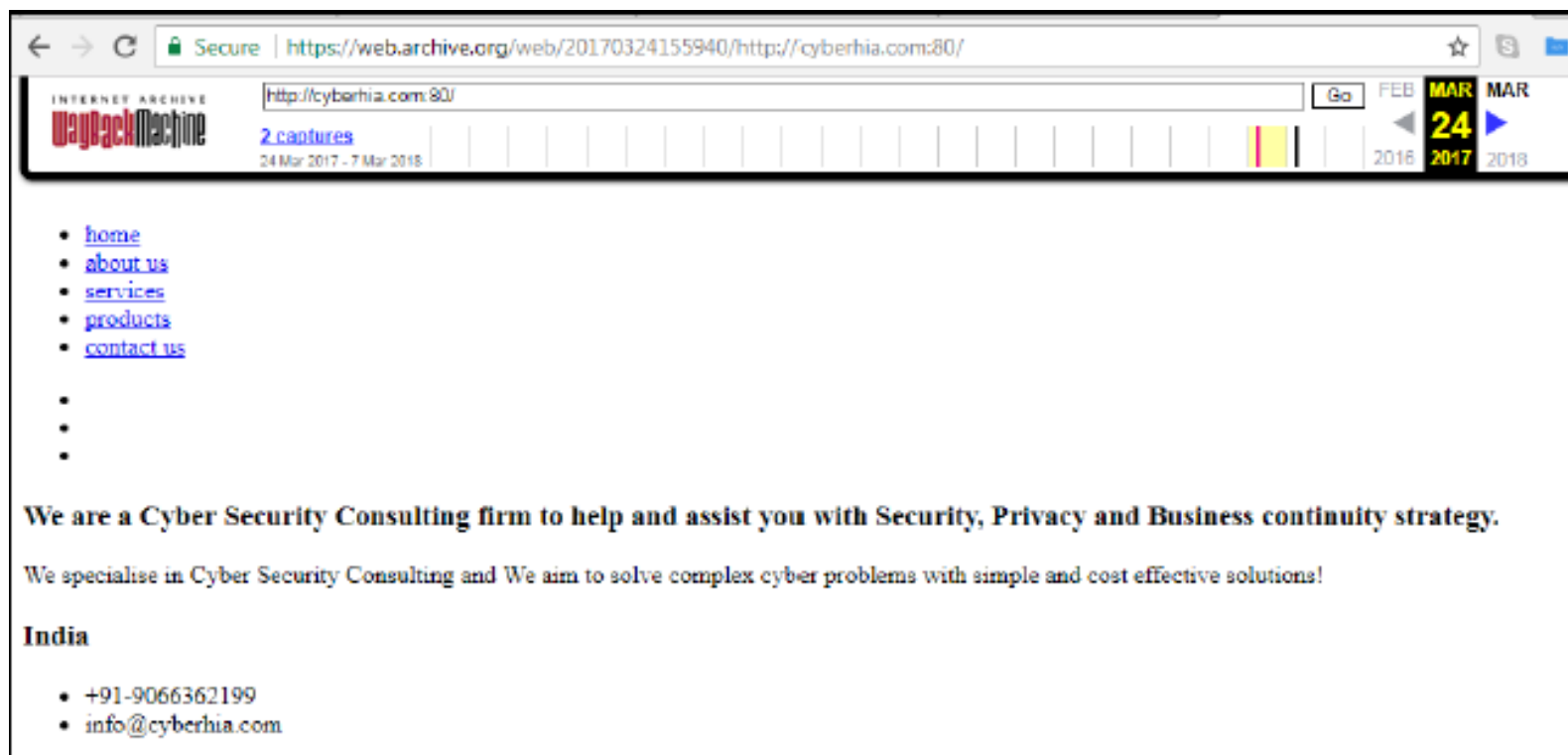
```
Sheet Name: Profiles recovered (2018-7-15_11h2m).
+-----+-----+-----+
|          i3visio_uri          | i3visio_alias | i3visio_platform |
+=====+=====+=====+
| http://twicsy.com/u/cyberhia  | cyberhia      | Twicsy           |
+-----+-----+-----+
| https://github.com/cyberhia  | cyberhia      | Github           |
+-----+-----+-----+
| https://www.freelancer.com/u/cyberhia.html | cyberhia      | Freelancer       |
+-----+-----+-----+
| https://www.facebook.com/cyberhia | cyberhia      | Facebook         |
+-----+-----+-----+
| http://realcarders.us/member.php?username=cyberhia | cyberhia      | Realcarders      |
+-----+-----+-----+
| http://twitter.com/cyberhia   | cyberhia      | Twitter          |
+-----+-----+-----+

2018-07-15 11:02:28.160567      You can find all the information here:
./profiles.csv
```


How to Collect Offensive OSINT?

Scraping: collecting sensitive information about the target from publicly accessible websites

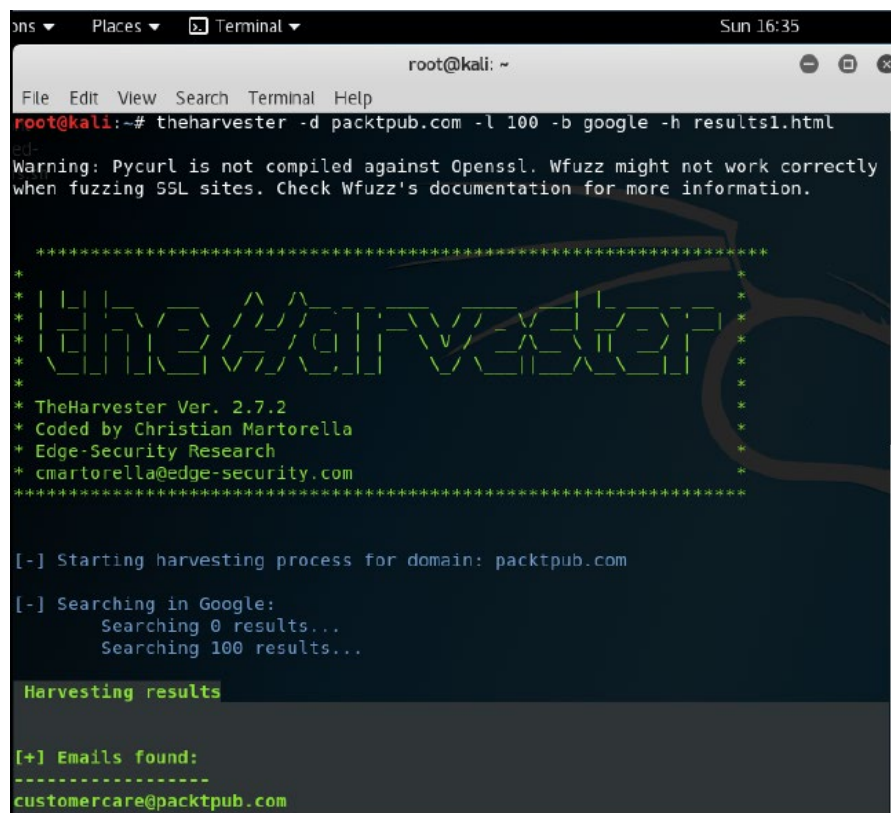
- Tool 1: <https://web.archive.org/web/>



How to Collect Offensive OSINT?

Scraping: collecting sensitive information about the target from publicly accessible websites

- Tool 2: theHarvester - A Python script that searches through popular search engines and other sites for email addresses, hosts, and sub-domains.



```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# theharvester -d packtpub.com -l 100 -b google -h results1.html
Warning: Pycurl is not compiled against Openssl. Wfuzz might not work correctly
when fuzzing SSL sites. Check Wfuzz's documentation for more information.

*****
*
*  theHarvester
*
* TheHarvester Ver. 2.7.2
* Coded by Christian Martorella
* Edge-Security Research
* cmartorella@edge-security.com
*****

[-] Starting harvesting process for domain: packtpub.com

[-] Searching in Google:
    Searching 0 results...
    Searching 100 results...

Harvesting results

[+] Emails found:
-----
customercare@packtpub.com
```

How to Collect Offensive OSINT?

How to find vulnerable hosts/machines (or specifically belonging to the target) on the Internet?

- Tool 1: Shodan: <https://www.shodan.io/>

The screenshot displays the Shodan search engine interface. The search query is 'IIS 5.0'. The results are categorized into several sections:

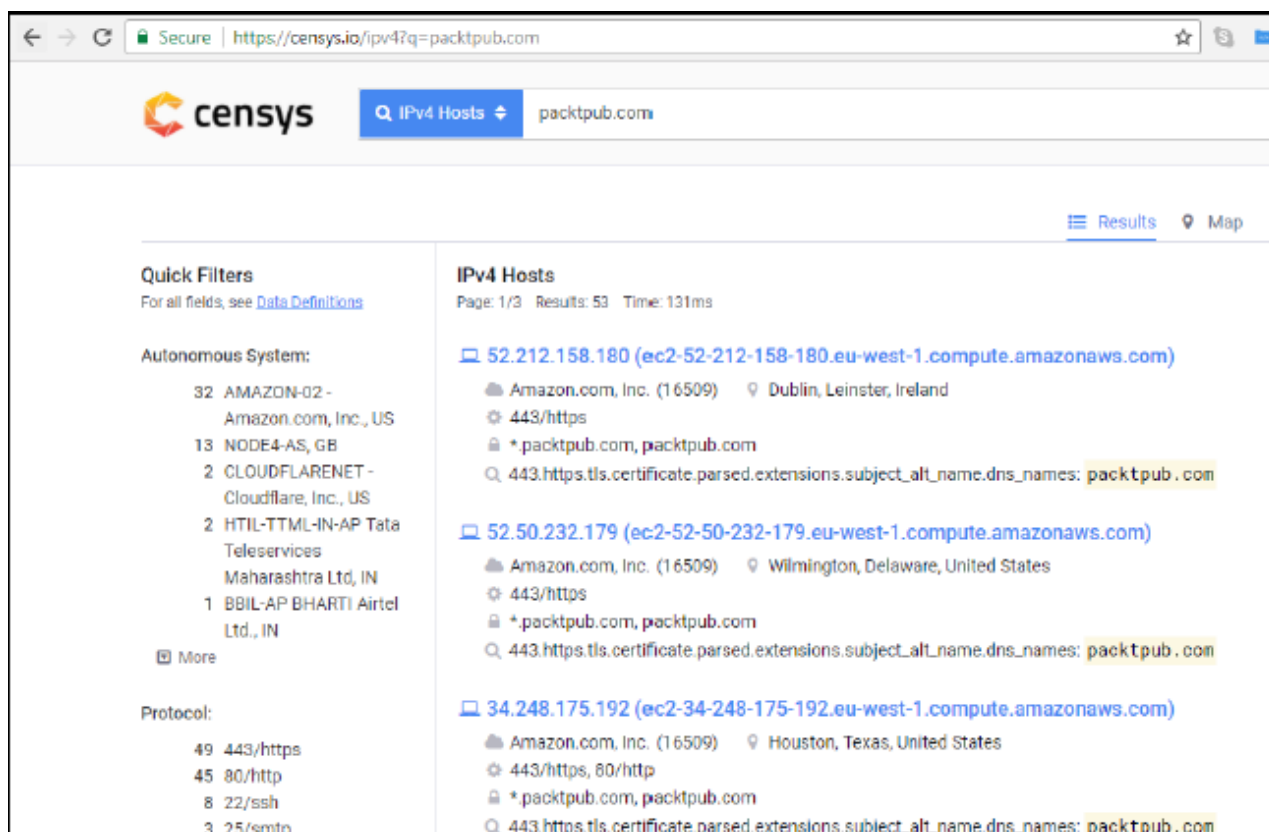
- TOTAL RESULTS:** 44,205
- TOP COUNTRIES:** A world map showing the distribution of results by country. The United States has the highest number of results (14,840), followed by Korea, Republic of (4,518), China (4,193), Canada (3,293), and Japan (1,593).
- TOP SERVICES:** A table showing the distribution of results by service type:

Service	Count
HTTP	32,976
HTTPS	5,778
HTTP (8080)	1,217
HTTP (81)	443
8081	289
- En construcción:** A result for a website in Spain (213.00.109.24) with a Microsoft IIS 5.0 server. The page is under construction.
- Equine Indemnity:** A result for a website in the United Kingdom (194.51.165.122) with a GoDaddy Secure SSL Certificate. The page is under construction.

How to Collect Offensive OSINT?

How to find vulnerable hosts/machines (or specifically belonging to the target) on the Internet?

- Tool 1: Censys: <https://censys.io/>



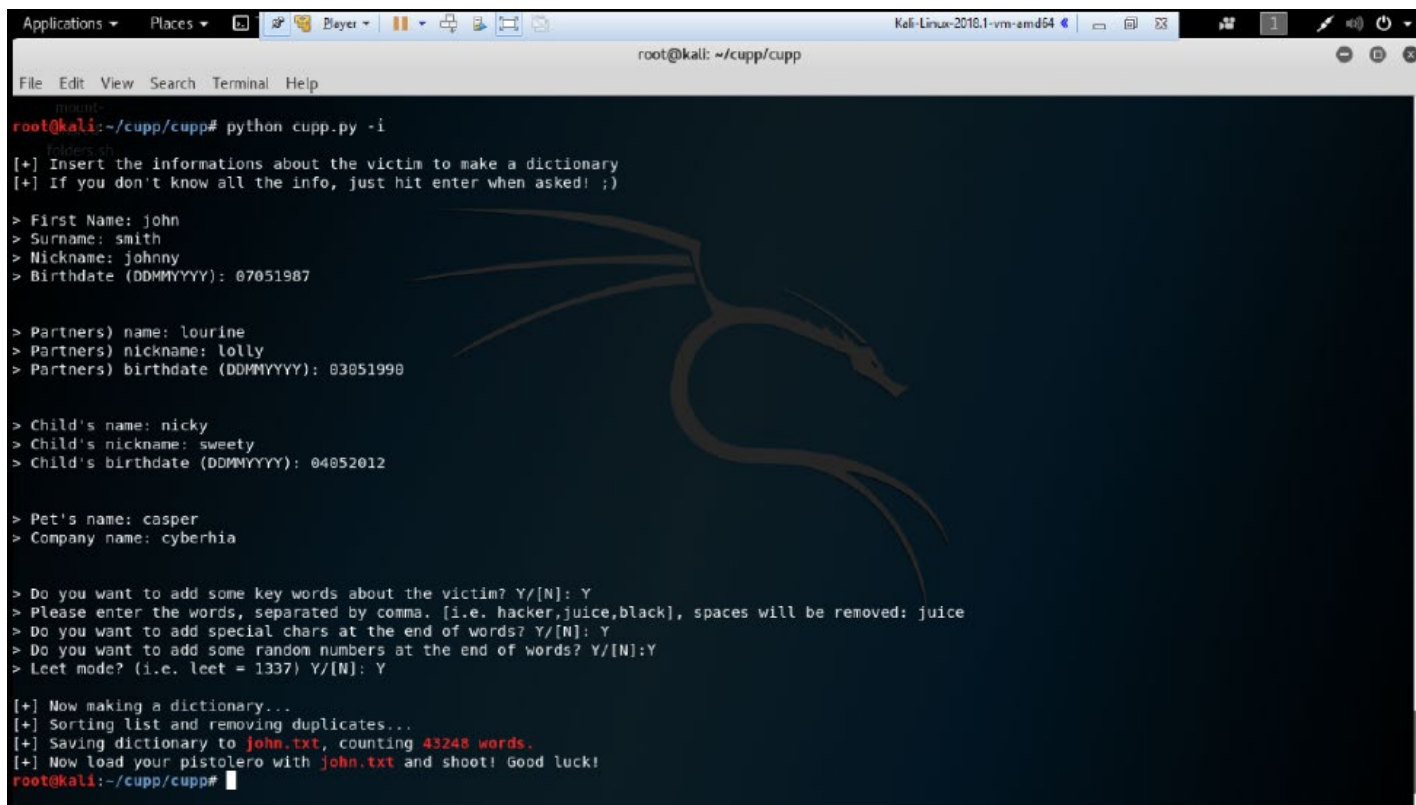
The screenshot shows the Censys website interface. The browser address bar displays the URL <https://censys.io/ipv4?q=packtpub.com>. The search bar contains the text "IPv4 Hosts" and "packtpub.com". The page displays search results for IPv4 hosts. On the left, there are "Quick Filters" for "Autonomous System" and "Protocol". The "Autonomous System" filter lists: 32 AMAZON-02 - Amazon.com, Inc., US; 13 NODE4-AS, GB; 2 CLOUDFLARENET - Cloudflare, Inc., US; 2 HTIL-TTML-IN-AP Tata Teleservices Maharashtra Ltd, IN; and 1 BBIL-AP BHARTI Airtel Ltd., IN. The "Protocol" filter lists: 49 443/https; 45 80/http; 8 22/ssh; and 3 25/smtp. The main content area shows "IPv4 Hosts" with 53 results. Three results are visible:

- 52.212.158.180 (ec2-52-212-158-180.eu-west-1.compute.amazonaws.com)**
 - Amazon.com, Inc. (16509) Dublin, Leinster, Ireland
 - 443/https
 - *.packtpub.com, packtpub.com
 - 443.https.tls.certificate.parsed.extensions.subject_alt_name.dns_names: packtpub.com
- 52.50.232.179 (ec2-52-50-232-179.eu-west-1.compute.amazonaws.com)**
 - Amazon.com, Inc. (16509) Wilmington, Delaware, United States
 - 443/https
 - *.packtpub.com, packtpub.com
 - 443.https.tls.certificate.parsed.extensions.subject_alt_name.dns_names: packtpub.com
- 34.248.175.192 (ec2-34-248-175-192.eu-west-1.compute.amazonaws.com)**
 - Amazon.com, Inc. (16509) Houston, Texas, United States
 - 443/https, 80/http
 - *.packtpub.com, packtpub.com
 - 443.https.tls.certificate.parsed.extensions.subject_alt_name.dns_names: packtpub.com

How to Collect Offensive OSINT?

How to create custom password lists/dictionaries for cracking?

- Tool 1: **Common User Password Profiler (CUPP)** is a python script that allows the tester to generate a wordlist that is specific to a particular user.



```
root@kali: ~/cupp/cupp
File Edit View Search Terminal Help
root@kali:~/cupp/cupp# python cupp.py -i
[+] Insert the informations about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)

> First Name: john
> Surname: smith
> Nickname: johnny
> Birthdate (DDMMYYYY): 07051987

> Partners) name: lourine
> Partners) nickname: lolly
> Partners) birthdate (DDMMYYYY): 03051990

> Child's name: nicky
> Child's nickname: sweety
> Child's birthdate (DDMMYYYY): 04052012

> Pet's name: casper
> Company name: cyberhia

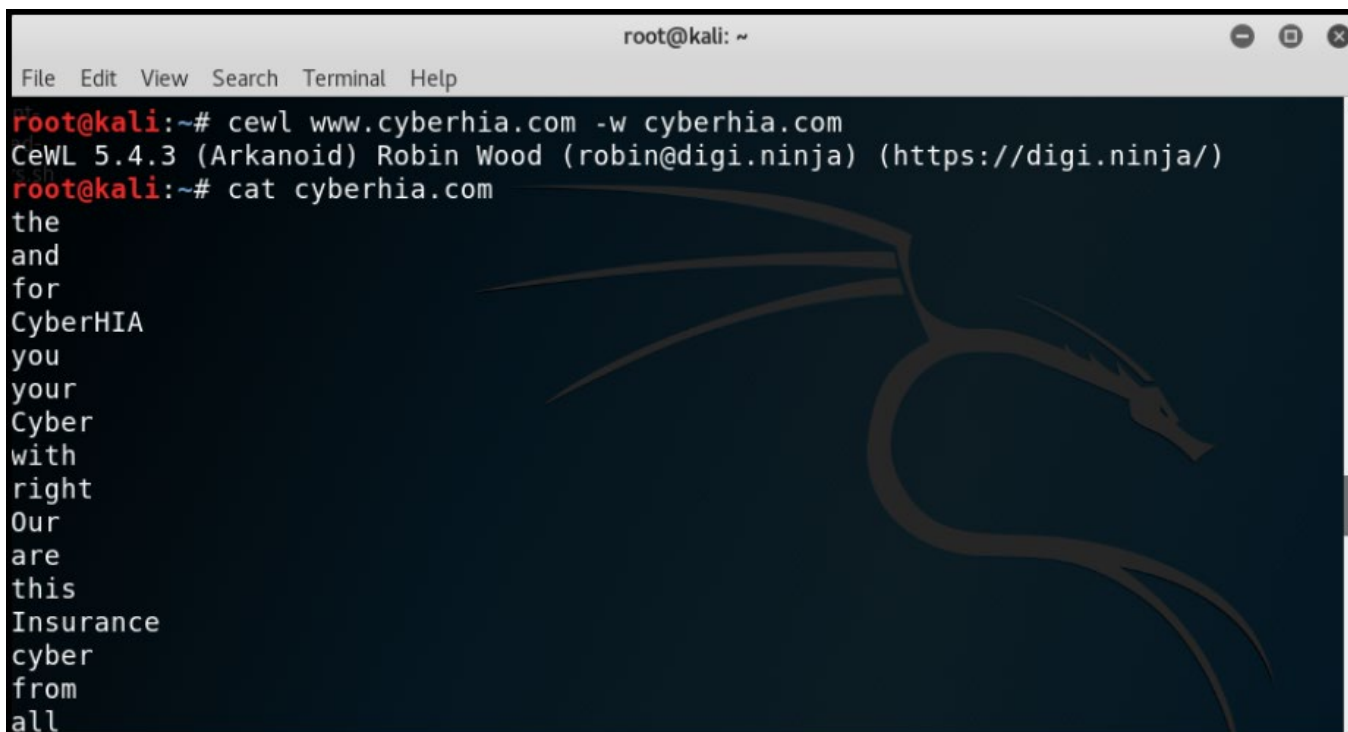
> Do you want to add some key words about the victim? Y/[N]: Y
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces will be removed: juice
> Do you want to add special chars at the end of words? Y/[N]: Y
> Do you want to add some random numbers at the end of words? Y/[N]:Y
> Leet mode? (i.e. leet = 1337) Y/[N]: Y

[+] Now making a dictionary...
[+] Sorting list and removing duplicates...
[+] Saving dictionary to john.txt, counting 43248 words.
[+] Now load your pistolero with john.txt and shoot! Good luck!
root@kali:~/cupp/cupp#
```

How to Collect Offensive OSINT?

How to create custom password lists/dictionaries for cracking?

- Tool 2: **CeWL** (Custom Word List Generator) is a Ruby app that spiders a given URL to a specified depth, optionally following external links, and returns a list of words that can then be used for password crackers such as **John the Ripper**.

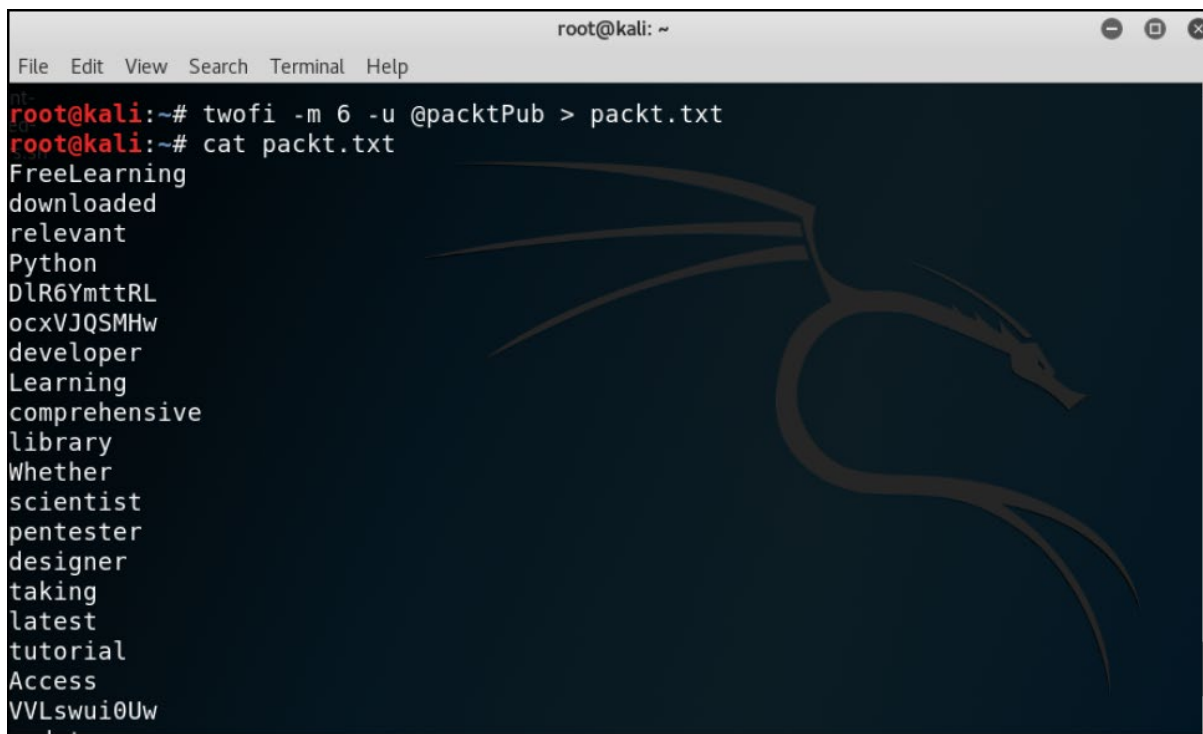
A terminal window titled 'root@kali: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the execution of the CeWL command: 'cewl www.cyberhia.com -w cyberhia.com'. The output of the command is displayed, showing a list of words extracted from the website. A large, stylized dragon logo is visible in the background of the terminal window.

```
root@kali:~# cewl www.cyberhia.com -w cyberhia.com
CeWL 5.4.3 (Arkanoid) Robin Wood (robin@digi.ninja) (https://digi.ninja/)
root@kali:~# cat cyberhia.com
the
and
for
CyberHIA
you
your
Cyber
with
right
Our
are
this
Insurance
cyber
from
all
```


How to Collect Offensive OSINT?

How to create custom password lists/dictionaries for cracking?

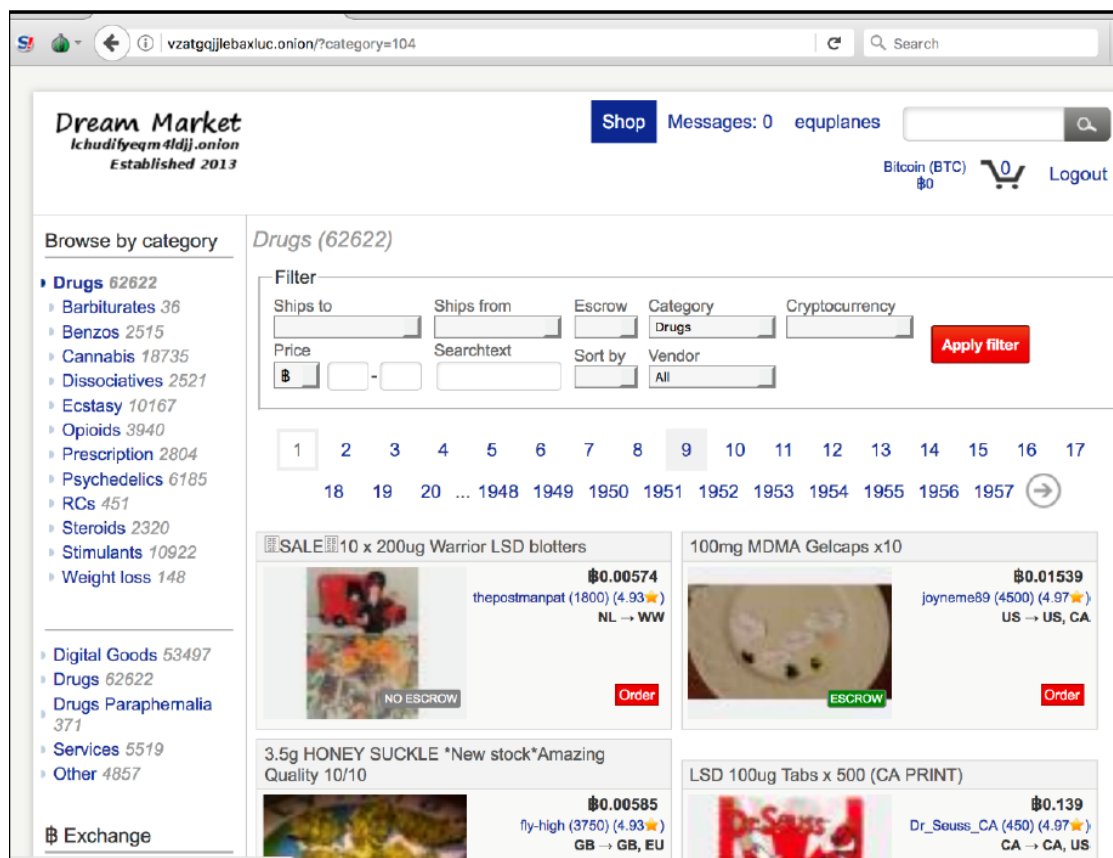
- Tool 3: **Twofi** (Twitter words of interest) is a tool written in Ruby script used to profile a user that has a social media account such as Twitter. It utilizes the Twitter API to generate a custom list of words (from a user's Twitter feed) that can be utilized for offline password cracking.

A terminal window titled 'root@kali: ~' with a menu bar (File, Edit, View, Search, Terminal, Help) and window control buttons. The terminal shows the execution of the 'twofi' tool to generate a password list from a Twitter user's feed. The command 'twofi -m 6 -u @packtPub > packt.txt' is run, followed by 'cat packt.txt' which displays a list of words extracted from the feed. A large, faint dragon logo is visible in the background of the terminal window.

```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# twofi -m 6 -u @packtPub > packt.txt
root@kali:~# cat packt.txt
FreeLearning
downloaded
relevant
Python
DlR6YmttRL
ocxVJQSMHw
developer
Learning
comprehensive
library
Whether
scientist
pentester
designer
taking
latest
tutorial
Access
VVLswui0Uw
```

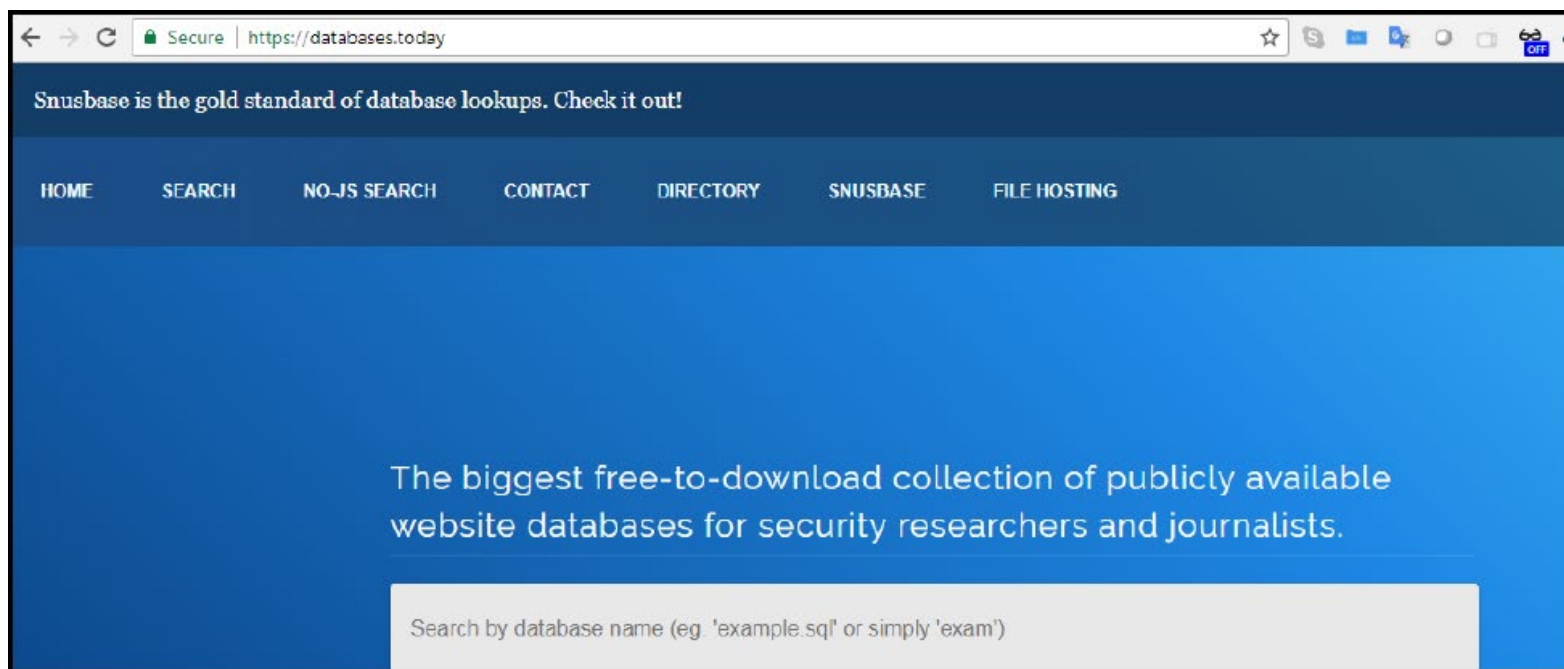
How to Collect Defensive OSINT?

- Source 1: Darkweb (e.g., service such as Dream Market)
 - A lot of such services operate as TOR services!



How to Collect Defensive OSINT?

- Source 2: Publicly visible websites that have an archive of breached data from compromised hosts or more information about such hosts.
 - E.g., <https://databases.today>, <https://haveibeenpwned.com>, <http://zone-h.com>.

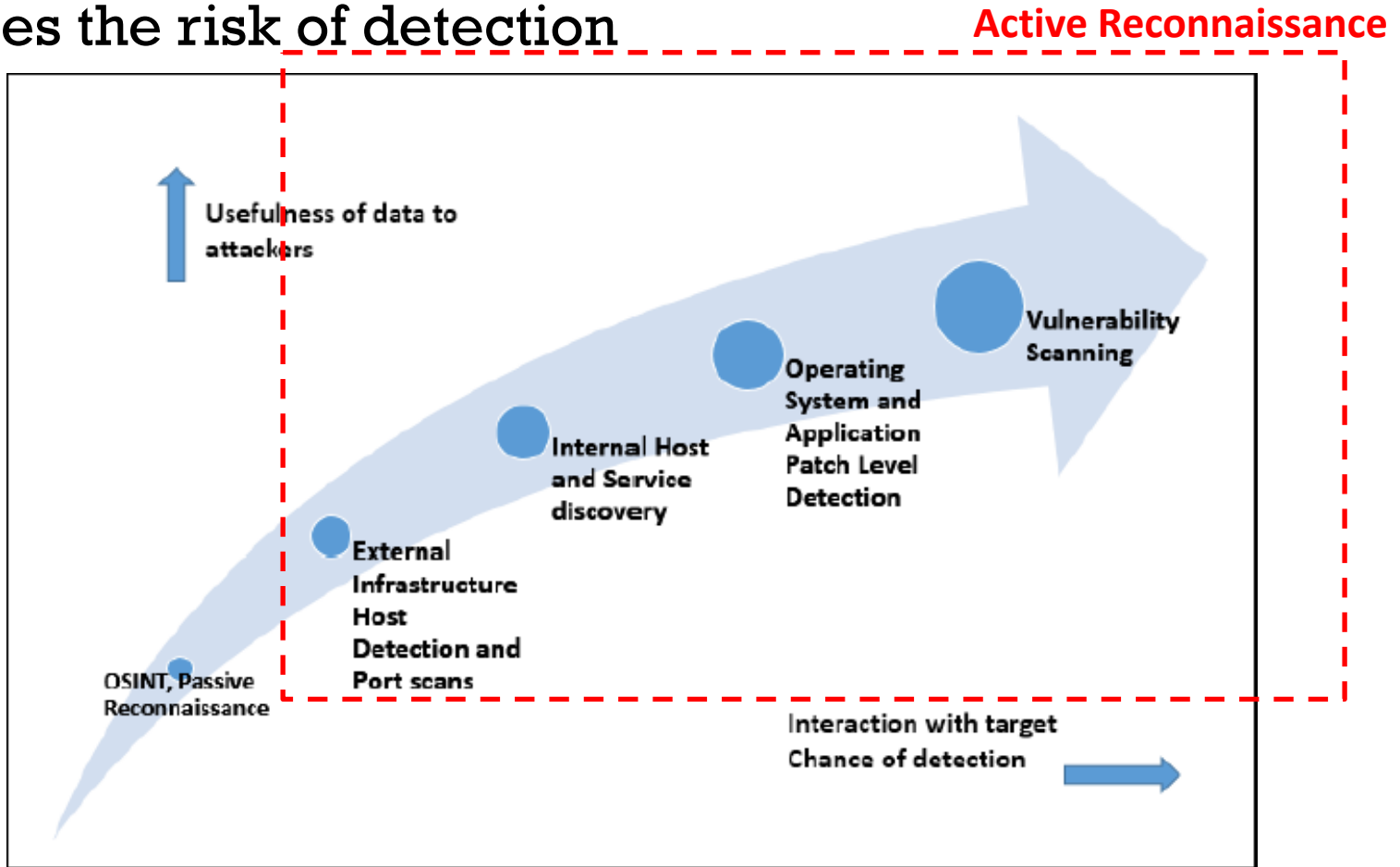


Recap - Phase I: Reconnaissance

- Process of learning about of the target system, its users and the exploitable resources on that system.
- Most important phase of the attack kill chain
 - Reconnaissance is important to determine the scope of the attack, attack surface and post-exploitable actions.
- Two types of reconnaissance activities:
 - **Passive:** Does not directly interact with the target system. Could employ publicly-available or open source intelligence (OSINT). For example, web scraping. It could involve interactions that are publicly available!
 - **Active:** collects intelligence by interacting (in a valid fashion) with the target system. For example, port scanning.
 - Passive reconnaissance is undetectable, while active reconnaissance can be detected by the target!
 - Reconnaissance phase is required for vulnerability assessment (or for determining the attack surface)!

Basic Principles of Active Reconnaissance

- As usefulness of the data to the attacker increases, so does the risk of detection



Stealthy Scanning

- Why does active reconnaissance (scanning of the target) need to be stealthy?
 - The greatest risk of conducting active reconnaissance is the discovery by the target.
 - Using information such as time stamps and payload data of scans, the source IP address/Port numbers, the target can not only determine if active reconnaissance is occurring but also identify the source of the incoming reconnaissance.
 - Stealth techniques are needed to minimize the detection chances by the target.
- How can stealth be achieved?
 - **Camouflage tool signatures** to avoid detection and triggering an alarm.
 - **Hide the active scan** within legitimate traffic.
 - **Modify the active scan** to hide the source and type of traffic.
 - **Make the active scan** invisible using nonstandard traffic types or encryption.
- Stealthy scanning strategies could include:
 - Adjusting source IP stack and tool identification settings.
 - Modifying packet parameters (nmap).
 - Using proxies with anonymity networks (ProxyChains and the Tor network).

Stealthy Scanning – Strategy 1

Modifying packet parameters

- A popular approach to active reconnaissance is to conduct a scan against the target by sending carefully crafted packets to the target and using the returned packets to gain information. Commonly used tool of accomplishing this is **Network Mapper** (nmap).
- Some stealth techniques while using nmap include:
 - To perform a targeted scan. For example, in order to confirm the presence of a web host, scan on port 80 (the default port for web-based services).
 - Try to avoid scans that may connect (e.g., TCP connection) with the target system and leak data.
 - Randomize or spoof packet settings, such as the source IP and port address, and the MAC address.
 - Adjust the timing to slow the arrival of packets at the target site.
 - Change the packet size by fragmenting packets or appending random data to

```
# nmap --spooof-mac Cisco --data-length 24 -T paranoid --max-hostgroup 1 --  
max-parallelism 10 -Pn -f -D 10.1.20.5,RND:5,ME -v -n -sS -sV -oA  
/desktop/pentest/nmap/out -p T:1-1024 --randomize-hosts 10.1.1.10 10.1.1.15
```

Stealthy Scanning – Strategy 2

Using proxies with anonymity networks

- In order to hide the source of the scan (i.e., identify of the scanner), proxy servers or an anonymity network such as TOR can be used.
- Tor (www.torproject.org) is an open source implementation of the third-generation onion routing that provides free access to an anonymous proxy network. Onion routing enables online anonymity by encrypting user traffic and then transmitting it through a series of onion routers.
- A web proxy can be used on the way to the TOR network for further anonymity.
- See the textbook for details on how to install TOR and list proxies to use (by editing the `proxychains.conf` file)

Objectives of Active Reconnaissance/Scan

- Active reconnaissance involves determination of **one or more** the following information related to the target system/network:
 - Domain ownership information (DNS records).
 - Host and subnet addresses (IPv4 and IPv6 addresses).
 - Identify geographical locations of (and other information related to) target hosts and individuals.
 - Identifying the presence of Load Balancers, Firewalls and Intrusion Detection Systems/Intrusion Prevention Systems.
 - Identifying running services, open ports, and operating systems on the target host(s).

Domain Ownership and DNS Info

```
root@kali:~# whois cyberhia.com
Domain Name: CYBERHIA.COM
Registry Domain ID: 1954580299_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.godaddy.com
Registrar URL: http://www.godaddy.com
Updated Date: 2018-07-28T11:48:19Z
Creation Date: 2015-08-22T04:14:35Z
Registry Expiry Date: 2018-08-22T04:14:35Z
Registrar: GoDaddy.com, LLC
Registrar IANA ID: 146
Registrar Abuse Contact Email: abuse@godaddy.com
Registrar Abuse Contact Phone: 480-624-2505
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhi
bited
Domain Status: clientRenewProhibited https://icann.org/epp#clientRenewProhibi
ted
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferP
rohibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhi
bited
Name Server: NS17.DOMAINCONTROL.COM
Name Server: NS18.DOMAINCONTROL.COM
DNSSEC: unsigned
```

Host & Subnet Addresses

- Host and subnet Addresses is another important information that can be obtained from DNS records.
 - Besides the whois command, many other Kali tools available for determining addressing information (for both IPv4 and IPv6 addressing schemes) related to the target.

Application	Description
<code>dnsenum</code> , <code>dnsmmap</code> , and <code>dnsrecon</code>	These are comprehensive DNS scanners—DNS record enumeration (A, MX, TXT, SOA, wildcard, and so on), subdomain brute-force attacks, Google lookup, reverse lookup, zone transfer, and zone walking. <code>dnsrecon</code> is usually the first choice—it is highly reliable, results are well parsed, and data can be directly imported into the Metasploit framework.
<code>dnstracer</code>	This determines where a given DNS gets its information from, and follows the chain of DNS servers back to the servers that know the data.
<code>dnswalk</code>	This DNS debugger checks specified domains for internal consistency and accuracy.
<code>fierce</code>	This locates non-contiguous IP space and hostnames against specified domains by attempting zone transfers and then attempting brute-force attacks to gain DNS information.

Geographic Location & Other Info

- The recon-ng framework and its modules, written in Python, can be used to conduct a variety of reconnaissance activities such as:
 - Harvest contacts using whois, Jigsaw, LinkedIn, and Twitter (use the **mangle** module to extract and present email data).
 - Identify hosts.
 - Identify geographical locations of hosts and individuals using **hostop**, **ipinfodb**, **maxmind**, **uniapple**, and **wiggle**.
 - Identify host information using **netcraft** and related modules.
 - Identify account and password information that has previously been compromised and leaked onto the Internet (the **pwnedlist** modules, **wascompanyhacked**, **xssed**, and **punkspider**)

Identifying Load Balancers, Firewalls & IDS

- Many popular web services employ load balancers to efficiently distribute HTTP requests.
 - Relieves load on a single webserver.
 - Provides faster, geographical location-based responses.
- Load balancers can inadvertently also confuse hackers/attackers, and it is in their best interest to detect such devices on the target's edge.
- lbd is a popular Kali tool that detects both DNS and HTTP-based load balancing.

```
root@kali:~# lbd www.████████.com

lbd - load balancing detector 0.4 - Checks if a given domain uses load-balancing.
      Written by Stefan Behte (http://ge.mine.nu)
      Proof-of-concept! Might give false positives.

Checking for DNS-Loadbalancing: NOT FOUND
Checking for HTTP-Loadbalancing [Server]:

NOT FOUND

Checking for HTTP-Loadbalancing [Date]: 16:33:49, 16:33:49, 16:33:49, 16:33:49, 16:33:49, 1
:33:50, 16:33:50, 16:33:50, 16:33:50, 16:33:50, 16:33:50, 16:33:50, 16:33:50, 16:33:51, 16:
3:51, 16:33:51, 16:33:51, 16:33:51, 16:33:51, 16:33:52, 16:33:52, 16:33:52, 16:33:52, 16:33
52, 16:33:52, 16:33:53, 16:33:53, 16:33:53, 16:33:53, 16:33:53, 16:33:53, 16:33:53, 16:33:5
, 16:33:54, 16:33:54, 16:33:54, 16:33:54, 16:33:54, NOT FOUND

Checking for HTTP-Loadbalancing [Diff]: FOUND
< X-FB-Debug: 7QIJSA6gveuWk7MayNx68HnFO3VstBsjsT/xfZ3C3bg7uxUDmCDhhu399VjLbn3FaP+uPMqO2TBHC
> X-FB-Debug: E2tJlH38PTVAcLkmE7qJIjcb9tmOBXJgyRB01jgKdHkBiBAjZ1bMDG41VTHBkUM4B1EuoA8LmJ49k

www.████████.com does Load-balancing. Found via Methods: HTTP[Diff]
```

Service Discovery

- Accomplished using Port Scanning - process of connecting to TCP and UDP ports to determine what services and applications are running on the target host/device.
- Ports are software interfaces (doors) that network applications use to communicate. There are 65,535 unique port numbers (for both TCP and UDP applications). First 1,024 port numbers are reserved for well-known services (for instance, TCP 20 and 21 are the usual ports for the File Transfer Protocol or FTP service).
- Kali tools like nmap and nc (netcat) are popular port scanning tools.

```
root@kali:~# while read r; do nc -v -z $r 1-65535; done < iplist
dlinkrouter [192.168.0.1] 56209 (?) open
dlinkrouter [192.168.0.1] 49152 (?) open
dlinkrouter [192.168.0.1] 45555 (?) open
dlinkrouter [192.168.0.1] 8183 (?) open
dlinkrouter [192.168.0.1] 8182 (?) open
dlinkrouter [192.168.0.1] 8181 (?) open
dlinkrouter [192.168.0.1] 7777 (?) open
dlinkrouter [192.168.0.1] 4433 (?) open
dlinkrouter [192.168.0.1] 443 (https) open
dlinkrouter [192.168.0.1] 80 (http) open
dlinkrouter [192.168.0.1] 53 (domain) open
DNS fwd/rev mismatch: kali != kali.secure
kali [192.168.0.124] 55982 (?) open
kali [192.168.0.124] 33658 (?) open
kali [192.168.0.124] 8000 (?) open
kali [192.168.0.124] 22 (ssh) open
```

Operating System Fingerprinting

- Two types of scans can be used to determine the operating system (OS) of a remote system:
 - **Active fingerprinting:** The attacker sends normal and malformed packets to the target and records the response, which can be used to determine the OS.
 - **Passive fingerprinting:** The attacker passively records (or sniffs) and analyzes the packet stream (from the target host) to determine the OS from packet characteristics/attributes.
- Active fingerprinting is faster and more accurate, but easily detectable, than passive fingerprinting. In Kali, the two primary active tools are nmap and xprobe2.
- The nmap tool can be used with the -O flag to determine the OS as shown below:
 - `nmap -sS -O target.com`
- Similarly, the xprobe2 can also be used to identify the OS as shown below. However, xprobe2 does not deterministically identify the OS, rather it is assigned the probability of being one of several possible variants.
 - `xprobe2 www.target.com`
- As these fingerprinting tools rely on packet settings, such as time-to-live or initial windows sizes, changes to these values or other user-configurable settings can change results output by them.