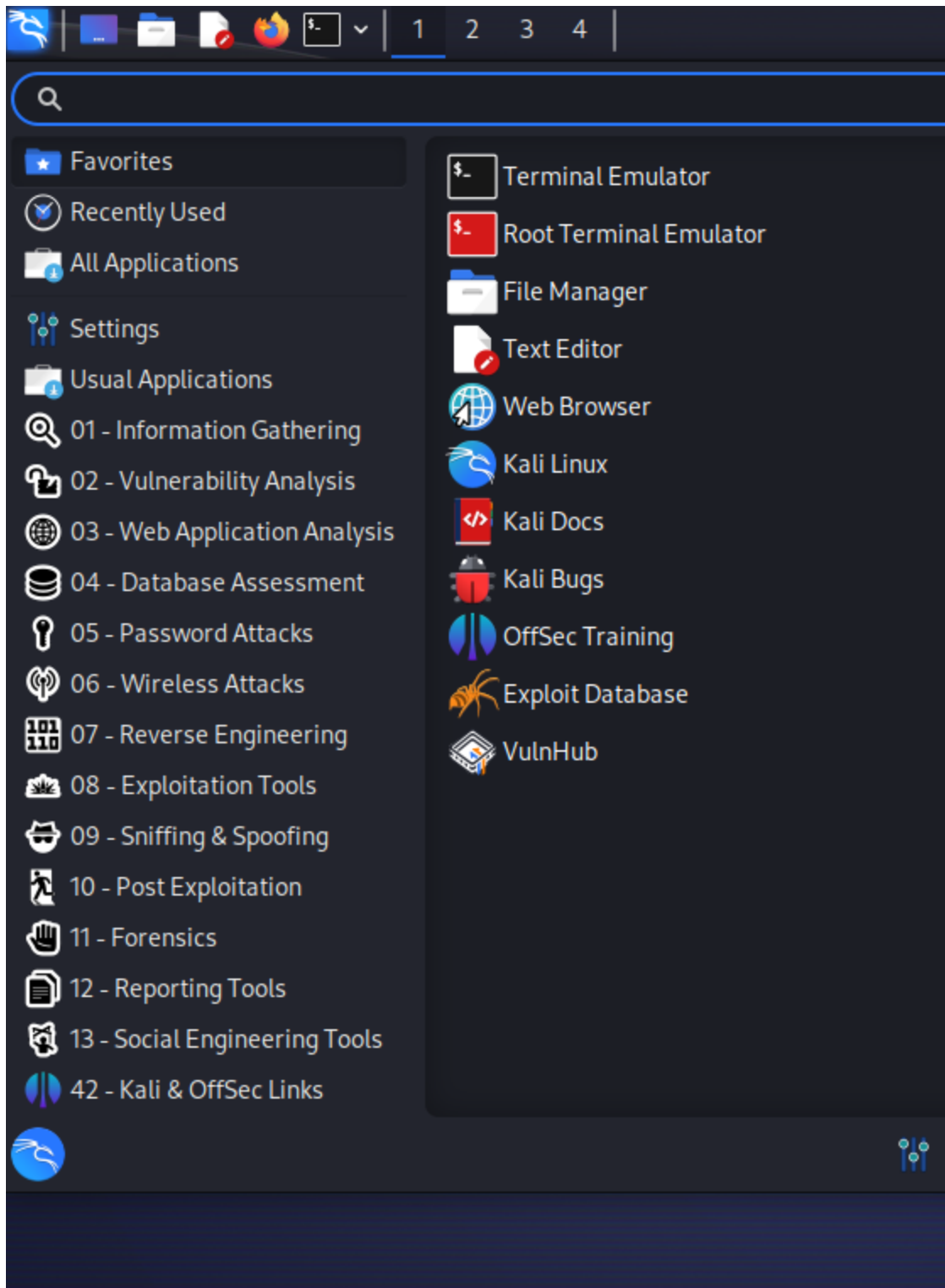


Week 2 - Assignment

Dev Mehta - 21BCE2888

The top 10 Kali Linux tools commonly used for cybersecurity:

Kali Linux is one of the best OS for Cyber Security Training. We can also use it to scan smaller scale websites or for personal projects. A popular use case is to use it for various bug bounty programs. It has various tools which we can find at:



1. Nmap (Network Mapper):

Nmap is a powerful open-source tool for network discovery and security auditing. It is used for network scanning to find open ports, services running on remote machines,

and potential vulnerabilities.

```
(kali㉿kali)-[~]
└─$ nmap -v -A -sV 192.168.1.1
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-05 09:29 EDT
NSE: Loaded 156 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 09:29
Completed NSE at 09:29, 0.00s elapsed
Initiating NSE at 09:29
Completed NSE at 09:29, 0.00s elapsed
Initiating NSE at 09:29
Completed NSE at 09:29, 0.00s elapsed
Initiating Ping Scan at 09:29
Scanning 192.168.1.1 [2 ports]
Completed Ping Scan at 09:29, 0.02s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 09:29
Completed Parallel DNS resolution of 1 host. at 09:29, 0.01s elapsed
Initiating Connect Scan at 09:29
Scanning 192.168.1.1 [1000 ports]
Discovered open port 80/tcp on 192.168.1.1
Discovered open port 443/tcp on 192.168.1.1
Discovered open port 53/tcp on 192.168.1.1
Completed Connect Scan at 09:29, 4.94s elapsed (1000 total ports)
Initiating Service scan at 09:29
Scanning 3 services on 192.168.1.1
Completed Service scan at 09:29, 16.03s elapsed (3 services on 1 host)
NSE: Script scanning 192.168.1.1.
Initiating NSE at 09:29
Completed NSE at 09:29, 8.43s elapsed
Initiating NSE at 09:29
Completed NSE at 09:29, 0.28s elapsed
Initiating NSE at 09:29
Completed NSE at 09:29, 0.00s elapsed
Nmap scan report for 192.168.1.1
Host is up (0.014s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
53/tcp    open  domain       (unknown banner: [secured])
| dns-nsid:
|_  bind.version: [secured]
| fingerprint-strings:
|   DNSVersionBindReqTCP:
|     version
|     bind
|_   [secured]
80/tcp    open  http
|_ http-title: F612W
|_ http-server-header: <empty>
| fingerprint-strings:
|   GetRequest:
```

In the example, we are trying to check IP-192.168.1.1 and we can what ports are open.

2. Metasploit Framework:

Metasploit is a penetration testing framework that helps security professionals find, exploit, and validate vulnerabilities in systems and networks. It provides a large collection of exploits and payloads.

```
msf6 > search portscan

Matching Modules

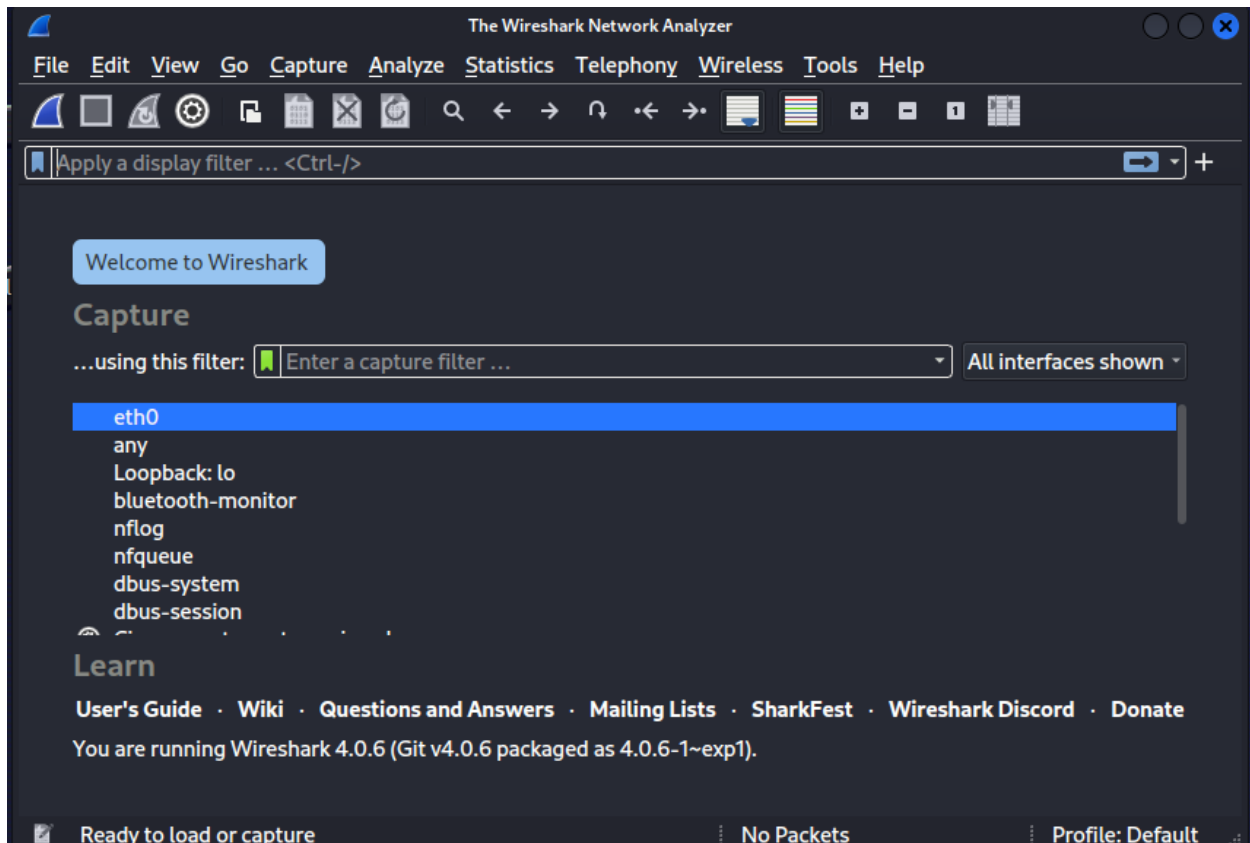
#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/scanner/portscan/ftpbounce      normal         No    FTP Bounce Port Scanner
1  auxiliary/scanner/natpmp/natpmp_portscan  normal         No    NAT-PMP External Port Scann
er
2  auxiliary/scanner/sap/sap_router_portscanner  normal         No    SAPRouter Port Scanner
3  auxiliary/scanner/portscan/xmas           normal         No    TCP "XMas" Port Scanner
4  auxiliary/scanner/portscan/ack            normal         No    TCP ACK Firewall Scanner
5  auxiliary/scanner/portscan/tcp            normal         No    TCP Port Scanner
6  auxiliary/scanner/portscan/syn            normal         No    TCP SYN Port Scanner
7  auxiliary/scanner/http/wordpress_pingback_access  normal         No    Wordpress Pingback Locator

Interact with a module by name or index. For example info 7, use 7 or use auxiliary/scanner/http/wordpress_pingback_access
```

In the example, we are able to portscan!

3. Wireshark:

Wireshark is a widely used network protocol analyzer. It allows you to capture and inspect data packets on a network, helping with network troubleshooting and security analysis.



The WireShark Tool in Kali Linux, which can be used to get information on incoming and outgoing data packets.

4. Aircrack-ng:

Aircrack-ng is a set of tools for auditing wireless networks. It includes utilities for capturing packets, cracking WEP and WPA/WPA2 keys, and assessing the security of Wi-Fi networks.

File Actions Edit View Help

\$ aircrack-ng --help

Aircrack-ng 1.7 - (C) 2006-2022 Thomas d'Otreppe
<https://www.aircrack-ng.org>

usage: aircrack-ng [options] <input file(s)>

Common options:

-a <amode> : force attack mode (1/WEP, 2/WPA-PSK)
-e <essid> : target selection: network identifier
-b <bssid> : target selection: access point's MAC
-p <nbcpu> : # of CPU to use (default: all CPUs)
-q : enable quiet mode (no status output)
-C <macs> : merge the given APs to a virtual one
-l <file> : write key to file. Overwrites file.

Static WEP cracking options:

-c : search alpha-numeric characters only
-t : search binary coded decimal chr only
-h : search the numeric key for Fritz!BOX
-d <mask> : use masking of the key (A1:XX:CF:YY)
-m <maddr> : MAC address to filter usable packets
-n <nbits> : WEP key length : 64/128/152/256/512
-i <index> : WEP key index (1 to 4), default: any
-f <fudge> : bruteforce fudge factor, default: 2
-k <korek> : disable one attack method (1 to 17)
-x or -x0 : disable bruteforce for last keybytes
-x1 : last keybyte bruteforcing (default)
-x2 : enable last 2 keybytes bruteforcing
-X : disable bruteforce multithreading
-y : experimental single bruteforce mode
-K : use only old KoreK attacks (pre-PTW)
-s : show the key in ASCII while cracking
-M <num> : specify maximum number of IVs to use
-D : WEP decloak, skips broken keystreams
-P <num> : PTW debug: 1: disable Klein, 2: PTW
-1 : run only 1 try to crack key with PTW
-V : run in visual inspection mode

WEP and WPA-PSK cracking options:

-w <words> : path to wordlist(s) filename(s)
-N <file> : path to new session filename
-R <file> : path to existing session filename

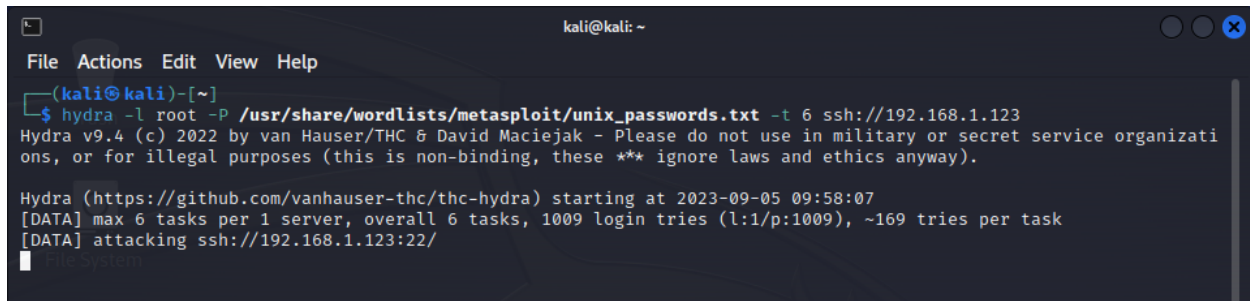
WPA-PSK options:

-E <file> : create EWSA Project file v3
-I <str> : PMKID string (hashcat -m 16800)

The above shows how we can use aircrack-ng.

5. Hydra:

Hydra is a popular password-cracking tool that supports various protocols and services, including SSH, FTP, HTTP, RDP, and more. It can be used to perform brute-force and dictionary attacks.

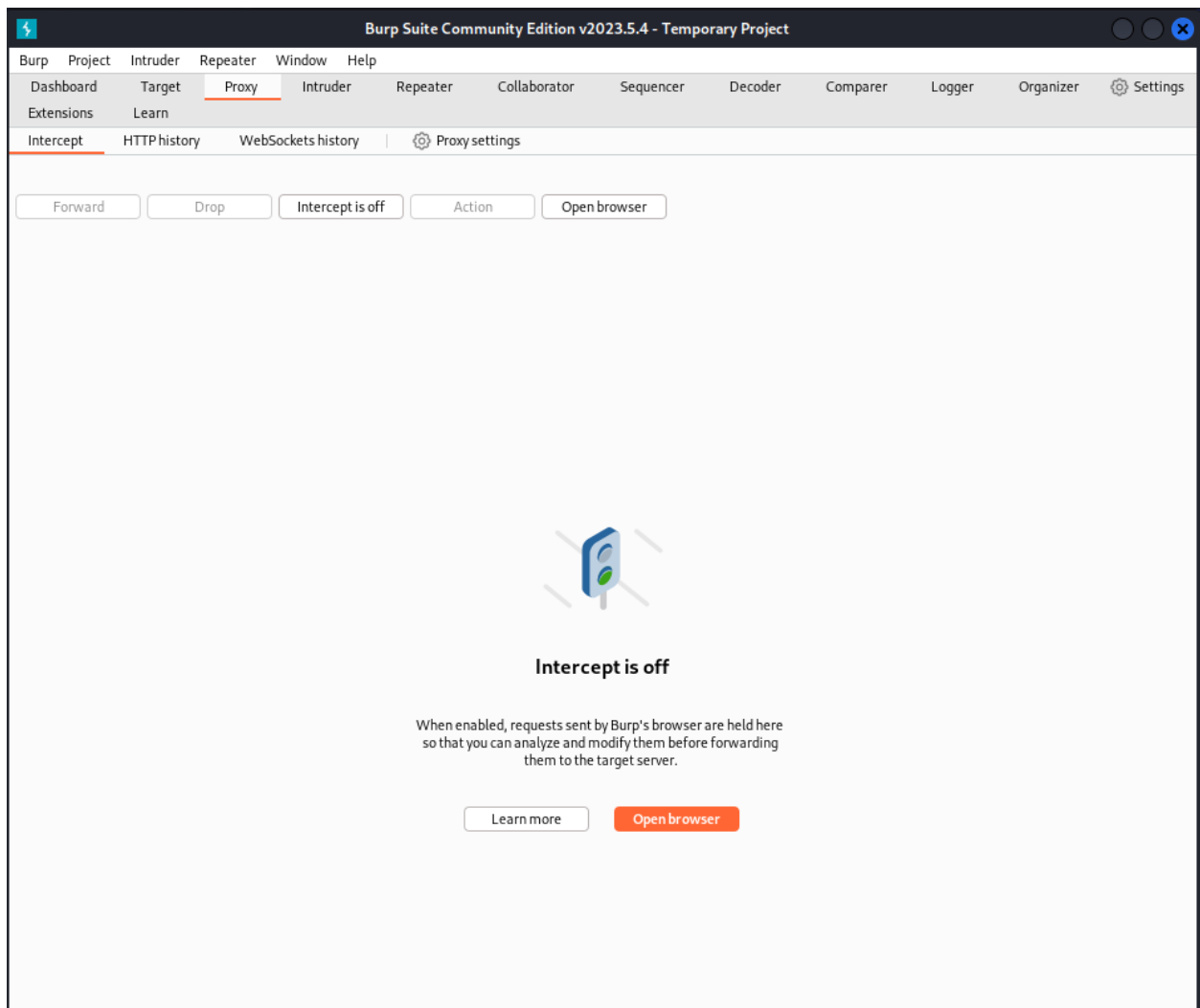
A screenshot of a terminal window on a Kali Linux system. The window title is 'kali@kali: ~'. The terminal shows the execution of the Hydra command: '\$ hydra -l root -P /usr/share/wordlists/metasploit/unix_passwords.txt -t 6 ssh://192.168.1.123'. The output indicates that Hydra v9.4 is starting at 2023-09-05 09:58:07, with a maximum of 6 tasks per server and 1009 login tries. The terminal also shows '[DATA] attacking ssh://192.168.1.123:22/' and a progress bar for the attack.

```
kali@kali: ~  
File Actions Edit View Help  
~(kali@kali)-[~]  
$ hydra -l root -P /usr/share/wordlists/metasploit/unix_passwords.txt -t 6 ssh://192.168.1.123  
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizati  
ons, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).  
  
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-09-05 09:58:07  
[DATA] max 6 tasks per 1 server, overall 6 tasks, 1009 login tries (l:1/p:1009), ~169 tries per task  
[DATA] attacking ssh://192.168.1.123:22/  
█ file system
```

In the above example, we are trying to login as the root user on the given SSH server.

6. Burp Suite:

Burp Suite is a web application security testing tool used by ethical hackers to find vulnerabilities in web applications. It offers features like scanning, crawling, and intercepting HTTP requests and responses.



The Application of Burp Suite, it is highly used to test Web Security. We can use it for various things like Parameter Tampering by intercepting the processes of a website.

7. John the Ripper:

John the Ripper is a password cracking tool that can crack various password hashes using different attack methods, including dictionary attacks and brute-force attacks.


```

(kali@kali)-[~]
$ john -h
John the Ripper 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP [linux-gnu 64-bit x86_64 SSE2 AC]
Copyright (c) 1996-2021 by Solar Designer and others
Homepage: https://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]

--help                Print usage summary
--single[=SECTION[,..]] "Single crack" mode, using default or named rules
--single=:rule[,..]    Same, using "immediate" rule(s)
--single-seed=WORD[,WORD] Add static seed word(s) for all salts in single mode
--single-wordlist=FILE *Short* wordlist with static seed words/morphemes
--single-user-seed=FILE Wordlist with seeds per username (user:password[s]
                        format)
--single-pair-max=N    Override max. number of word pairs generated (6)
--no-single-pair       Disable single word pair generation
--[no-]single-retest-guess Override config for SingleRetestGuess
--wordlist[=FILE] --stdin Wordlist mode, read words from FILE or stdin
                        --pipe like --stdin, but bulk reads, and allows rules
--rules[=SECTION[,..]] Enable word mangling rules (for wordlist or PRINCE
                        modes), using default or named rules
--rules=:rule[;..]     Same, using "immediate" rule(s)
--rules-stack=SECTION[,..] Stacked rules, applied after regular rules or to
                        modes that otherwise don't support rules
--rules-stack=:rule[;..] Same, using "immediate" rule(s)
--rules-skip-nop       Skip any NOP ":" rules (you already ran w/o rules)
--loopback[=FILE]      Like --wordlist, but extract words from a .pot file
--mem-file-size=SIZE   Size threshold for wordlist preload (default 2048 MB)
--dupe-suppression     Suppress all dupes in wordlist (and force preload)
--incremental[=MODE]   "Incremental" mode [using section MODE]
--incremental-charcount=N Override CharCount for incremental mode
--external=MODE        External mode or word filter
--mask[=MASK]          Mask mode using MASK (or default from john.conf)
--markov[=OPTIONS]     "Markov" mode (see doc/MARKOV)
--mkv-stats=FILE       "Markov" stats file
--prince[=FILE]        PRINCE mode, read words from FILE
--prince-loopback[=FILE] Fetch words from a .pot file
--prince-elem-cnt-min=N Minimum number of elements per chain (1)
--prince-elem-cnt-max=[-]N Maximum number of elements per chain (negative N is
                        relative to word length) (8)
--prince-skip=N        Initial skip
--prince-limit=N       Limit number of candidates generated
--prince-wl-dist-len    Calculate length distribution from wordlist
--prince-wl-max=N      Load only N words from input wordlist
--prince-case-permute   Permute case of first letter
--prince-mmap          Memory-map infile (not available with case permute)
--prince-keystore      Just show total keyspace that would be produced
                        (disregarding skip and limit)
--subsets[=CHARSET]    "Subsets" mode (see doc/SUBSETS)
--subsets-required=N   The N first characters of "subsets" charset are

```

This tool, is used widely to crack the passwords. The methos used by this is Brute-Forcing.

8. SQLMap:

SQLMap is an open-source tool for detecting and exploiting SQL injection vulnerabilities in web applications. It automates the process of finding and exploiting SQL injection flaws.

```
$ sqlmap --wizard
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 10:05:39 /2023-09-05/

[10:05:39] [INFO] starting wizard interface
Please enter full target URL (-u): https://chat.openai.com/
POST data (--data) [Enter for None]:
[10:06:00] [WARNING] No GET and/or POST parameter(s) found for testing (e.g. GET parameter 'id' in 'http://www.site.com/vuln.php?id=1'). Will search for forms
Injection difficulty (--level/--risk). Please choose:
[1] Normal (default)
[2] Medium
[3] Hard
> 1
Enumeration (--banner/--current-user/etc). Please choose:
[1] Basic (default)
[2] Intermediate
[3] All
> 1

sqlmap is running, please wait..

[*] ending @ 10:06:24 /2023-09-05/
```

In the above example, we try to do sql injections on chat.openai.com website

9. Nikto:

Nikto is a web server scanner that detects potential security issues in web servers and web applications. It checks for known vulnerabilities, misconfigurations, and outdated software.

```

File Actions Edit View Help
$ nikto -h
Option host requires an argument

Options:
  -ask+          Whether to ask about submitting updates
                  yes   Ask about each (default)
                  no   Don't ask, don't send
                  auto  Don't ask, just send
  -check6        Check if IPv6 is working (connects to ipv6.google.com or value set in nikto.conf)
  -Cgidirs+      Scan these CGI dirs: "none", "all", or values like "/cgi/ /cgi-a/"
  -config+       Use this config file
  -Display+      Turn on/off display outputs:
                  1     Show redirects
                  2     Show cookies received
                  3     Show all 200/OK responses
                  4     Show URLs which require authentication
                  D     Debug output
                  E     Display all HTTP errors
                  P     Print progress to STDOUT
                  S     Scrub output of IPs and hostnames
                  V     Verbose output
  -dbcheck       Check database and other key files for syntax errors
  -evasion+      Encoding technique:
                  1     Random URI encoding (non-UTF8)
                  2     Directory self-reference (../)
                  3     Premature URL ending
                  4     Prepend long random string
                  5     Fake parameter
                  6     TAB as request spacer
                  7     Change the case of the URL
                  8     Use Windows directory separator (\)
                  A     Use a carriage return (0x0d) as a request spacer
                  B     Use binary value 0x0b as a request spacer
  -followredirects Follow 3xx redirects to new location
  -Format+       Save file (-o) format:
                  csv   Comma-separated-value
                  json  JSON Format
                  htm   HTML Format
                  nbe   Nessus NBE format
                  sql   Generic SQL (see docs for schema)
                  txt   Plain text
                  xml   XML Format
                  (if not specified the format will be taken from the file extension passed to -output)

  -Help          This help information
  -host+         Target host/URL
  -id+          Host authentication to use, format is id:pass or id:pass:realm
  -ipv4          IPv4 Only
  -ipv6          IPv6 Only
  -key+         Client certificate key file
  -list-plugins  List all available plugins, perform no testing

```

We can use Nikto for web security scans, in the above example we see the commands to execute Nikto.

10. Gobuster:

Gobuster is a directory and file brute-forcing tool used for discovering hidden paths and files on web servers. It can help identify unprotected or sensitive resources.

```

(kali㉿kali)-[~]
$ gobuster completion bash
# bash completion V2 for gobuster                                -*- shell-script -*-

__gobuster_debug()
{
    if [[ -n ${BASH_COMP_DEBUG_FILE:-} ]]; then
        echo "$*" >> "${BASH_COMP_DEBUG_FILE}"
    fi
}

# Macs have bash3 for which the bash-completion package doesn't include
# _init_completion. This is a minimal version of that function.
__gobuster_init_completion()
{
    COMPREPLY=()
    _get_comp_words_by_ref "$@" cur prev words cword
}

# This function calls the gobuster program to obtain the completion
# results and the directive. It fills the 'out' and 'directive' vars.
__gobuster_get_completion_results() {
    local requestComp lastParam lastChar args

    # Prepare the command to request completions for the program.
    # Calling ${words[0]} instead of directly gobuster allows to handle aliases
    args=("${words[@]:1}")
    requestComp="${words[0]} __complete ${args[*]}"

    lastParam=${words[${#words[@]}-1]}
    lastChar=${lastParam:${#lastParam}-1}
    __gobuster_debug "lastParam ${lastParam}, lastChar ${lastChar}"

    if [ -z "${cur}" ] && [ "${lastChar}" != "=" ]; then
        # If the last parameter is complete (there is a space following it)
        # We add an extra empty parameter so we can indicate this to the go method.
        __gobuster_debug "Adding extra empty parameter"
        requestComp="${requestComp} "
    fi

    # When completing a flag with an = (e.g., gobuster -n=<TAB>)
    # bash focuses on the part after the =, so we need to remove
    # the flag part from $cur
    if [[ "${cur}" = -*=* ]]; then
        cur="${cur#*=}"
    fi

    __gobuster_debug "Calling ${requestComp}"
    # Use eval to handle any environment variables and such
    out=$(eval "${requestComp}" 2>/dev/null)
}

```

This is used in OS based securities, here we can see, we used gobuster to give a complete script for bash.