

# ArcheryOS

Tovi

June 8, 2018

## Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Philosophy of ArcheryOS . . . . .	4
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>i3</b>	<b>6</b>
3.1	Introduction to i3 . . . . .	6
3.2	keybinds . . . . .	6
3.2.1	i3 basics . . . . .	6
3.2.2	Programs . . . . .	8
<b>4</b>	<b>Pentesting</b>	<b>9</b>
4.1	List of pentesting tools installed . . . . .	9
4.1.1	airbase-ng . . . . .	9
4.1.2	aircrack-ng . . . . .	10
4.1.3	airdecap-ng . . . . .	12
4.1.4	airdecloak-ng . . . . .	12
4.1.5	aireplay-ng . . . . .	13
4.1.6	airmon-ng . . . . .	15
4.1.7	airodump-ng . . . . .	15
4.1.8	airolib-ng . . . . .	17
4.1.9	airserv-ng . . . . .	18
4.1.10	airtun-ng . . . . .	18
4.1.11	airventriloquist-ng . . . . .	19
4.1.12	apktool . . . . .	19
4.1.13	autopsy . . . . .	20
4.1.14	bettercap . . . . .	20
4.1.15	burpsuite . . . . .	21
4.1.16	chkrootkit . . . . .	21
4.1.17	chntpw . . . . .	21
4.1.18	cowpatty . . . . .	22
4.1.19	crowbar . . . . .	22
4.1.20	crunch . . . . .	23

4.1.21	cupp . . . . .	24
4.1.22	cymothoa . . . . .	24
4.1.23	dirb . . . . .	25
4.1.24	dirbuster . . . . .	26
4.1.25	dnsspoof . . . . .	27
4.1.26	exe2hex . . . . .	27
4.1.27	ettercap . . . . .	28
4.1.28	searchsploit . . . . .	29
4.1.29	giskismet . . . . .	31
4.1.30	hashcat . . . . .	31
4.1.31	hashdeep . . . . .	44
4.1.32	httrack . . . . .	45
4.1.33	hydra . . . . .	51
4.1.34	intrace . . . . .	53
4.1.35	john . . . . .	53
4.1.36	kismet . . . . .	54
4.1.37	lynis . . . . .	56
4.1.38	macchanger . . . . .	57
4.1.39	maltego-community . . . . .	58
4.1.40	masscan . . . . .	58
4.1.41	mdk3 . . . . .	59
4.1.42	medusa . . . . .	60
4.1.43	movfuscator . . . . .	61
4.1.44	msfconsole . . . . .	71
4.1.45	msfd . . . . .	72
4.1.46	msfpc . . . . .	72
4.1.47	msfrpc . . . . .	74
4.1.48	msfrpcd . . . . .	74
4.1.49	msfvenom . . . . .	75
4.1.50	mimikatz . . . . .	76
4.1.51	miranda . . . . .	76
4.1.52	mitmf . . . . .	76
4.1.53	nbtscan . . . . .	80
4.1.54	ncrack . . . . .	82
4.1.55	netdiscover . . . . .	84
4.1.56	nikto . . . . .	84
4.1.57	nmap . . . . .	87
4.1.58	openvas . . . . .	90
4.1.59	ophcrack . . . . .	90
4.1.60	piCrypt . . . . .	91
4.1.61	pixiewps . . . . .	92
4.1.62	powersploit . . . . .	93
4.1.63	pyrit . . . . .	93
4.1.64	radare2 . . . . .	94
4.1.65	reaver . . . . .	95
4.1.66	rkhunter . . . . .	97

---

4.1.67	social-engineer-toolkit . . . . .	98
4.1.68	sslsan . . . . .	99
4.1.69	steghide . . . . .	100
4.1.70	smbmap . . . . .	100
4.1.71	sqlmap . . . . .	102
4.1.72	swaks . . . . .	104
4.1.73	thc-ipv6 . . . . .	104
4.1.74	twofi . . . . .	104
4.1.75	unicornscan . . . . .	105
4.1.76	urlcrazy . . . . .	107
4.1.77	unix-privesc-check . . . . .	107
4.1.78	veracrypt . . . . .	107
4.1.79	wapiti . . . . .	108
4.1.80	websploit . . . . .	110
4.1.81	wfuzz . . . . .	110
4.1.82	whois . . . . .	113
4.1.83	wifit . . . . .	114
4.1.84	wireshark-gtk . . . . .	115
4.1.85	wpscan . . . . .	116
4.1.86	zaproxy . . . . .	119
4.1.87	zmap . . . . .	119
<b>5</b>	<b>Known bugs and issues</b>	<b>122</b>
5.1	Screen size . . . . .	122
5.2	Ranger image preview . . . . .	122
5.3	Found more bugs? . . . . .	122
<b>6</b>	<b>Thanks</b>	<b>123</b>

# 1 Introduction

ArcheryOS is a rolling release arch based distribution with a distinct focus on penetration testing, digital forensics, programming, and privacy. It uses the default arch linux repositories, to ensure your software is always kept up to date.

## 1.1 Philosophy of ArcheryOS

- Simple

ArcheryOS is designed to be simple to use and install, coming with a curses installer to simplify the arch installation process. It comes preinstalled with a minimal window manager of your choice, such as i3, openbox, or awesome.

*Please note the installer is an offline installation program, simply copying all installed programs to the selected disk. Please update after installation*

- Privacy

ArcheryOS has been configured with privacy in mind. Firefox has been configured to minimize any data leaks that may identify the user, such as browser fingerprinting and WebRTC IP leaks.

You can visit [privacytools](#) for more information.

You can also visit [panopticlick](#) to check your browser fingerprint and [privacytools webrtc check](#) to check for WebRTC IP leaks.

- Penetration Testing

ArcheryOS has a suite of 61 basic and essential tools that every aspiring pentester should have. Unlike other pentesting centric distributions, we believe that providing hundreds of tools, that many people have no idea what they do, let alone how to use them, is not helpful. ArcheryOS comes with a much smaller selection, and documentation on how to use each and every one of them. When the time comes that this suite of tools become insufficient, the user may install any additional programs from the arch repositories or AUR that better suit the target the user is attacking.

## 2 Installation

ArcheryOS works as both a live boot OS and an installed OS.

To install ArcheryOS, press **Mod+Shift+F12**.

1. Choose language
2. Prepare installation
  - (a) Set virtual console
  - (b) Set desktop keyboard layout
  - (c) Partition disks and encrypt with luks, if you so choose
  - (d) Mount partitions
3. Install base
  - (a) Install base packages
  - (b) Run mkinitpcio
  - (c) Install bootloader
4. Configure base
  - (a) Generate FSTAB
  - (b) Set hostname
  - (c) Set system locale
  - (d) Set timezone
  - (e) Set root password
  - (f) Add new users (optional)
  - (g) Set security and systemd tweaks (optional)
5. Close installer and reboot
  - (a) To reboot, press **Mod+Shift+s** and then press **r**. Make sure to unmount the live boot medium.

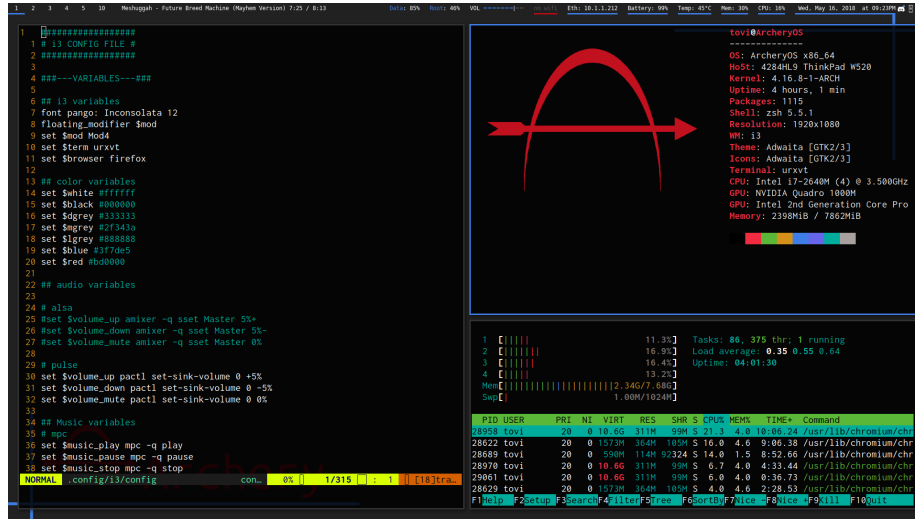


Figure 1: i3 running vim, htop, and neofetch in 3 terminals

## 3 i3

### 3.1 Introduction to i3

“i3 is a tiling window manager, completely written from scratch. The target platforms are GNU/Linux and BSD operating systems, our code is Free and Open Source Software (FOSS) under the BSD license*i3* is primarily targeted at advanced users and developers.” — Stapelberg [2]

### 3.2 keybinds

“Mod” is a reference to the super key, what is known as the “Windows key”. Mod+F1 will show this document at any time

#### 3.2.1 i3 basics

- **Mod+Enter** — Open a terminal window
- **Mod+Shift+Enter** — Open a terminal window running tmux
- **Mod+q** — Close active window
- **Mod+Shift+q** — Close active window
- **Mod+Space** — Toggles between a floating and non floating window

- **Mod+Shift+Space** — Makes a tiled window into a floating window
- **Mod+Shift+r** — Restart i3
- **Mod+d** — rofi (a program launcher)
- **Mod+Shift+d** — rofi in “show window” mode (allows user to navigate to running programs)
- **Mod+t** — Toggle between spawning new windows horizontally or vertically to the active window
- **Mod+v** — Spawn new windows vertically from active window
- **Mod+Shift+v** — Spawn new windows horizontally from active window
- **Mod+f** — Fullscreen
- **Mod+h** — Move to window left of active window
- **Mod+Shift+h** — Move active window left
- **Mod+j** — Move to window below of active window
- **Mod+Shift+j** — Move active window down
- **Mod+k** — Move to window above of active window
- **Mod+Shift+k** — Move active window up
- **Mod+l** — Move to window right of active window
- **Mod+Shift+l** — Move active window right
- **Mod+Shift+y** — Expand active windows width by 10 px
- **Mod+Shift+u** — Shrink active windows hight by 10px
- **Mod+Shift+i** — Expand active windows hight by 10px
- **Mod+o** — Opens a GUI program menu
- **Mod+Shift+o** — Shrink active windows width by 10 px
- **Mod+e** — Change to default layout
- **Mod+w** — Change to tabbed layout
- **Mod+s** — Change to stacked layout
- **Mod+Shift+s** — Lock/logout/shutdown/reboot system
- **Mod+a** — Focuses parent program
- **Mod+n** — Expand outer gaps

- **Mod+Shift+n** — Shrink outer gaps
- **Mod+g** — Expand inner gaps
- **Mod+Shift+g** — Shrink inner gaps
- **Mod+c** — Sets gaps to default width
- **Mod+Shift+c** — Turns off gaps
- **Mod+u** — Next song
- **Mod+y** — Previous song

### 3.2.2 Programs

- **Mod+Shift+a** — Audio (pavucontrol)
- **Mod+b** — Browser (firefox)
- **Mod+i** — System information (htop)
- **Mod+m** — Music (ncmpcpp)
- **Mod+Shift+m** — Mute audio
- **Mod+p** — Play/pause music
- **Mod+Shift+p** — Take screenshot (scrot)
- **Mod+r** — File manager (ranger)
- **Mod+Shift+w** — Newsboat
- **Mod+z** — Toggle dropdown terminal
- **Mod+Shift+z** — Reopen dropdown terminal (in case you accidentally close it)



## 4 Pentesting

### 4.1 List of pentesting tools installed

#### 4.1.1 airbase-ng

Airbase-ng is a multi-purpose tool aimed at attacking clients as opposed to the Access Point (AP) itself. Since it is so versatile and flexible, summarizing it is a challenge. Here are some of the feature highlights:

- Implements the Caffe Latte WEP client attack
- Implements the Hirte WEP client attack
- Ability to cause the WPA/WPA2 handshake to be captured
- Ability to act as an ad-hoc Access Point
- Ability to act as a full Access Point
- Ability to filter by SSID or client MAC addresses
- Ability to manipulate and resend packets
- Ability to encrypt sent packets and decrypt received packets

The main idea is of the implementation is that it should encourage clients to associate with the fake AP, not prevent them from accessing the real AP.

---

```
[root@ArcheryOS ~]# airbase-ng --help
```

```
Airbase-ng 1.2 - (C) 2008-2018 Thomas d'Otreppe
Original work: Martin Beck
https://www.aircrack-ng.org
```

```
usage: airbase-ng <options> <replay interface>
```

```
Options:
```

```

-a bssid      : set Access Point MAC address
-i iface      : capture packets from this interface
-w WEP key    : use this WEP key to en-/decrypt packets
-h MAC        : source mac for MITM mode
-f disallow   : disallow specified client MACs (default: allow)
-W 0|1        : [don't] set WEP flag in beacons 0|1 (default:
                auto)
-q            : quiet (do not print statistics)
-v            : verbose (print more messages)
-A            : Ad-Hoc Mode (allows other clients to peer)
-Y in|out|both : external packet processing
-c channel    : sets the channel the AP is running on
-X            : hidden ESSID
```

---

```

-s          : force shared key authentication (default: auto)
-S          : set shared key challenge length (default: 128)
-L          : Caffe-Latte WEP attack (use if driver can't send
             frags)
-N          : cfrag WEP attack (recommended)
-x nbpps    : number of packets per second (default: 100)
-y          : disables responses to broadcast probes
-0          : set all WPA,WEP,open tags. can't be used with -z
             & -Z
-z type     : sets WPA1 tags. 1=WEP40 2=TKIP 3=WRAP 4=CCMP
             5=WEP104
-Z type     : same as -z, but for WPA2
-V type     : fake EAPOL 1=MD5 2=SHA1 3=auto
-F prefix   : write all sent and received frames into pcap file
-P          : respond to all probes, even when specifying ESSIDs
-I interval : sets the beacon interval value in ms
-C seconds  : enables beaconing of probed ESSID values
             (requires -P)
-n hex      : User specified ANonce when doing the 4-way
             handshake

Filter options:
--bssid MAC  : BSSID to filter/use
--bssids file : read a list of BSSIDs out of that file
--client MAC : MAC of client to filter
--clients file : read a list of MACs out of that file
--essid ESSID : specify a single ESSID (default: default)
--essids file : read a list of ESSIDs out of that file

--help      : Displays this usage screen

```

---

#### 4.1.2 aircrack-ng

aircrack-ng is an 802.11 WEP and WPA/WPA2-PSK key cracking program. It can recover the WEP key once enough encrypted packets have been captured with airodump-ng. This part of the aircrack-ng suite determines the WEP key using two fundamental methods. The first method is via the PTW approach (Pyshkin, Tews, Weinmann). The main advantage of the PTW approach is that very few data packets are required to crack the WEP key. The second method is the FMS/KoreK method. The FMS/KoreK method incorporates various statistical attacks to discover the WEP key and uses these in combination with brute forcing. Additionally, the program offers a dictionary method for determining the WEP key. For cracking WPA/WPA2 pre-shared keys, a wordlist (file or stdin) or an airolib-ng has to be used.

---

```
[root@ArcheryOS ~]# aircrack-ng --help
```

```
Aircrack-ng 1.2 - (C) 2006-2018 Thomas d'Otreppe
```

<https://www.aircrack-ng.org>

usage: aircrack-ng [options] <.cap / .ivs 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
```

WEP and WPA-PSK cracking options:

```
-w <words> : path to wordlist(s) filename(s)
```

WPA-PSK options:

```
-E <file> : create EWSA Project file v3
-j <file> : create Hashcat v3.6+ file (HCCAPX)
-J <file> : create Hashcat file (HCCAP)
-S        : WPA cracking speed test
-r <DB>    : path to airolib-ng database
            (Cannot be used with -w)
```

Other options:

-u : Displays # of CPUs & MMX/SSE support  
--help : Displays this usage screen

---

#### 4.1.3 airdecap-ng

airdecap-ng decrypts a WEP/WPA crypted pcap file to a unencrypted one by using the right WEP/WPA keys.

---

```
[root@ArcheryOS ~]# airdecap-ng --help
```

```
Airdecap-ng 1.2 - (C) 2006-2018 Thomas d'Ottreppe  
https://www.aircrack-ng.org
```

```
usage: airdecap-ng [options] <pcap file>
```

Common options:

```
-l : don't remove the 802.11 header  
-b <bssid> : access point MAC address filter  
-e <essid> : target network SSID  
-o <fname> : output file for decrypted packets (default <src>-dec)
```

WEP specific option:

```
-w <key> : target network WEP key in hex  
-c <fname> : output file for corrupted WEP packets (default  
             <src>-bad)
```

WPA specific options:

```
-p <pass> : target network WPA passphrase  
-k <pmk> : WPA Pairwise Master Key in hex
```

```
--help : Displays this usage screen
```

---

#### 4.1.4 airdecloak-ng

airdecloak-ng is a tool that removes wep cloaking from a pcap file. Some WIPS (actually one) can actively "prevent" cracking a WEP key by inserting chaff (fake wep frames) in the air to fool aircrack-ng. In some rare cases, cloaking fails and the key can be recovered without removing this chaff. In the cases where the key cannot be recovered, use this tool to filter out chaff.

---

```
[root@ArcheryOS ~]# airdecloak-ng --help
```

```
Airdecloak-ng 1.2 - (C) 2006-2018 Thomas d'Ottreppe  
https://www.aircrack-ng.org
```

```
usage: airdecloak-ng [options]

options:

Mandatory:
  -i <file>           : Input capture file
  --ssid <ESSID>      : ESSID of the network to filter
  or
  --bssid <BSSID>     : BSSID of the network to filter

Optional:
  -o <file>           : Output packets (valid) file (default:
    <src>-filtered.pcap)
  -c <file>           : Output packets (cloaked) file (default:
    <src>-cloaked.pcap)
  -u <file>           : Output packets (unknown/ignored) file
    (default: invalid_status.pcap)
  --filters <filters> : Apply filters (separated by a comma). Filters:
    signal:          Try to filter based on signal.
    duplicate_sn:     Remove all duplicate sequence numbers
                      for both the AP and the client.
    duplicate_sn_ap:  Remove duplicate sequence number for
                      the AP only.
    duplicate_sn_client: Remove duplicate sequence number for the
                      client only.
    consecutive_sn:   Filter based on the fact that IV should
                      be consecutive (only for AP).
    duplicate_iv:     Remove all duplicate IV.
    signal_dup_consec_sn: Use signal (if available), duplicate and
                      consecutive sequence number (filtering is
                      much more precise than using all these
                      filters one by one).
  --null-packets      : Assume that null packets can be cloaked.
  --disable-base_filter : Do not apply base filter.
  --drop-frag         : Drop fragmented packets

  --help              : Displays this usage screen
```

---

#### 4.1.5 aireplay-ng

aireplay-ng is used to inject/replay frames. The primary function is to generate traffic for the later use in aircrack-ng for cracking the WEP and WPA-PSK keys. There are different attacks which can cause deauthentications for the purpose of capturing WPA handshake data, fake authentications, Interactive packet replay, hand-crafted ARP request injection and ARP-request reinjection. With the packetforge-ng tool it's possible to create arbitrary frames.

---

```
[root@ArcheryOS ~]# aireplay-ng --help
```

Aireplay-ng 1.2 - (C) 2006-2018 Thomas d'Ottreppe  
<https://www.aircrack-ng.org>

usage: aireplay-ng <options> <replay interface>

Filter options:

- b bssid : MAC address, Access Point
- d dmac : MAC address, Destination
- s smac : MAC address, Source
- m len : minimum packet length
- n len : maximum packet length
- u type : frame control, type field
- v subt : frame control, subtype field
- t tods : frame control, To DS bit
- f fromds : frame control, From DS bit
- w iswep : frame control, WEP bit
- D : disable AP detection

Replay options:

- x nbpps : number of packets per second
- p fctrl : set frame control word (hex)
- a bssid : set Access Point MAC address
- c dmac : set Destination MAC address
- h smac : set Source MAC address
- g value : change ring buffer size (default: 8)
- F : choose first matching packet

Fakeauth attack options:

- e essid : set target AP SSID
- o npckts : number of packets per burst (0=auto, default: 1)
- q sec : seconds between keep-alives
- Q : send reassociation requests
- y prga : keystream for shared key auth
- T n : exit after retry fake auth request n time

Arp Replay attack options:

- j : inject FromDS packets

Fragmentation attack options:

- k IP : set destination IP in fragments
- l IP : set source IP in fragments

Test attack options:



computer, airodump-ng is capable of logging the coordinates of the found access points. Additionally, airodump-ng writes out a text file containing the details of all access points and clients seen.

---

```
[root@ArcheryOS ~]# airodump-ng --help

Airodump-ng 1.2 - (C) 2006-2018 Thomas d'Ottreppe
https://www.aircrack-ng.org

usage: airodump-ng <options> <interface>[,<interface>,...]

Options:
  --ivs                : Save only captured IVs
  --gpsd               : Use GPSd
  --write <prefix>    : Dump file prefix
  -w                  : same as --write
  --beacons            : Record all beacons in dump file
  --update <secs>     : Display update delay in seconds
  --showack            : Prints ack/cts/rts statistics
  -h                  : Hides known stations for --showack
  -f <msecs>          : Time in ms between hopping channels
  --berlin <secs>     : Time before removing the AP/client
                        from the screen when no more packets
                        are received (Default: 120 seconds)
  -r <file>           : Read packets from that file
  -x <msecs>          : Active Scanning Simulation
  --manufacturer      : Display manufacturer from IEEE OUI list
  --uptime            : Display AP Uptime from Beacon Timestamp
  --wps               : Display WPS information (if any)
  --output-format
    <formats>         : Output format. Possible values:
                        pcap, ivs, csv, gps, kismet, netxml
  --ignore-negative-one : Removes the message that says
                        fixed channel <interface>: -1
  --write-interval
    <seconds>         : Output file(s) write interval in seconds

Filter options:
  --encrypt <suite>   : Filter APs by cipher suite
  --netmask <netmask> : Filter APs by mask
  --bssid <bssid>     : Filter APs by BSSID
  --essid <essid>     : Filter APs by ESSID
  --essid-regex <regex> : Filter APs by ESSID using a regular
                        expression
  -a                  : Filter unassociated clients

By default, airodump-ng hop on 2.4GHz channels.
You can make it capture on other/specific channel(s) by using:
  --ht20              : Set channel to HT20 (802.11n)
  --ht40-             : Set channel to HT40- (802.11n)
```



---

```

--ht40+          : Set channel to HT40+ (802.11n)
--channel <channels> : Capture on specific channels
--band <abg>      : Band on which airodump-ng should hop
-C <frequencies> : Uses these frequencies in MHz to hop
--cswitch <method> : Set channel switching method
                  0 : FIFO (default)
                  1 : Round Robin
                  2 : Hop on last
-s              : same as --cswitch

--help          : Displays this usage screen

```

---

#### 4.1.8 airolib-ng

airolib-ng is a tool for the aircrack-ng suite to store and manage essid and password lists, compute their Pairwise Master Keys (PMKs) and use them in WPA/WPA2 cracking. The program uses the lightweight SQLite3 database as the storage mechanism which is available on most platforms. The SQLite3 database was selected taking in consideration platform availability plus management, memory and disk overhead.

---

```
[root@ArcheryOS ~]# airolib-ng --help
```

```

Airolib-ng 1.2 - (C) 2007, 2008, 2009 ebfe
https://www.aircrack-ng.org

```

```
Usage: airolib-ng <database> <operation> [options]
```

```
Operations:
```

```

--stats          : Output information about the database.
--sql <sql>      : Execute specified SQL statement.
--clean [all]    : Clean the database from old junk. 'all' will also
                  reduce filesize if possible and run an integrity
                  check.
--batch          : Start batch-processing all combinations of ESSIDs
                  and passwords.
--verify [all]   : Verify a set of randomly chosen PMKs.
                  If 'all' is given, all invalid PMK will be deleted.

--import [ssid|passwd] <file> :
                  Import a text file as a list of ESSIDs or
                  passwords.
--import cowpatty <file>      :
                  Import a cowpatty file.

--export cowpatty <ssid> <file> :
                  Export to a cowpatty file.

```

---

### 4.1.9 aircserv-ng

airserv-ng is a wireless card server which allows multiple wireless application programs to independently use a wireless card via a client-server TCP network connection. All operating system and wireless card driver specific code is incorporated into the server. This eliminates the need for each wireless application to contain the complex wireless card and driver logic. It is also supports multiple operating systems.

---

```
[root@ArcheryOS ~]# aircserv-ng -h

Aircserv-ng 1.2 - (C) 2007, 2008, 2009 Andrea Bittau
https://www.aircrack-ng.org

Usage: aircserv-ng <options>

Options:

    -h          : This help screen
    -p <port>   : TCP port to listen on (default:666)
    -d <iface>  : Wifi interface to use
    -c <chan>   : Channel to use
    -v <level>  : Debug level (1 to 3; default: 1)
```

---

### 4.1.10 airtun-ng

airtun-ng creates a virtual tunnel interface (atX) for sending arbitrary IP packets by using raw ieee802.11 packet injection.

---

```
[root@ArcheryOS ~]# airtun-ng --help

Airtun-ng 1.2 - (C) 2006-2018 Thomas d'Otreppe
Original work: Martin Beck
https://www.aircrack-ng.org

usage: airtun-ng <options> <replay interface>

    -x nbpps      : number of packets per second (default: 100)
    -a bssid      : set Access Point MAC address
                   In WDS Mode this sets the Receiver
    -i iface      : capture packets from this interface
    -y file       : read PRGA from this file
    -w wepkey     : use this WEP-KEY to encrypt packets
    -p pass       : use this WPA passphrase to decrypt packets
                   (use with -a and -e)
    -e essid      : target network SSID (use with -p)
    -t tods       : send frames to AP (1) or to client (0)
                   or tunnel them into a WDS/Bridge (2)
    -r file       : read frames out of pcap file
```

-h MAC : source MAC address

WDS/Bridge Mode options:

-s transmitter : set Transmitter MAC address for WDS Mode  
 -b : bidirectional mode. This enables communication  
       in Transmitter's AND Receiver's networks.  
       Works only if you can see both stations.

Repeater options:

--repeat : activates repeat mode  
 --bssid <mac> : BSSID to repeat  
 --netmask <mask> : netmask for BSSID filter  
  
 --help : Displays this usage screen

#### 4.1.11 airventriloquist-ng

```
[root@ArcheryOS ~]# airventriloquist-ng --help
```

Airventriloquist-ng 1.2 - (C) 2015 Tim de Waal  
<https://www.aircrack-ng.org>

usage: airventriloquist-ng [options]

-i <replay interface> : Interface to listen and inject on  
 -d | --deauth : Send active deauths to encrypted stations  
 -e | --essid <value> : ESSID of target network  
 -p | --passphrase <val> : WPA Passphrase of target network  
 -c | --icmp : Respond to all ICMP frames (Debug)  
 -n | --dns : IP to resolve all DNS queries to  
 -s | --hijack <URL> : URL to look for in HTTP requests  
                       <URL> can have wildcards  
                       eg: \*jquery\*.js\*  
 -r | --redirect <URL> : URL to redirect to  
 -v | --verbose : Verbose output  
 --help : This super helpful message

#### 4.1.12 apktool

apktool is a tool for reverse engineering 3rd party, closed, binary Android apps. It can decode resources to nearly original form and rebuild them after making some modifications.

```
[root@ArcheryOS ~]# apktool
```

Apktool v2.3.3 - a tool for reengineering Android apk files  
 with smali v2.2.2 and baksmali v2.2.2  
 Copyright 2014 Ryszard Winiewski <brut.all@gmail.com>

Updated by Connor Tumbleson <connor.tumbleson@gmail.com>

```
usage: apktool
  -advance,--advanced prints advance information.
  -version,--version prints the version then exits
usage: apktool if|install-framework [options] <framework.apk>
  -p,--frame-path <dir> Stores framework files into <dir>.
  -t,--tag <tag> Tag frameworks using <tag>.
usage: apktool d[ecode] [options] <file_apk>
  -f,--force Force delete destination directory.
  -o,--output <dir> The name of folder that gets written. Default is
                    apk.out
  -p,--frame-path <dir> Uses framework files located in <dir>.
  -r,--no-res Do not decode resources.
  -s,--no-src Do not decode sources.
  -t,--frame-tag <tag> Uses framework files tagged by <tag>.
usage: apktool b[uild] [options] <app_path>
  -f,--force-all Skip changes detection and build all files.
  -o,--output <dir> The name of apk that gets written. Default is
                    dist/name.apk
  -p,--frame-path <dir> Uses framework files located in <dir>.
```

For additional info, see: <http://ibotpeaches.github.io/Apktool/>  
 For smali/baksmali info, see: <https://github.com/JesusFreke/smali>

#### 4.1.13 autopsy

Autopsy is a digital forensics platform and graphical interface to The Sleuth Kit and other digital forensics tools. It is used by law enforcement, military, and corporate examiners to investigate what happened on a computer. You can even use it to recover photos from your camera's memory card.

---

```
[root@ArcheryOS ~]# autopsy
```

---

#### 4.1.14 bettercap

bettercap is the Swiss army knife for network attacks and monitoring.

---

```
[root@ArcheryOS ~]# bettercap --help
Usage of bettercap:
  -autostart string
    Comma separated list of modules to auto start. (default
    "events.stream, net.recon, update.check")
  -caplet string
    Read commands from this file and execute them in the interactive
    session.
  -cpu-profile file
```

---

```

    Write cpu profile file.
-debug
    Print debug messages.
-env-file string
    Load environment variables from this file if found, set to empty
    to disable environment persistence.
-eval string
    Run one or more commands separated by ; in the interactive
    session, used to set variables via command line.
-iface string
    Network interface to bind to, if empty the default interface will
    be auto selected.
-mem-profile file
    Write memory profile to file.
-no-colors
    Disable output color effects.
-no-history
    Disable interactive session history file.
-silent
    Suppress all logs which are not errors.

```

---

#### 4.1.15 burpsuite

is a graphical tool for testing Web application security. The tool is written in Java and developed by PortSwigger Security.

---

```
[root@ArcheryOS ~]# burpsuite
```

---

#### 4.1.16 chkrootkit

chkrootkit is a tool to locally check for signs of a rootkit.

---

```
[root@ArcheryOS ~]# chkrootkit
```

---

#### 4.1.17 chntpw

chntpw is a program used to remove or change windows passwords when run from a live usb or cd.

---

```

[root@ArcheryOS ~]# chntpw -h
chntpw: change password of a user in a Windows SAM file, or invoke
    registry editor. Should handle both 32 and 64 bit windows and all
    version from NT3.x to Win8.1
chntpw [OPTIONS] <samfile> [systemfile] [securityfile] [otherreghive]
    [...]
-h          This message
-u <user>   Username or RID (0x3e9 for example) to interactively edit

```

---

```

-l      list all users in SAM file and exit
-i      Interactive Menu system
-e      Registry editor. Now with full write support!
-d      Enter buffer debugger instead (hex editor),
-v      Be a little more verbose (for debugging)
-L      For scripts, write names of changed files to /tmp/changed
-N      No allocation mode. Only same length overwrites possible
        (very safe mode)
-E      No expand mode, do not expand hive file (safe mode)

```

---

Usernames can be given as name or RID (in hex with 0x first)

---

#### 4.1.18 cowpatty

cowpatty is a program that performs an offline dictionary attack against WPA/WPA2 .cap files.

---

```

[root@ArcheryOS ~]# cowpatty
cowpatty 4.6 - WPA-PSK dictionary attack. <jwright@hasborg.com>
cowpatty: Must supply a pcap file with -r

```

Usage: cowpatty [options]

```

-f      Dictionary file
-d      Hash file (genpmk)
-r      Packet capture file
-s      Network SSID (enclose in quotes if SSID includes spaces)
-c      Check for valid 4-way frames, does not crack
-h      Print this help information and exit
-v      Print verbose information (more -v for more verbosity)
-V      Print program version and exit

```

---

#### 4.1.19 crowbar

Crowbar is a brute forcing tool that can be used when hydra is not applicable.

---

```

[root@ArcheryOS ~]# crowbar -h
usage: Usage: use --help for further information

```

Crowbar is a brute force tool which supports OpenVPN, Remote Desktop Protocol, SSH Private Keys and VNC Keys.

positional arguments:  
options

optional arguments:  
-h, --help show this help message and exit

---

```

-b {vnckey,sshkey,rdp,openvpn}, --brute {vnckey,sshkey,rdp,openvpn}
    Target service
-s SERVER, --server SERVER
    Static target
-S SERVER_FILE, --serverfile SERVER_FILE
    Multiple targets stored in a file
-u USERNAME [USERNAME ...], --username USERNAME [USERNAME ...]
    Static name to login with
-U USERNAME_FILE, --usernamefile USERNAME_FILE
    Multiple names to login with, stored in a file
-n THREAD, --number THREAD
    Number of threads to be active at once
-l FILE, --log FILE Log file (only write attempts)
-o FILE, --output FILE
    Output file (write everything else)
-c PASSWD, --passwd PASSWD
    Static password to login with
-C FILE, --passwdfile FILE
    Multiple passwords to login with, stored in a file
-t TIMEOUT, --timeout TIMEOUT
    [SSH] How long to wait for each thread (seconds)
-p PORT, --port PORT Alter the port if the service is not using the
    default
    value
-k KEY_FILE, --keyfile KEY_FILE
    [SSH/VNC] (Private) Key file or folder containing
    multiple files
-m CONFIG, --config CONFIG
    [OpenVPN] Configuration file
-d, --discover Port scan before attacking open ports
-v, --verbose Enable verbose output (-vv for more)
-D, --debug Enable debug mode
-q, --quiet Only display successful logins

```

---

#### 4.1.20 crunch

crunch is used to brute force logins and passwords by creating all iterations of a specified length, and specified charset.

---

```

[root@ArcheryOS ~]# crunch --help
crunch version 3.6

```

Crunch can create a wordlist based on criteria you specify. The output from crunch can be sent to the screen, file, or to another program.

Usage: crunch <min> <max> [options]  
 where min and max are numbers

Please refer to the man page [for](#) instructions and examples on how to use `crunch`

#### 4.1.21 cupp

cupp (Common User Passwords Profiler) creates a custom wordlist based on keywords about the target.

```
[root@ArcheryOS ~]# cupp3 -h
usage: cupp3 [-h] (-i | -w FILENAME | -l | -a | -v) [-q]
```

Common User Passwords Profiler

optional arguments:

```
-h, --help          show this help message and exit
-i, --interactive   Interactive questions for user password profiling
-w FILENAME         Use this option to improve existing dictionary, or
                    WyD.pl
                    output to make some pwnsauce
-l                 Download huge wordlists from repository
-a                 Parse default usernames and passwords directly from
                    Alecto DB. Project Alecto uses purified databases of
                    Phenoelit and CIRT which were merged and enhanced
-v, --version       version of this program
-q, --quiet         Quiet mode (don't print banner)
```

#### 4.1.22 cymothoa

Cymothoa is a backdooring tool, that inject backdoor's shellcode directly into running applications. Stealth and lightweight...

```
[root@ArcheryOS ~]# cymothoa -h
```

-----  
 / ( ) | | | \ / ( ) - \ / \ ( ) |  
 ( ( ) | | | | | | | | | | | / --- |  
 \ ) \ | | | \ / \ ) | | \ \ / \ --- |  
 ( /

Runtime shellcode injection, **for** stealthy backdoors...

By codwizard (codwizard@gmail.com) and crossbower (crossbower@gmail.com)  
from ES-Malaria by ElectronicSouls (<http://www.0x4553.org>).

Usage:

```
cymothoa -p <pid> -s <shellcode_number> [options]
```

Main options:



```

-p process pid
-s shellcode number
-l memory region name for shellcode injection (default /lib/ld)
  see /proc/pid/maps...
-h print this help screen
-S list available shellcodes

```

Payload personalization options:

```

-x set the IP
-y set the port number
-r set the port number 2
-z set the username (3 bytes)
-o set the password (8 bytes)
-i set the interpreter (def /bin/bash)
-c set the script code (from cmd line)
-F do not fork parent process

```

#### 4.1.23 dirb

DIRB is a Web Content Scanner. It looks for web files by launching a dictionary attack.

```
[root@ArcheryOS ~]# dirb
```

```

-----
DIRB v2.22
By The Dark Raver
-----

```

```
dirb <url_base> [<wordlist_file(s)>] [options]
```

```
===== NOTES =====
```

```

<url_base> : Base URL to scan. (Use -resume for session resuming)
<wordlist_file(s)> : List of wordfiles.
                  (wordfile1,wordfile2,wordfile3...)

```

```
===== HOTKEYS =====
```

```

'n' -> Go to next directory.
'q' -> Stop scan. (Saving state for resume)
'r' -> Remaining scan stats.

```

```
===== OPTIONS =====
```

```

-a <agent_string> : Specify your custom USER_AGENT.
-c <cookie_string> : Set a cookie for the HTTP request.
-f : Fine tuning of NOT_FOUND (404) detection.
-H <header_string> : Add a custom header to the HTTP request.
-i : Use case-insensitive search.
-l : Print "Location" header when found.
-N <nf_code>: Ignore responses with this HTTP code.

```

```

-o <output_file> : Save output to disk.
-p <proxy[:port]> : Use this proxy. (Default port is 1080)
-P <proxy_username:proxy_password> : Proxy Authentication.
-r : Don't search recursively.
-R : Interactive recursion. (Asks for each directory)
-S : Silent Mode. Don't show tested words. (For dumb terminals)
-t : Don't force an ending '/' on URLs.
-u <username:password> : HTTP Authentication.
-v : Show also NOT_FOUND pages.
-w : Don't stop on WARNING messages.
-X <extensions> / -x <exts_file> : Append each word with this
    extensions.
-z <milisecs> : Add a miliseconds delay to not cause excessive Flood.

```

---

===== EXAMPLES =====

```

dirb http://url/directory/ (Simple Test)
dirb http://url/ -X .html (Test files with '.html' extension)
dirb http://url/ /usr/share/dirb/wordlists/vulns/apache.txt (Test with
    apache.txt wordlist)
dirb https://secure_url/ (Simple Test with SSL)

```

---

#### 4.1.24 dirbuster

DirBuster is a multi threaded java application designed to brute force directories and files names on web/application servers. Often is the case now of what looks like a web server in a state of default installation is actually not, and has pages and applications hidden within. DirBuster attempts to find these.

---

```

[root@ArcheryOS ~]# dirbuster --help
DirBuster - 1.0-RC1
Usage: java -jar DirBuster-1.0-RC1 -u <URL http://example.com/> [Options]

```

Options:

```

-h : Display this help message
-H : Start DirBuster in headless mode (no gui), report will be auto
    saved on exit
-l <Word list to use> : The Word list to use for the list based
    brute force. Default: /home/user/directory-list-2.3-small.txt
-g : Only use GET requests. Default Not Set
-e <File Extention list> : File Extention list eg asp,aspx. Default:
    php
-t <Number of Threads> : Number of connection threads to use.
    Default: 10
-s <Start point> : Start point of the scan. Default: /
-v : Verbose output, Default: Not set
-P : Don't Parse html, Default: Not Set
-R : Don't be recursive, Default: Not Set
-r <location> : File to save report to. Default:

```

```
/home/user/DirBuster-Report-[hostname]-[port].txt
```

Examples:

Run DirBuster **in** headless mode

```
java -jar DirBuster-1.0-RC1.jar -H -u https://www.target.com/
```

Start GUI with target prepopulated

```
java -jar DirBuster-1.0-RC1.jar -u https://www.target.com/
```

---

#### 4.1.25 dnsspoof

dnsspoof forges replies to arbitrary DNS address / pointer queries on the LAN. This is useful in bypassing hostname-based access controls, or in implementing a variety of man-in-the-middle attacks.

---

```
[root@ArcheryOS ~]# dnsspoof -h
```

```
Version: 2.4
```

```
Usage: dnsspoof [-i interface] [-f hostsfile] [expression]
```

---

#### 4.1.26 exe2hex

exe2hex encodes an executable binary file into ASCII text format. The result then can be transferred to the target machine (It is much easier to echo a ASCII file than binary data). Upon executing exe2hex's output file, the original program is restored by using DEBUG.exe or PowerShell (which are pre-installed by default on Windows).

---

```
[root@ArcheryOS ~]# exe2hex -h
```

```
[*] exe2hex v1.5
```

```
Usage: exe2hex [options]
```

---

Options:

```
-h, --help show this help message and exit
-x EXE      The EXE binary file to convert
-s          Read from STDIN
-b BAT      BAT output file (DEBUG.exe method - x86)
-p POSH     PoSh output file (PowerShell method - x86/x64)
-e          URL encode the output
-r TEXT     pRefix - text to add before the command on each line
-f TEXT     suFfix - text to add after the command on each line
-l INT      Maximum HEX values per line
-c          Clones and compress the file before converting (-cc for
            higher
            compression)
-t          Create a Expect file, to automate to a Telnet session.
-w          Create a Expect file, to automate to a WinEXE session.
```

---

-v            Enable verbose mode

---

#### 4.1.27 ettercap

ettercap is a multipurpose sniffer/content filter for man in the middle attacks.

---

```
[root@ArcheryOS ~]# ettercap --help
```

ettercap 0.8.2 copyright 2001-2015 Ettercap Development Team

Usage: ettercap [OPTIONS] [TARGET1] [TARGET2]

TARGET is in the format MAC/IP/IPv6/PORTs (see the man for further detail)

##### Sniffing and Attack options:

```
-M, --mitm <METHOD:ARGS> perform a mitm attack
-o, --only-mitm           don't sniff, only perform the mitm attack
-b, --broadcast           sniff packets destined to broadcast
-B, --bridge <IFACE>      use bridged sniff (needs 2 ifaces)
-p, --nopromisc           do not put the iface in promisc mode
-S, --nossllmitm         do not forge SSL certificates
-u, --unoffensive         do not forward packets
-r, --read <file>        read data from pcapfile <file>
-f, --pcapfilter <string> set the pcap filter <string>
-R, --reversed            use reversed TARGET matching
-t, --proto <proto>      sniff only this proto (default is all)
    --certificate <file> certificate file to use for SSL MiTM
    --private-key <file> private key file to use for SSL MiTM
```

##### User Interface Type:

```
-T, --text                use text only GUI
    -q, --quiet           do not display packet contents
    -s, --script <CMD>    issue these commands to the GUI
-C, --curses              use curses GUI
-D, --daemon              daemonize ettercap (no GUI)
-G, --gtk                 use GTK+ GUI
```

##### Logging options:

```
-w, --write <file>        write sniffed data to pcapfile <file>
-L, --log <logfile>       log all the traffic to this <logfile>
-l, --log-info <logfile>  log only passive infos to this <logfile>
-m, --log-msg <logfile>   log all the messages to this <logfile>
-c, --compress            use gzip compression on log files
```

##### Visualization options:

```
-d, --dns                 resolves ip addresses into hostnames
-V, --visual <format>    set the visualization format
```

```
-e, --regex <regex>      visualize only packets matching this regex
-E, --ext-headers        print extended header for every pck
-Q, --superquiet         do not display user and password
```

LUA options:

```
--lua-script <script1>,[<script2>,...] comma-separted list of LUA
scripts
--lua-args n1=v1,[n2=v2,...]      comma-separated arguments
to LUA script(s)
```

General options:

```
-i, --iface <iface>      use this network interface
-I, --liface             show all the network interfaces
-Y, --secondary <ifaces> list of secondary network interfaces
-n, --netmask <netmask> force this <netmask> on iface
-A, --address <address> force this local <address> on iface
-P, --plugin <plugin>    launch this <plugin>
-F, --filter <file>      load the filter <file> (content filter)
-z, --silent             do not perform the initial ARP scan
-6, --ip6scan            send ICMPv6 probes to discover IPv6 nodes on
the link
-j, --load-hosts <file>  load the hosts list from <file>
-k, --save-hosts <file>  save the hosts list to <file>
-W, --wifi-key <wkey>    use this key to decrypt wifi packets (wep or
wpa)
-a, --config <config>    use the alterative config file <config>
```

Standard options:

```
-v, --version            prints the version and exit
-h, --help               this help screen
```

#### 4.1.28 searchsploit

searchsploit is a program used to search the exploit database.

```
[root@ArcheryOS ~]# searchsploit --help
Usage: searchsploit [options] term1 [term2] ... [termN]
```

```
=====
```

Examples

```
=====
```

```
searchsploit afd windows local
searchsploit -t oracle windows
searchsploit -p 39446
searchsploit linux kernel 3.2 --exclude="(PoC)|/dos/"
```

For more examples, see the manual:

<https://www.exploit-db.com/searchsploit/>

=====

#### Options

=====

```
-c, --case [Term]      Perform a case-sensitive search (Default is
                        inSEnsITiVe).
-e, --exact [Term]     Perform an EXACT match on exploit title
                        (Default is AND) [Implies "-t"].
-h, --help             Show this help screen.
-j, --json [Term]      Show result in JSON format.
-m, --mirror [EDB-ID]  Mirror (aka copies) an exploit to the current
                        working directory.
-o, --overflow [Term]  Exploit titles are allowed to overflow their
                        columns.
-p, --path [EDB-ID]    Show the full path to an exploit (and also
                        copies the path to the clipboard if possible).
-t, --title [Term]     Search JUST the exploit title (Default is
                        title AND the file's path).
-u, --update           Check for and install any exploithub package
                        updates (deb or git).
-w, --www [Term]       Show URLs to Exploit-DB.com rather than the
                        local path.
-x, --examine [EDB-ID] Examine (aka opens) the exploit using $PAGER.
--colour              Disable colour highlighting in search results.
--id                  Display the EDB-ID value rather than local
                        path.
--nmap [file.xml]      Checks all results in Nmap's XML output with
                        service version (e.g.: nmap -sV -oX file.xml).
                        Use "-v" (verbose) to try even more
                        combinations
--exclude="term"       Remove values from results. By using "|" to
                        separated you can chain multiple values.
                        e.g. --exclude="term1|term2|term3".
```

=====

#### Notes

=====

- \* You can use any number of search terms.
- \* Search terms are not case-sensitive (by default), and ordering is irrelevant.
  - \* Use '-c' if you wish to reduce results by case-sensitive searching.
  - \* And/Or '-e' if you wish to filter results by using an exact match.
- \* Use '-t' to exclude the file's path to filter the search results.
- \* Remove false positives (especially when searching using numbers - i.e. versions).
- \* When updating or displaying help, search terms will be ignored.

### 4.1.29 giskismet

GISKismet is a wireless recon visualization tool to represent data gathered using Kismet in a flexible manner. GISKismet stores the information in a database so that the user can generate graphs using SQL.

---

```
[root@ArcheryOS ~]# giskismet --help
Usage: giskismet [Options]

Input File:
    --csv <csv-file>          Parse the input from Kismet-devel CSV
    -x --xml <xml-file>       Parse the input from Kismet-newcore
                                NETXML

Input Filters:
    --bssid file | list       Filter based on BSSID
    --essid file | list       Filter based on ESSID
    --encryption file | list  Filter based on Encryption
    --channel file | list     Filter based on Channel

file | list (list = comma separated lists(needs quotes))

Kismet-newcore Options:
    -a --ap                   Insert only the APs

Query
    -q --query [sql]         SQL query
    -m --manual [csv]        CSV output of manual SQL query

    -o --output [file]       Output filename
    -n --name [str]          Name of the KML layer
    --desc [str]             Description of the KML layer

General Options:
    --ignore-gps             Import data even when GPS fields are
                                missing
    --database [file]        SQLite3 database name [default:
                                wireless.db]
    -d --debug [num]         Display debug information
    -s --silent              No output when adding APs
    -v --version             Display version
    -h --help               Display this information
```

Send Comments to Joshua "Jabra" Abraham ( [jabra@spl0it.org](mailto:jabra@spl0it.org) )

---

### 4.1.30 hashcat

Hashcat is the worlds fastest CPU-based password recovery tool.

---

```
[root@ArcheryOS ~]# hashcat --help
hashcat - advanced password recovery
```

```
Usage: hashcat [options]... hash|hashfile|hccapxfile
       [dictionary|mask|directory]...
```

```
- [ Options ] -
```

Options	Short / Long	Type	Description	Example
-m, --hash-type		Num	Hash-type, see references below	
	-m 1000			
-a, --attack-mode		Num	Attack-mode, see references below	
	-a 3			
-V, --version			Print version	
-h, --help			Print help	
--quiet			Suppress output	
--hex-charset			Assume charset is given in hex	
--hex-salt			Assume salt is given in hex	
--hex-wordlist			Assume words in wordlist are given	
in hex				
--force			Ignore warnings	
--status			Enable automatic update of the	
status screen				
--status-timer		Num	Sets seconds between status screen	
updates to X				
--machine-readable			Display the status view in a	
machine-readable format				
--keep-guessing			Keep guessing the hash after it has	
been cracked				
--loopback			Add new plains to induct directory	
--weak-hash-threshold		Num	Threshold X when to stop checking	
for weak hashes				
--weak=0				
--markov-hcstat		File	Specify hcstat file to use	
--markov-hc=my.hcstat				
--markov-disable			Disables markov-chains, emulates	
classic brute-force				
--markov-classic			Enables classic markov-chains, no	
per-position				
-t, --markov-threshold		Num	Threshold X when to stop accepting	
new markov-chains				
-t 50				
--runtime		Num	Abort session after X seconds of	



```

runtime          | --runtime=10
--session         | Str | Define specific session name
                  | --session=mysession
--restore         |     | Restore session from --session
                  |
--restore-disable |     | Do not write restore file
                  |
--restore-file-path | File | Specific path to restore file
                  | --restore-file-path=my.restore
-o, --outfile     | File | Define outfile for recovered hash
                  | -o outfile.txt
--outfile-format  | Num | Define outfile-format X for
recovered hash   | --outfile-format=7
--outfile-autohex-disable | | Disable the use of $HEX[] in output
plains          |
--outfile-check-timer | Num | Sets seconds between outfile checks
to X            | --outfile-check=30
-p, --separator   | Char | Separator char for hashlists and
outfile          | -p :
--stdout          |     | Do not crack a hash, instead print
candidates only |
--show            |     | Compare hashlist with potfile; show
cracked hashes |
--left            |     | Compare hashlist with potfile; show
uncracked hashes |
--username        |     | Enable ignoring of usernames in
hashfile         |
--remove          |     | Enable removal of hashes once they
are cracked |
--remove-timer    | Num | Update input hash file each X
seconds          | --remove-timer=30
--potfile-disable |     | Do not write potfile
                  |
--potfile-path    | Dir | Specific path to potfile
                  | --potfile-path=my.pot
--encoding-from   | Code | Force internal wordlist encoding
from X           | --encoding-from=iso-8859-15
--encoding-to     | Code | Force internal wordlist encoding to
X                | --encoding-to=utf-32le
--debug-mode      | Num | Defines the debug mode (hybrid only
by using rules) | --debug-mode=4
--debug-file      | File | Output file for debugging rules
                  | --debug-file=good.log
--induction-dir   | Dir | Specify the induction directory to
use for loopback | --induction=inducts
--outfile-check-dir | Dir | Specify the outfile directory to
monitor for plains | --outfile-check-dir=x
--logfile-disable |     | Disable the logfile
                  |
--hccapx-message-pair | Num | Load only message pairs from hccapx

```

```

    matching X | --hccapx-message-pair=2
--nonce-error-corrections | Num | The BF size range to replace AP's
    nonce last bytes | --nonce-error-corrections=16
--truecrypt-keyfiles      | File | Keyfiles to use, separated with
    commas            | --truecrypt-key=x.png
--veracrypt-keyfiles      | File | Keyfiles to use, separated with
    commas            | --veracrypt-key=x.txt
--veracrypt-pim           | Num | VeraCrypt personal iterations
    multiplier        | --veracrypt-pim=1000
-b, --benchmark           |      | Run benchmark
                        |
--speed-only              |      | Return expected speed of the
    attack, then quit |
--progress-only           |      | Return ideal progress step size and
    time to process |
-c, --segment-size        | Num | Sets size in MB to cache from the
    wordfile to X | -c 32
--bitmap-min              | Num | Sets minimum bits allowed for
    bitmaps to X      | --bitmap-min=24
--bitmap-max              | Num | Sets maximum bits allowed for
    bitmaps to X      | --bitmap-max=24
--cpu-affinity             | Str | Locks to CPU devices, separated
    with commas       | --cpu-affinity=1,2,3
-I, --opencl-info          |      | Show info about detected OpenCL
    platforms/devices | -I
--opencl-platforms        | Str | OpenCL platforms to use, separated
    with commas       | --opencl-platforms=2
-d, --opencl-devices       | Str | OpenCL devices to use, separated
    with commas       | -d 1
-D, --opencl-device-types | Str | OpenCL device-types to use,
    separated with commas | -D 1
--opencl-vector-width      | Num | Manually override OpenCL
    vector-width to X    | --opencl-vector=4
-w, --workload-profile     | Num | Enable a specific workload profile,
    see pool below | -w 3
-n, --kernel-accel         | Num | Manual workload tuning, set
    outerloop step size to X | -n 64
-u, --kernel-loops         | Num | Manual workload tuning, set
    innerloop step size to X | -u 256
--nvidia-spin-damp         | Num | Workaround NVIDIAs CPU burning loop
    bug, in percent | --nvidia-spin-damp=50
--gpu-temp-disable         |      | Disable temperature and fanspeed
    reads and triggers |
--gpu-temp-abort           | Num | Abort if GPU temperature reaches X
    degrees Celsius | --gpu-temp-abort=100
--gpu-temp-retain          | Num | Try to retain GPU temperature at X
    degrees Celsius | --gpu-temp-retain=95
--powertune-enable         |      | Enable power tuning. Restores
    settings when finished |
--scrypt-tmto              | Num | Manually override TMTO value for

```

```

        script to X      | --script-tmto=3
-s, --skip                | Num | Skip X words from the start
                        | -s 1000000
-l, --limit                | Num | Limit X words from the start +
    skipped words      | -l 1000000
--keyspace                |      | Show keyspace base:mod values and
    quit                |
-j, --rule-left            | Rule | Single rule applied to each word
    from left wordlist | -j 'c'
-k, --rule-right            | Rule | Single rule applied to each word
    from right wordlist | -k '~'
-r, --rules-file            | File | Multiple rules applied to each word
    from wordlists | -r rules/best64.rule
-g, --generate-rules        | Num | Generate X random rules
                        | -g 10000
--generate-rules-func-min | Num | Force min X functions per rule
                        |
--generate-rules-func-max | Num | Force max X functions per rule
                        |
--generate-rules-seed     | Num | Force RNG seed set to X
                        |
-1, --custom-charset1      | CS  | User-defined charset ?1
                        | -1 ?l?d?u
-2, --custom-charset2      | CS  | User-defined charset ?2
                        | -2 ?l?d?s
-3, --custom-charset3      | CS  | User-defined charset ?3
                        |
-4, --custom-charset4      | CS  | User-defined charset ?4
                        |
-i, --increment            |      | Enable mask increment mode
                        |
--increment-min            | Num | Start mask incrementing at X
                        | --increment-min=4
--increment-max            | Num | Stop mask incrementing at X
                        | --increment-max=8

- [ Hash modes ] -

```

#	Name	Category
900	MD4	Raw Hash
0	MD5	Raw Hash
5100	Half MD5	Raw Hash
100	SHA1	Raw Hash
1300	SHA-224	Raw Hash
1400	SHA-256	Raw Hash
10800	SHA-384	Raw Hash
1700	SHA-512	Raw Hash
5000	SHA-3 (Keccak)	Raw Hash
600	BLAKE2b-512	Raw Hash

10100	SipHash	Raw Hash
6000	RIPEMD-160	Raw Hash
6100	Whirlpool	Raw Hash
6900	GOST R 34.11-94	Raw Hash
11700	GOST R 34.11-2012 (Streebog) 256-bit	Raw Hash
11800	GOST R 34.11-2012 (Streebog) 512-bit	Raw Hash
10	md5(\$pass.\$salt)	Raw Hash, Salted
	and/or Iterated	
20	md5(\$salt.\$pass)	Raw Hash, Salted
	and/or Iterated	
30	md5(utf16le(\$pass).\$salt)	Raw Hash, Salted
	and/or Iterated	
40	md5(\$salt.utf16le(\$pass))	Raw Hash, Salted
	and/or Iterated	
3800	md5(\$salt.\$pass.\$salt)	Raw Hash, Salted
	and/or Iterated	
3710	md5(\$salt.md5(\$pass))	Raw Hash, Salted
	and/or Iterated	
4010	md5(\$salt.md5(\$salt.\$pass))	Raw Hash, Salted
	and/or Iterated	
4110	md5(\$salt.md5(\$pass.\$salt))	Raw Hash, Salted
	and/or Iterated	
2600	md5(md5(\$pass))	Raw Hash, Salted
	and/or Iterated	
3910	md5(md5(\$pass).md5(\$salt))	Raw Hash, Salted
	and/or Iterated	
4300	md5(strtoupper(md5(\$pass)))	Raw Hash, Salted
	and/or Iterated	
4400	md5(sha1(\$pass))	Raw Hash, Salted
	and/or Iterated	
110	sha1(\$pass.\$salt)	Raw Hash, Salted
	and/or Iterated	
120	sha1(\$salt.\$pass)	Raw Hash, Salted
	and/or Iterated	
130	sha1(utf16le(\$pass).\$salt)	Raw Hash, Salted
	and/or Iterated	
140	sha1(\$salt.utf16le(\$pass))	Raw Hash, Salted
	and/or Iterated	
4500	sha1(sha1(\$pass))	Raw Hash, Salted
	and/or Iterated	
4520	sha1(\$salt.sha1(\$pass))	Raw Hash, Salted
	and/or Iterated	
4700	sha1(md5(\$pass))	Raw Hash, Salted
	and/or Iterated	
4900	sha1(\$salt.\$pass.\$salt)	Raw Hash, Salted
	and/or Iterated	
14400	sha1(CX)	Raw Hash, Salted
	and/or Iterated	
1410	sha256(\$pass.\$salt)	Raw Hash, Salted
	and/or Iterated	

1420   sha256(\$salt.\$pass)	Raw Hash, Salted
and/or Iterated	
1430   sha256(utf16le(\$pass).\$salt)	Raw Hash, Salted
and/or Iterated	
1440   sha256(\$salt.utf16le(\$pass))	Raw Hash, Salted
and/or Iterated	
1710   sha512(\$pass.\$salt)	Raw Hash, Salted
and/or Iterated	
1720   sha512(\$salt.\$pass)	Raw Hash, Salted
and/or Iterated	
1730   sha512(utf16le(\$pass).\$salt)	Raw Hash, Salted
and/or Iterated	
1740   sha512(\$salt.utf16le(\$pass))	Raw Hash, Salted
and/or Iterated	
50   HMAC-MD5 (key = \$pass)	Raw Hash,
Authenticated	
60   HMAC-MD5 (key = \$salt)	Raw Hash,
Authenticated	
150   HMAC-SHA1 (key = \$pass)	Raw Hash,
Authenticated	
160   HMAC-SHA1 (key = \$salt)	Raw Hash,
Authenticated	
1450   HMAC-SHA256 (key = \$pass)	Raw Hash,
Authenticated	
1460   HMAC-SHA256 (key = \$salt)	Raw Hash,
Authenticated	
1750   HMAC-SHA512 (key = \$pass)	Raw Hash,
Authenticated	
1760   HMAC-SHA512 (key = \$salt)	Raw Hash,
Authenticated	
14000   DES (PT = \$salt, key = \$pass)	Raw Cipher,
Known-Plaintext attack	
14100   3DES (PT = \$salt, key = \$pass)	Raw Cipher,
Known-Plaintext attack	
14900   Skip32 (PT = \$salt, key = \$pass)	Raw Cipher,
Known-Plaintext attack	
15400   ChaCha20	Raw Cipher,
Known-Plaintext attack	
400   phpass	Generic KDF
8900   scrypt	Generic KDF
11900   PBKDF2-HMAC-MD5	Generic KDF
12000   PBKDF2-HMAC-SHA1	Generic KDF
10900   PBKDF2-HMAC-SHA256	Generic KDF
12100   PBKDF2-HMAC-SHA512	Generic KDF
23   Skype	Network Protocols
2500   WPA/WPA2	Network Protocols
4800   iSCSI CHAP authentication, MD5(CHAP)	Network Protocols
5300   IKE-PSK MD5	Network Protocols
5400   IKE-PSK SHA1	Network Protocols
5500   NetNTLMv1	Network Protocols

5500	NetNTLMv1+ESS	Network Protocols
5600	NetNTLMv2	Network Protocols
7300	IPMI2 RAKP HMAC-SHA1	Network Protocols
7500	Kerberos 5 AS-REQ Pre-Auth etype 23	Network Protocols
8300	DNSSEC (NSEC3)	Network Protocols
10200	CRAM-MD5	Network Protocols
11100	PostgreSQL CRAM (MD5)	Network Protocols
11200	MySQL CRAM (SHA1)	Network Protocols
11400	SIP digest authentication (MD5)	Network Protocols
13100	Kerberos 5 TGS-REP etype 23	Network Protocols
121	SMF (Simple Machines Forum) > v1.1 E-Commerce, Frameworks	Forums, CMS,
400	phpBB3 (MD5) E-Commerce, Frameworks	Forums, CMS,
2611	vBulletin < v3.8.5 E-Commerce, Frameworks	Forums, CMS,
2711	vBulletin >= v3.8.5 E-Commerce, Frameworks	Forums, CMS,
2811	MyBB 1.2+ E-Commerce, Frameworks	Forums, CMS,
2811	IPB2+ (Invision Power Board) E-Commerce, Frameworks	Forums, CMS,
8400	WBB3 (Woltlab Burning Board) E-Commerce, Frameworks	Forums, CMS,
11	Joomla < 2.5.18 E-Commerce, Frameworks	Forums, CMS,
400	Joomla >= 2.5.18 (MD5) E-Commerce, Frameworks	Forums, CMS,
400	WordPress (MD5) E-Commerce, Frameworks	Forums, CMS,
2612	PHPS E-Commerce, Frameworks	Forums, CMS,
7900	Drupal7 E-Commerce, Frameworks	Forums, CMS,
21	osCommerce E-Commerce, Frameworks	Forums, CMS,
21	xt:Commerce E-Commerce, Frameworks	Forums, CMS,
11000	PrestaShop E-Commerce, Frameworks	Forums, CMS,
124	Django (SHA-1) E-Commerce, Frameworks	Forums, CMS,
10000	Django (PBKDF2-SHA256) E-Commerce, Frameworks	Forums, CMS,
3711	MediaWiki B <a href="#">type</a> E-Commerce, Frameworks	Forums, CMS,
13900	OpenCart E-Commerce, Frameworks	Forums, CMS,
4521	Redmine E-Commerce, Frameworks	Forums, CMS,

4522	PunBB	Forums, CMS,
	E-Commerce, Frameworks	
12001	Atlassian (PBKDF2-HMAC-SHA1)	Forums, CMS,
	E-Commerce, Frameworks	
12	PostgreSQL	Database Server
131	MSSQL (2000)	Database Server
132	MSSQL (2005)	Database Server
1731	MSSQL (2012, 2014)	Database Server
200	MySQL323	Database Server
300	MySQL4.1/MySQL5	Database Server
3100	Oracle H: Type (Oracle 7+)	Database Server
112	Oracle S: Type (Oracle 11+)	Database Server
12300	Oracle T: Type (Oracle 12+)	Database Server
8000	Sybase ASE	Database Server
141	Episerver 6.x < .NET 4 Server	HTTP, SMTP, LDAP
1441	Episerver 6.x >= .NET 4 Server	HTTP, SMTP, LDAP
1600	Apache \$apri\$ MD5, md5apri, MD5 (APR) Server	HTTP, SMTP, LDAP
12600	ColdFusion 10+ Server	HTTP, SMTP, LDAP
1421	hMailServer Server	HTTP, SMTP, LDAP
101	nsldap, SHA-1(Base64), Netscape LDAP SHA Server	HTTP, SMTP, LDAP
111	nsldaps, SSHA-1(Base64), Netscape LDAP SSHA Server	HTTP, SMTP, LDAP
1411	SSHA-256(Base64), LDAP {SSHA256} Server	HTTP, SMTP, LDAP
1711	SSHA-512(Base64), LDAP {SSHA512} Server	HTTP, SMTP, LDAP
15000	FileZilla Server >= 0.9.55	FTP Server
11500	CRC32	Checksums
3000	LM	Operating Systems
1000	NTLM	Operating Systems
1100	Domain Cached Credentials (DCC), MS Cache	Operating Systems
2100	Domain Cached Credentials 2 (DCC2), MS Cache 2 Systems	Operating Systems
15300	DPAPI masterkey file v1 and v2	Operating Systems
12800	MS-AzureSync PBKDF2-HMAC-SHA256	Operating Systems
1500	descrypt, DES (Unix), Traditional DES	Operating Systems
12400	BSDi Crypt, Extended DES	Operating Systems
500	md5crypt, MD5 (Unix), Cisco-IOS \$1\$ (MD5)	Operating Systems
3200	bcrypt \$2*\$, Blowfish (Unix)	Operating Systems
7400	sha256crypt \$5\$, SHA256 (Unix)	Operating Systems
1800	sha512crypt \$6\$, SHA512 (Unix)	Operating Systems
122	OSX v10.4, OSX v10.5, OSX v10.6	Operating Systems
1722	OSX v10.7	Operating Systems
7100	OSX v10.8+ (PBKDF2-SHA512)	Operating Systems

6300	AIX {smd5}	Operating Systems
6700	AIX {ssh1}	Operating Systems
6400	AIX {ssh256}	Operating Systems
6500	AIX {ssh512}	Operating Systems
2400	Cisco-PIX MD5	Operating Systems
2410	Cisco-ASA MD5	Operating Systems
500	Cisco-IOS \$1\$ (MD5)	Operating Systems
5700	Cisco-IOS type 4 (SHA256)	Operating Systems
9200	Cisco-IOS \$\$ (PBKDF2-SHA256)	Operating Systems
9300	Cisco-IOS \$9\$ (scrypt)	Operating Systems
22	Juniper NetScreen/SSG (ScreenOS)	Operating Systems
501	Juniper IVE	Operating Systems
15100	Juniper/NetBSD sha1crypt	Operating Systems
7000	FortiGate (FortiOS)	Operating Systems
5800	Samsung Android Password/PIN	Operating Systems
13800	Windows Phone 8+ PIN/password	Operating Systems
8100	Citrix NetScaler	Operating Systems
8500	RACF	Operating Systems
7200	GRUB 2	Operating Systems
9900	Radmin2	Operating Systems
125	ArubaOS	Operating Systems
7700	SAP CODVN B (BCODE)	Enterprise
	Application Software (EAS)	
7800	SAP CODVN F/G (PASSCODE)	Enterprise
	Application Software (EAS)	
10300	SAP CODVN H (PWDSALTEDHASH) iSSHA-1	Enterprise
	Application Software (EAS)	
8600	Lotus Notes/Domino 5	Enterprise
	Application Software (EAS)	
8700	Lotus Notes/Domino 6	Enterprise
	Application Software (EAS)	
9100	Lotus Notes/Domino 8	Enterprise
	Application Software (EAS)	
133	PeopleSoft	Enterprise
	Application Software (EAS)	
13500	PeopleSoft PS_TOKEN	Enterprise
	Application Software (EAS)	
11600	7-Zip	Archives
12500	RAR3-hp	Archives
13000	RAR5	Archives
13200	AxCrypt	Archives
13300	AxCrypt in-memory SHA1	Archives
13600	WinZip	Archives
14700	iTunes backup < 10.0	Backup
14800	iTunes backup >= 10.0	Backup
62XY	TrueCrypt	Full-Disk
	Encryption (FDE)	
X	1 = PBKDF2-HMAC-RIPEMD160	Full-Disk
	Encryption (FDE)	
X	2 = PBKDF2-HMAC-SHA512	Full-Disk



	Encryption (FDE)	
X	3 = PBKDF2-HMAC-Whirlpool	Full-Disk
	Encryption (FDE)	
X	4 = PBKDF2-HMAC-RIPEMD160 + boot-mode	Full-Disk
	Encryption (FDE)	
Y	1 = XTS 512 bit pure AES	Full-Disk
	Encryption (FDE)	
Y	1 = XTS 512 bit pure Serpent	Full-Disk
	Encryption (FDE)	
Y	1 = XTS 512 bit pure Twofish	Full-Disk
	Encryption (FDE)	
Y	2 = XTS 1024 bit pure AES	Full-Disk
	Encryption (FDE)	
Y	2 = XTS 1024 bit pure Serpent	Full-Disk
	Encryption (FDE)	
Y	2 = XTS 1024 bit pure Twofish	Full-Disk
	Encryption (FDE)	
Y	2 = XTS 1024 bit cascaded AES-Twofish	Full-Disk
	Encryption (FDE)	
Y	2 = XTS 1024 bit cascaded Serpent-AES	Full-Disk
	Encryption (FDE)	
Y	2 = XTS 1024 bit cascaded Twofish-Serpent	Full-Disk
	Encryption (FDE)	
Y	3 = XTS 1536 bit all	Full-Disk
	Encryption (FDE)	
8800	Android FDE <= 4.3	Full-Disk
	Encryption (FDE)	
12900	Android FDE (Samsung DEK)	Full-Disk
	Encryption (FDE)	
12200	eCryptfs	Full-Disk
	Encryption (FDE)	
137XY	VeraCrypt	Full-Disk
	Encryption (FDE)	
X	1 = PBKDF2-HMAC-RIPEMD160	Full-Disk
	Encryption (FDE)	
X	2 = PBKDF2-HMAC-SHA512	Full-Disk
	Encryption (FDE)	
X	3 = PBKDF2-HMAC-Whirlpool	Full-Disk
	Encryption (FDE)	
X	4 = PBKDF2-HMAC-RIPEMD160 + boot-mode	Full-Disk
	Encryption (FDE)	
X	5 = PBKDF2-HMAC-SHA256	Full-Disk
	Encryption (FDE)	
X	6 = PBKDF2-HMAC-SHA256 + boot-mode	Full-Disk
	Encryption (FDE)	
Y	1 = XTS 512 bit pure AES	Full-Disk
	Encryption (FDE)	
Y	1 = XTS 512 bit pure Serpent	Full-Disk
	Encryption (FDE)	
Y	1 = XTS 512 bit pure Twofish	Full-Disk

Encryption (FDE)		
Y	2 = XTS 1024 bit pure AES	Full-Disk
Encryption (FDE)		
Y	2 = XTS 1024 bit pure Serpent	Full-Disk
Encryption (FDE)		
Y	2 = XTS 1024 bit pure Twofish	Full-Disk
Encryption (FDE)		
Y	2 = XTS 1024 bit cascaded AES-Twofish	Full-Disk
Encryption (FDE)		
Y	2 = XTS 1024 bit cascaded Serpent-AES	Full-Disk
Encryption (FDE)		
Y	2 = XTS 1024 bit cascaded Twofish-Serpent	Full-Disk
Encryption (FDE)		
Y	3 = XTS 1536 bit all	Full-Disk
Encryption (FDE)		
14600	LUKS	Full-Disk
Encryption (FDE)		
9700	MS Office <= 2003 \$0/\$1, MD5 + RC4	Documents
9710	MS Office <= 2003 \$0/\$1, MD5 + RC4, collider #1	Documents
9720	MS Office <= 2003 \$0/\$1, MD5 + RC4, collider #2	Documents
9800	MS Office <= 2003 \$3/\$4, SHA1 + RC4	Documents
9810	MS Office <= 2003 \$3, SHA1 + RC4, collider #1	Documents
9820	MS Office <= 2003 \$3, SHA1 + RC4, collider #2	Documents
9400	MS Office 2007	Documents
9500	MS Office 2010	Documents
9600	MS Office 2013	Documents
10400	PDF 1.1 - 1.3 (Acrobat 2 - 4)	Documents
10410	PDF 1.1 - 1.3 (Acrobat 2 - 4), collider #1	Documents
10420	PDF 1.1 - 1.3 (Acrobat 2 - 4), collider #2	Documents
10500	PDF 1.4 - 1.6 (Acrobat 5 - 8)	Documents
10600	PDF 1.7 Level 3 (Acrobat 9)	Documents
10700	PDF 1.7 Level 8 (Acrobat 10 - 11)	Documents
9000	Password Safe v2	Password Managers
5200	Password Safe v3	Password Managers
6800	LastPass + LastPass sniffed	Password Managers
6600	1Password, agilekeychain	Password Managers
8200	1Password, cloudkeychain	Password Managers
11300	Bitcoin/Litecoin wallet.dat	Password Managers
12700	Blockchain, My Wallet	Password Managers
15200	Blockchain, My Wallet, V2	Password Managers
13400	KeePass 1 (AES/Twofish) and KeePass 2 (AES)	Password Managers
15500	JKS Java Key Store Private Keys (SHA1)	Password Managers
15600	Ethereum Wallet, PBKDF2-HMAC-SHA256	Password Managers
15700	Ethereum Wallet, SCRYPT	Password Managers
99999	Plaintext	Plaintext
- [ Outfile Formats ] -		
#   Format		
====+=====		

```

1 | hash[:salt]
2 | plain
3 | hash[:salt]:plain
4 | hex_plain
5 | hash[:salt]:hex_plain
6 | plain:hex_plain
7 | hash[:salt]:plain:hex_plain
8 | crackpos
9 | hash[:salt]:crack_pos
10 | plain:crack_pos
11 | hash[:salt]:plain:crack_pos
12 | hex_plain:crack_pos
13 | hash[:salt]:hex_plain:crack_pos
14 | plain:hex_plain:crack_pos
15 | hash[:salt]:plain:hex_plain:crack_pos

- [ Rule Debugging Modes ] -

# | Format
===+=====
1 | Finding-Rule
2 | Original-Word
3 | Original-Word:Finding-Rule
4 | Original-Word:Finding-Rule:Processed-Word

- [ Attack Modes ] -

# | Mode
===+=====
0 | Straight
1 | Combination
3 | Brute-force
6 | Hybrid Wordlist + Mask
7 | Hybrid Mask + Wordlist

- [ Built-in Charsets ] -

? | Charset
===+=====
l | abcdefghijklmnopqrstuvwxyz
u | ABCDEFGHIJKLMNOPQRSTUVWXYZ
d | 0123456789
h | 0123456789abcdef
H | 0123456789ABCDEF
s | !"#$%&'()*+,-./:;<=>?@[\]^_`{|}~
a | ?l?u?d?s
b | 0x00 - 0xff

- [ OpenCL Device Types ] -

```

```

# | Device Type
===+=====
1 | CPU
2 | GPU
3 | FPGA, DSP, Co-Processor

- [ Workload Profiles ] -

# | Performance | Runtime | Power Consumption | Desktop Impact
===+=====+=====+=====+=====
1 | Low          | 2 ms   | Low               | Minimal
2 | Default      | 12 ms  | Economic          | Noticeable
3 | High         | 96 ms  | High              | Unresponsive
4 | Nightmare    | 480 ms | Insane            | Headless

- [ Basic Examples ] -

Attack-      | Hash- |
Mode         | Type  | Example command
=====+=====+=====
Wordlist      | $P$   | hashcat -a 0 -m 400 example400.hash
              |       | example.dict
Wordlist + Rules | MD5   | hashcat -a 0 -m 0 example0.hash example.dict
              |       | -r rules/best64.rule
Brute-Force   | MD5   | hashcat -a 3 -m 0 example0.hash ?a?a?a?a?a
Combinator    | MD5   | hashcat -a 1 -m 0 example0.hash example.dict
              |       | example.dict

If you still have no idea what just happened, try the following pages:

* https://hashcat.net/wiki/#howtos\_videos\_papers\_articles\_etc\_in\_the\_wild
* https://hashcat.net/faq/

```

#### 4.1.31 hashdeep

hashdeep computes multiple hashes, or message digests, for any number of files while optionally recursively digging through the directory structure. By default the program computes MD5 and SHA-256 hashes, equivalent to `-c md5,sha256`.

```

[root@ArcheryOS ~]# hashdeep -h
hashdeep version 4.4 by Jesse Kornblum and Simson Garfinkel.
$ hashdeep [OPTION]... [FILES]...
-c <alg1,[alg2]> - Compute hashes only. Defaults are MD5 and SHA-256
                  legal values: md5,sha1,sha256,tiger,whirlpool,
-p <size> - piecewise mode. Files are broken into blocks for hashing
-r          - recursive mode. All subdirectories are traversed
-d          - output in DFXML (Digital Forensics XML)
-k <file> - add a file of known hashes

```

---

```

-a      - audit mode. Validates FILES against known hashes. Requires -k
-m      - matching mode. Requires -k
-x      - negative matching mode. Requires -k
-w      - in -m mode, displays which known file was matched
-M and -X act like -m and -x, but display hashes of matching files
-e      - compute estimated time remaining for each file
-s      - silent mode. Suppress all error messages
-b      - prints only the bare name of files; all path information is
        omitted
-l      - print relative paths for filenames
-i/-I   - only process files smaller than the given threshold
-o      - only process certain types of files. See README/manpage
-v      - verbose mode. Use again to be more verbose
-d      - output in DFXML; -W FILE - write to FILE.
-j <num> - use num threads (default 1)

```

---

#### 4.1.32 httrack

httrack - offline browser : copy websites to a local directory

---

```
[root@ArcheryOS ~]# httrack --help
```

HTTrack version 3.49-2

```

usage: httrack <URLs> [-option] [+<URL_FILTER>] [-<URL_FILTER>]
      [+<mime:MIME_FILTER>] [-<mime:MIME_FILTER>]
with options listed below: (* is the default value)

```

General options:

```

O path for mirror/logfiles+cache (-O
  path_mirror[,path_cache_and_logfiles]) (--path <param>)

```

Action options:

```

w *mirror web sites (--mirror)
W mirror web sites, semi-automatic (asks questions) (--mirror-wizard)
g just get files (saved in the current directory) (--get-files)
i continue an interrupted mirror using the cache (--continue)
Y mirror ALL links located in the first level pages (mirror links)
  (--mirrorlinks)

```

Proxy options:

```

P proxy use (-P proxy:port or -P user:pass@proxy:port) (--proxy
  <param>)
%f *use proxy for ftp (f0 don't use) (--httpproxy-ftp[=N])
%b use this local hostname to make/send requests (-%b hostname)
  (--bind <param>)

```

Limits options:

```

rN set the mirror depth to N (* r9999) (--depth[=N])
%eN set the external links depth to N (* %e0) (--ext-depth[=N])

```

```

mN maximum file length for a non-html file (--max-files[=N])
mN,N2 maximum file length for non html (N) and html (N2)
MN maximum overall size that can be uploaded/scanned (--max-size[=N])
EN maximum mirror time in seconds (60=1 minute, 3600=1 hour)
    (--max-time[=N])
AN maximum transfer rate in bytes/seconds (1000=1KB/s max)
    (--max-rate[=N])
%cN maximum number of connections/seconds (*%c10)
    (--connection-per-second[=N])
GN pause transfer if N bytes reached, and wait until lock file is
    deleted (--max-pause[=N])

Flow control:
cN number of multiple connections (*c8) (--sockets[=N])
TN timeout, number of seconds after a non-responding link is shutdown
    (--timeout[=N])
RN number of retries, in case of timeout or non-fatal errors (*R1)
    (--retries[=N])
JN traffic jam control, minimum transfert rate (bytes/seconds)
    tolerated for a link (--min-rate[=N])
HN host is abandonned if: 0=never, 1=timeout, 2=slow, 3=timeout or
    slow (--host-control[=N])

Links options:
%P *extended parsing, attempt to parse all links, even in unknown tags
    or Javascript (%P0 don't use) (--extended-parsing[=N])
n get non-html files 'near' an html file (ex: an image located
    outside) (--near)
t test all URLs (even forbidden ones) (--test)
%L <file> add all URL located in this text file (one URL per line)
    (--list <param>)
%S <file> add all scan rules located in this text file (one scan rule
    per line) (--urllist <param>)

Build options:
NN structure type (0 *original structure, 1+: see below)
    (--structure[=N])
    or user defined structure (-N "%h%p/%n%q.%t")
%N delayed type check, don't make any link test but wait for files
    download to start instead (experimental) (%N0 don't use, %N1 use
    for unknown extensions, * %N2 always use)
%D cached delayed type check, don't wait for remote type during
    updates, to speedup them (%D0 wait, * %D1 don't wait)
    (--cached-delayed-type-check)
%M generate a RFC MIME-encapsulated full-archive (.mht) (--mime-html)
LN long names (L1 *long names / L0 8-3 conversion / L2 ISO9660
    compatible) (--long-names[=N])
KN keep original links (e.g. http://www.adr/link) (K0 *relative link,
    K absolute links, K4 original links, K3 absolute URI links, K5
    transparent proxy link) (--keep-links[=N])

```

```

x replace external html links by error pages (--replace-external)
%x do not include any password for external password protected
  websites (%x0 include) (--disable-passwords)
%q *include query string for local files (useless, for information
  purpose only) (%q0 don't include) (--include-query-string)
o *generate output html file in case of error (404..) (o0 don't
  generate) (--generate-errors)
X *purge old files after update (X0 keep delete) (--purge-old[=N])
%p preserve html files 'as is' (identical to '-K4 -%F ""') (--preserve)
%T links conversion to UTF-8 (--utf8-conversion)

```

## Spider options:

```

bN accept cookies in cookies.txt (0=do not accept,* 1=accept)
  (--cookies[=N])
u check document type if unknown (cgi,asp..) (u0 don't check, * u1
  check but /, u2 check always) (--check-type[=N])
j *parse Java Classes (j0 don't parse, bitmask: |1 parse default, |2
  don't parse .class |4 don't parse .js |8 don't be aggressive)
  (--parse-java[=N])
sN follow robots.txt and meta robots tags (0=never,1=sometimes,*
  2=always, 3=always (even strict rules)) (--robots[=N])
%h force HTTP/1.0 requests (reduce update features, only for old
  servers or proxies) (--http-10)
%k use keep-alive if possible, greatly reducing latency for small
  files and test requests (%k0 don't use) (--keep-alive)
%B tolerant requests (accept bogus responses on some servers, but not
  standard!) (--tolerant)
%s update hacks: various hacks to limit re-transfers when updating
  (identical size, bogus response..) (--updatehack)
%u url hacks: various hacks to limit duplicate URLs (strip //,
  www.foo.com==foo.com..) (--urlhack)
%A assume that a type (cgi,asp..) is always linked with a mime type
  (-%A php3,cgi=text/html;dat,bin=application/x-zip) (--assume
  <param>)
  shortcut: '--assume standard' is equivalent to -%A php2 php3 php4
    php cgi asp jsp pl cfm nsf=text/html
  can also be used to force a specific file type: --assume
    foo.cgi=text/html
%iN internet protocol (0=both ipv6+ipv4, 4=ipv4 only, 6=ipv6 only)
  (--protocol[=N])
%w disable a specific external mime module (-%w htsswf -%w htsjava)
  (--disable-module <param>)

```

## Browser ID:

```

F user-agent field sent in HTTP headers (-F "user-agent name")
  (--user-agent <param>)
%R default referer field sent in HTTP headers (--referer <param>)
%E from email address sent in HTTP headers (--from <param>)
%F footer string in Html code (-%F "Mirrored [from host %s [file %s
  [at %s]]]" (--footer <param>)

```

```
%l preferred language (-%l "fr, en, jp, *" (--language <param>)
%a accepted formats (-%a "text/html,image/png;q=0.9,*/*;q=0.1"
    (--accept <param>)
%X additional HTTP header line (-%X "X-Magic: 42" (--headers <param>)
```

#### Log, index, cache

```
C create/use a cache for updates and retries (C0 no cache,C1 cache is
    priority,* C2 test update before) (--cache[=N])
k store all files in cache (not useful if files on disk)
    (--store-all-in-cache)
%n do not re-download locally erased files (--do-not-recatch)
%v display on screen filenames downloaded (in realtime) - * %v1 short
    version - %v2 full animation (--display)
Q no log - quiet mode (--do-not-log)
q no questions - quiet mode (--quiet)
z log - extra infos (--extra-log)
Z log - debug (--debug-log)
v log on screen (--verbose)
f *log in files (--file-log)
f2 one single log file (--single-log)
I *make an index (I0 don't make) (--index)
%i make a top index for a project folder (* %i0 don't make)
    (--build-top-index)
%I make an searchable index for this mirror (* %I0 don't make)
    (--search-index)
```

#### Expert options:

```
pN priority mode: (* p3) (--priority[=N])
    p0 just scan, don't save anything (for checking links)
    p1 save only html files
    p2 save only non html files
    *p3 save all files
    p7 get html files before, then treat other files
S stay on the same directory (--stay-on-same-dir)
D *can only go down into subdirs (--can-go-down)
U can only go to upper directories (--can-go-up)
B can both go up&down into the directory structure
    (--can-go-up-and-down)
a *stay on the same address (--stay-on-same-address)
d stay on the same principal domain (--stay-on-same-domain)
l stay on the same TLD (eg: .com) (--stay-on-same-tld)
e go everywhere on the web (--go-everywhere)
%H debug HTTP headers in logfile (--debug-headers)
```

#### Guru options: (do NOT use if possible)

```
#X *use optimized engine (limited memory boundary checks)
    (--fast-engine)
#0 filter test (-#0 '*.gif' 'www.bar.com/foo.gif')
    (--debug-testfilters <param>)
#1 simplify test (-#1 ./foo/bar/./foobar)
```



```

#2 type test (-#2 /foo/bar.php)
#C cache list (-#C '*.com/spider*.gif' (--debug-cache <param>))
#R cache repair (damaged cache) (--repair-cache)
#d debug parser (--debug-parsing)
#E extract new.zip cache meta-data in meta.zip
#f always flush log files (--advanced-flushlogs)
#FN maximum number of filters (--advanced-maxfilters[=N])
#h version info (--version)
#K scan stdin (debug) (--debug-scanstdin)
#L maximum number of links (-#L1000000) (--advanced-maxlinks[=N])
#p display ugly progress information (--advanced-progressinfo)
#P catch URL (--catch-url)
#R old FTP routines (debug) (--repair-cache)
#T generate transfer ops. log every minutes (--debug-xfrstats)
#u wait time (--advanced-wait)
#Z generate transfer rate statistics every minutes (--debug-ratestats)

```

Dangerous options: (do NOT use unless you exactly know what you are doing)

```

%! bypass built-in security limits aimed to avoid bandwidth abuses
  (bandwidth, simultaneous connections) (--disable-security-limits)
  IMPORTANT NOTE: DANGEROUS OPTION, ONLY SUITABLE FOR EXPERTS
  USE IT WITH EXTREME CARE

```

Command-line specific options:

```

V execute system command after each files ($0 is the filename: -V "rm
  \b$0") (--userdef-cmd <param>)
%W use an external library function as a wrapper (-%W
  myfoo.so[,myparameters]) (--callback <param>)

```

Details: Option N

```

N0 Site-structure (default)
N1 HTML in web/, images/other files in web/images/
N2 HTML in web/HTML, images/other in web/images
N3 HTML in web/, images/other in web/
N4 HTML in web/, images/other in web/xxx, where xxx is the file
  extension (all gif will be placed onto web/gif, for example)
N5 Images/other in web/xxx and HTML in web/HTML
N99 All files in web/, with random names (gadget !)
N100 Site-structure, without www.domain.xxx/
N101 Identical to N1 except that "web" is replaced by the site's name
N102 Identical to N2 except that "web" is replaced by the site's name
N103 Identical to N3 except that "web" is replaced by the site's name
N104 Identical to N4 except that "web" is replaced by the site's name
N105 Identical to N5 except that "web" is replaced by the site's name
N199 Identical to N99 except that "web" is replaced by the site's name
N1001 Identical to N1 except that there is no "web" directory
N1002 Identical to N2 except that there is no "web" directory
N1003 Identical to N3 except that there is no "web" directory (option
  set for g option)

```

N1004 Identical to N4 except that there is no "web" directory  
 N1005 Identical to N5 except that there is no "web" directory  
 N1099 Identical to N99 except that there is no "web" directory  
 Details: User-defined option N  
 '%n' Name of file without file type (ex: image)  
 '%N' Name of file, including file type (ex: image.gif)  
 '%t' File type (ex: gif)  
 '%p' Path [without ending /] (ex: /someimages)  
 '%h' Host name (ex: www.someweb.com)  
 '%M' URL MD5 (128 bits, 32 ascii bytes)  
 '%Q' query string MD5 (128 bits, 32 ascii bytes)  
 '%k' full query string  
 '%r' protocol name (ex: http)  
 '%q' small query string MD5 (16 bits, 4 ascii bytes)  
 '%s?' Short name version (ex: %sN)  
 '%[param]' param variable in query string  
 '%[param:before:after:empty:notfound]' advanced variable extraction  
 Details: User-defined option N and advanced variable extraction  
 %[param:before:after:empty:notfound]  
 param : parameter name  
 before : string to prepend if the parameter was found  
 after : string to append if the parameter was found  
 notfound : string replacement if the parameter could not be found  
 empty : string replacement if the parameter was empty  
 all fields, except the first one (the parameter name), can be empty  
  
 Details: Option K  
 K0 foo.cgi?q=45 -> foo4B54.html?q=45 (relative URI, default)  
 K -> http://www.foobar.com/folder/foo.cgi?q=45  
 (absolute URL) (--keep-links[=N])  
 K3 -> /folder/foo.cgi?q=45 (absolute URI)  
 K4 -> foo.cgi?q=45 (original URL)  
 K5 -> http://www.foobar.com/folder/foo4B54.html?q=45  
 (transparent proxy URL)  
  
 Shortcuts:  
 --mirror <URLs> \*make a mirror of site(s) (default)  
 --get <URLs> get the files indicated, do not seek other URLs (-qg)  
 --list <text file> add all URL located in this text file (-%L)  
 --mirrorlinks <URLs> mirror all links in 1st level pages (-Y)  
 --testlinks <URLs> test links in pages (-r1p0C0I0t)  
 --spider <URLs> spider site(s), to test links: reports Errors &  
 Warnings (-p0C0I0t)  
 --testsite <URLs> identical to --spider  
 --skeleton <URLs> make a mirror, but gets only html files (-p1)  
 --update update a mirror, without confirmation (-iC2)  
 --continue continue a mirror, without confirmation (-iC1)  
  
 --catchurl create a temporary proxy to capture an URL or a form  
 post URL

--clean                    erase cache & log files

--http10                  force http/1.0 requests (-%h)

Details: Option %W: External callbacks prototypes  
see htsdefines.h

example: httrack www.someweb.com/bob/  
means: mirror site www.someweb.com/bob/ and only this site

example: httrack www.someweb.com/bob/ www.anothertest.com/mike/  
          +\*.com/\*.jpg -mime:application/\*  
means: mirror the two sites together (with shared links) and accept  
any .jpg files on .com sites

example: httrack www.someweb.com/bob/bobby.html +\* -r6  
means get all files starting from bobby.html, with 6 link-depth, and  
possibility of going everywhere on the web

example: httrack www.someweb.com/bob/bobby.html --spider -P  
          proxy.myhost.com:8080  
runs the spider on www.someweb.com/bob/bobby.html using a proxy

example: httrack --update  
updates a mirror [in](#) the current folder

example: httrack  
will bring you to the interactive mode

example: httrack --continue  
continues a mirror [in](#) the current folder

HTTrack version 3.49-2  
Copyright (C) 1998-2017 Xavier Roche and other contributors

---

#### 4.1.33 hydra

hydra is a very fast network logon cracker which support many different services

---

[root@ArcheryOS ~]# **hydra -h**  
Hydra v8.6 (c) 2017 by van Hauser/THC - Please **do** not use [in](#) military or  
secret service organizations, or [for](#) illegal purposes.

Syntax: hydra [[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e  
nsr] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME]  
[-f] [-s PORT] [-x MIN:MAX:CHARSET] [-c TIME] [-ISOuvVd46]  
[service://server[:PORT] [/OPT]]

Options:

```

-R      restore a previous aborted/crashed session
-I      ignore an existing restore file (don't wait 10 seconds)
-S      perform an SSL connect
-s PORT if the service is on a different default port, define it here
-l LOGIN or -L FILE login with LOGIN name, or load several logins from
    FILE
-p PASS or -P FILE try password PASS, or load several passwords from
    FILE
-x MIN:MAX:CHARSET password bruteforce generation, type "-x -h" to get
    help
-y      disable use of symbols in bruteforce, see above
-e nsr  try "n" null password, "s" login as pass and/or "r" reversed
    login
-u      loop around users, not passwords (effective! implied with -x)
-C FILE colon separated "login:pass" format, instead of -L/-P options
-M FILE list of servers to attack, one entry per line, ':' to specify
    port
-o FILE write found login/password pairs to FILE instead of stdout
-b FORMAT specify the format for the -o FILE: text(default), json,
    jsonv1
-f / -F exit when a login/pass pair is found (-M: -f per host, -F
    global)
-t TASKS run TASKS number of connects in parallel per target (default:
    16)
-T TASKS run TASKS connects in parallel overall (for -M, default: 64)
-w / -W TIME wait time for a response (32) / between connects per
    thread (0)
-c TIME wait time per login attempt over all threads (enforces -t 1)
-4 / -6 use IPv4 (default) / IPv6 addresses (put always in [] also in
    -M)
-v / -V / -d verbose mode / show login+pass for each attempt / debug
    mode
-O      use old SSL v2 and v3
-q      do not print messages about connection errors
-U      service module usage details
-h      more command line options (COMPLETE HELP)
server  the target: DNS, IP or 192.168.0.0/24 (this OR the -M option)
service the service to crack (see below for supported protocols)
OPT     some service modules support additional input (-U for module
    help)

```

Supported services: adam6500 asterisk cisco cisco-enable cvs firebird  
ftp ftps http[s]-{head|get|post} http[s]-{get|post}-form http-proxy  
http-proxy-urlenum icq imap[s] irc ldap2[s]  
ldap3[-{cram|digest}md5][s] mssql mysql nntp oracle-listener  
oracle-sid pcanywhere pcnfs pop3[s] postgres radmin2 rdp redis  
rexec rlogin rpcap rsh rtsp s7-300 sip smb smtp[s] smtp-enum snmp  
socks5 ssh sshkey svn teamspeak telnet[s] vmauthd vnc xmpp

Hydra is a tool to guess/crack valid login/password pairs. Licensed

---

under AGPL  
v3.0. The newest version is always available at  
<http://www.thc.org/thc-hydra>  
Don't use **in** military or secret service organizations, or **for** illegal  
purposes.  
These services were not compiled **in**: afp ncp oracle sapr3.

Use HYDRA\_PROXY\_HTTP or HYDRA\_PROXY environment variables **for** a proxy  
setup.  
E.g. % **export** HYDRA\_PROXY=socks5://l:p@127.0.0.1:9150 (or: socks4://  
connect://)  
% **export** HYDRA\_PROXY=connect\_and\_socks\_proxylist.txt (up to 64  
entries)  
% **export** HYDRA\_PROXY\_HTTP=http://login:pass@proxy:8080  
% **export** HYDRA\_PROXY\_HTTP=proxylist.txt (up to 64 entries)

Examples:

```
hydra -l user -P passlist.txt ftp://192.168.0.1
hydra -L userlist.txt -p defaultpw imap://192.168.0.1/PLAIN
hydra -C defaults.txt -6 pop3s://[2001:db8::1]:143/TLS:DIGEST-MD5
hydra -l admin -p password ftp://[192.168.0.0/24]/
hydra -L logins.txt -P pws.txt -M targets.txt ssh
```

---

#### 4.1.34 intrace

InTrace is a traceroute-like application that enables users to enumerate IP hops exploiting existing TCP connections, both initiated from local network (local system) or from remote hosts.

---

```
[root@ArcheryOS ~]# intrace
InTrace, version 1.5 (C)2007-2011 Robert Swiecki <robert@swiecki.net>
2018/06/02 19:55:42.796514 <INFO> Usage: intrace <-h hostname> [-p
    <port>] [-d <debuglevel>] [-s <payloadsize>] [-6]
```

---

#### 4.1.35 john

john, better known as John the Ripper, is a tool to find weak passwords of users in a server. John can use a dictionary or some search pattern as well as a password file to check for passwords. John supports different cracking modes and understands many ciphertext formats, like several DES variants, MD5 and blowfish. It can also be used to extract AFS and Windows NT passwords.

---

```
[root@ArcheryOS ~]# john -h
Created directory: /root/.john
John the Ripper password cracker, version 1.8.0.6-jumbo-1-bleeding
[linux-x86-64-avx]
Copyright (c) 1996-2015 by Solar Designer and others
```

---

Homepage: <http://www.openwall.com/john/>

```
Usage: john [OPTIONS] [PASSWORD-FILES]
--single[=SECTION]      "single crack" mode
--wordlist[=FILE] --stdin wordlist mode, read words from FILE or stdin
                        --pipe like --stdin, but bulk reads, and allows rules
--loopback[=FILE]       like --wordlist, but fetch words from a .pot file
--dupe-suppression      suppress all dupes in wordlist (and force
                        preload)
--prince[=FILE]         PRINCE mode, read words from FILE
--encoding=NAME          input encoding (eg. UTF-8, ISO-8859-1). See also
                        doc/ENCODING and --list=hidden-options.
--rules[=SECTION]       enable word mangling rules for wordlist modes
--incremental[=MODE]    "incremental" mode [using section MODE]
--mask=MASK             mask mode using MASK
--markov[=OPTIONS]      "Markov" mode (see doc/MARKOV)
--external=MODE         external mode or word filter
--stdout[=LENGTH]       just output candidate passwords [cut at LENGTH]
--restore[=NAME]         restore an interrupted session [called NAME]
--session=NAME          give a new session the NAME
--status[=NAME]         print status of a session [called NAME]
--make-charset=FILE     make a charset file. It will be overwritten
--show[=LEFT]           show cracked passwords [if =LEFT, then uncracked]
--test[=TIME]           run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..] [do not] load this (these) user(s) only
--groups=[-]GID[,..]    load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..] load users with[out] this (these) shell(s) only
--salts=[-]COUNT[:MAX] load salts with[out] COUNT [to MAX] hashes
--save-memory=LEVEL     enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL this node's number range out of TOTAL count
--fork=N               fork N processes
--pot=NAME             pot file to use
--list=WHAT            list capabilities, see --list=help or doc/OPTIONS
--format=NAME          force hash of type NAME. The supported formats
                        can
                        be seen with --list=formats and --list=subformats
```

#### 4.1.36 kismet

Kismet is an 802.11 layer2 wireless network detector, sniffer, and intrusion detection system. Kismet will work with any wireless card which supports raw monitoring (rfmon) mode, and can sniff 802.11b, 802.11a, and 802.11g traffic.

```
[root@ArcheryOS ~]# kismet --help
```

```
Usage: /usr/bin/kismet_server [OPTION]
```

Nearly all of these options are run-time overrides for values in the kismet.conf configuration file. Permanent changes should be made to the configuration file.

---

```

*** Generic Options ***
-v, --version          Show version
-f, --config-file <file> Use alternate configuration file
    --no-line-wrap      Turn off linewrapping of output
                        (for grep, speed, etc)
-s, --silent           Turn off stdout output after setup phase
    --daemonize         Spawn detached in the background
    --no-plugins        Do not load plugins
    --no-root           Do not start the kismet_capture binary
                        when not running as root. For no-priv
                        remote capture ONLY.

*** Kismet Client/Server Options ***
-l, --server-listen     Override Kismet server listen options

*** Kismet Remote Drone Options ***
    --drone-listen      Override Kismet drone listen options

*** Dump/Logging Options ***
-T, --log-types <types> Override activated log types
-t, --log-title <title> Override default log title
-p, --log-prefix <prefix> Directory to store log files
-n, --no-logging        Disable logging entirely

*** Packet Capture Source Options ***
-c, --capture-source    Specify a new packet capture source
                        (Identical syntax to the config file)
-C, --enable-capture-sources Enable capture sources (comma-separated
                        list of names or interfaces)

*** Kismet Net Tracking Options ***
    --filter-tracker     Tracker filtering

*** Kismet GPS Options ***
    --use-gpsd-gps (h:p) Use GPSD-controlled GPS at host:port
                        (default: localhost:2947)
    --use-nmea-gps (dev) Use local NMEA serial GPS on device
                        (default: /dev/ttyUSB0)
    --use-virtual-gps
        (lat,lon,alt) Use a virtual fixed-position gps record
    --gps-modelock <t:f> Force broken GPS units to act as if they
                        have a valid signal (true/false)
    --gps-reconnect <t:f> Reconnect if a GPS device fails
                        (true/false)

```

---

### 4.1.37 lynis

Lynis is a security auditing tool for Linux, Mac OSX, and UNIX systems. It checks the system and the software configuration, to see if there is any room for improvement the security defenses. All details are stored in a log file.

---

```
[root@ArcheryOS ~]# lynis --help

[ Lynis 2.5.0 ]

#####
Lynis comes with ABSOLUTELY NO WARRANTY. This is free software, and
you are
welcome to redistribute it under the terms of the GNU General Public
License.
See the LICENSE file for details about using this software.

2007-2017, CISOfy - https://cisofy.com/lynis/
Enterprise support available (compliance, plugins, interface and tools)
#####

[+] Initializing program
-----

Usage: lynis command [options]

Command:

audit
  audit system          : Perform local security scan
  audit system remote <host> : Remote security scan
  audit dockerfile <file>  : Analyze Dockerfile

show
  show                  : Show all commands
  show version          : Show Lynis version
  show help             : Show help

update
  update info          : Show update details
  update release       : Update Lynis release

Options:

--no-log          : Don't create a log file
--pentest        : Non-privileged scan (useful for
```



---

```

    pentest)
--profile <profile>           : Scan the system with the given
    profile file
--quick (-Q)                  : Quick mode, don't wait for user
    input

Layout options
--no-colors                   : Don't use colors in output
--quiet (-q)                  : No output
--reverse-colors              : Optimize color display for light
    backgrounds

Misc options
--debug                       : Debug logging to screen
--view-manpage (--man)        : View man page
--verbose                     : Show more details on screen
--version (-V)                : Display version number and quit

Enterprise options
--plugin-dir "<path>"          : Define path of available plugins
--upload                      : Upload data to central node

More options available. Run '/usr/sbin/lynis show options', or use
the man page.

```

---

#### 4.1.38 macchanger

macchanger is a GNU/Linux utility for viewing/manipulating the MAC address for network interfaces.

---

```

[root@ArcheryOS ~]# macchanger --help
GNU MAC Changer
Usage: macchanger [options] device

-h, --help                Print this help
-V, --version              Print version and exit
-s, --show                 Print the MAC address and exit
-e, --ending               Don't change the vendor bytes
-a, --another              Set random vendor MAC of the same kind
-A                         Set random vendor MAC of any kind
-p, --permanent            Reset to original, permanent hardware MAC
-r, --random               Set fully random MAC
-l, --list[=keyword]       Print known vendors
-b, --bia                  Pretend to be a burned-in-address
-m, --mac=XX:XX:XX:XX:XX:XX
    --mac XX:XX:XX:XX:XX:XX Set the MAC XX:XX:XX:XX:XX:XX

```

Report bugs to <https://github.com/alobbs/macchanger/issues>

---

### 4.1.39 maltego-community

Maltego is used for open-source intelligence and forensics, developed by Paterva. Maltego focuses on providing a library of transforms for discovery of data from open sources, and visualizing that information in a graph format, suitable for link analysis and data mining.

---

```
[root@ArcheryOS ~]# maltego --help
Module reload options:
  --reload /path/to/module.jar install or reinstall a module JAR file

Additional module options:
-a, --alias <arg>
-u, --updates <arg>
-i, --import <arg>
-p, --automationPort <arg>
-m, --machine <arg>
-o, --open <arg1>...<argN>

Core options:
--laf <LaF classname> use given LookAndFeel class instead of the
                        default
--fontsize <size>      set the base font size of the user interface, in
                        points
--locale <language[:country[:variant]]> use specified locale
--userdir <path>       use specified directory to store user settings
--cachedir <path>     use specified directory to store user cache, must
                        be different from userdir
--nosplash             do not show the splash screen
```

---

### 4.1.40 masscan

masscan is an Internet-scale port scanner, useful for large scale surveys of the Internet, or of internal networks.

---

```
[root@ArcheryOS ~]# masscan --help
MASSCAN is a fast port scanner. The primary input parameters are the
IP addresses/ranges you want to scan, and the port numbers. An example
is the following, which scans the 10.x.x.x network for web servers:
masscan 10.0.0.0/8 -p80
The program auto-detects network interface/adaptor settings. If this
fails, you'll have to set these manually. The following is an
example of all the parameters that are needed:
--adapter-ip 192.168.10.123
--adapter-mac 00-11-22-33-44-55
--router-mac 66-55-44-33-22-11
Parameters can be set either via the command-line or config-file. The
names are the same for both. Thus, the above adapter settings would
appear as follows in a configuration file:
```

```

adapter-ip = 192.168.10.123
adapter-mac = 00-11-22-33-44-55
router-mac = 66-55-44-33-22-11

```

All single-dash parameters have a spelled out double-dash equivalent, so '-p80' is the same as '--ports 80' (or 'ports = 80' in config file). To use the config file, `type:`

```
masscan -c <filename>
```

To generate a config-file from the current settings, use the `--echo` option. This stops the program from actually running, and just echoes the current configuration instead. This is a useful way to generate your first config file, or see a list of parameters you didn't know about. I suggest you try it now:

```
masscan -p1234 --echo
```

---

#### 4.1.41 mdk3

MDK is a proof-of-concept tool to exploit common IEEE 802.11 protocol weaknesses.

---

```
[root@ArcheryOS ~]# mdk3 --help
```

```

MDK 3.0 v6 - "Yeah, well, whatever"
by ASPj of k2wrlz, using the osdep library from aircrack-ng
And with lots of help from the great aircrack-ng community:
Antragon, moongray, Ace, Zero_Chaos, Hirte, thefkboss, ducttape,
telekOmiker, Le_Vert, sorbo, Andy Green, bahathir and Dawid Gajownik
THANK YOU!

```

MDK is a proof-of-concept tool to exploit common IEEE 802.11 protocol weaknesses.

IMPORTANT: It is your responsibility to make sure you have permission from the network owner before running MDK against it.

This code is licenced under the GPLv2

MDK USAGE:

```
mdk3 <interface> <test_mode> [test_options]
```

Try `mdk3 --fullhelp` for all test options

Try `mdk3 --help <test_mode>` for info about one test only

TEST MODES:

- b - Beacon Flood Mode
  - Sends beacon frames to show fake APs at clients.
  - This can sometimes crash network scanners and even drivers!
- a - Authentication DoS mode
  - Sends authentication frames to all APs found in range.
  - Too much clients freeze or reset some APs.

- 
- p - Basic probing and ESSID Bruteforce mode  
Probes AP and check **for** answer, useful **for** checking **if** SSID has been correctly de cloaked or **if** AP is **in** your adaptors sending range SSID Bruteforcing is also possible with this **test** mode.
  - d - Deauthentication / Disassociation Amok Mode  
Kicks everybody found from AP
  - m - Michael shutdown exploitation (TKIP)  
Cancels all traffic continuously
  - x - 802.1X tests
  - w - WIDS/WIPS Confusion  
Confuse/Abuse Intrusion Detection and Prevention Systems
  - f - MAC filter bruteforce mode  
This **test** uses a list of known client MAC Addresses and tries to authenticate them to the given AP **while** dynamically changing its response timeout **for** best performance. It currently works only on APs who deny an open authentication request properly
  - g - WPA Downgrade **test**  
deauthenticates Stations and APs sending WPA encrypted packets. With this **test** you can check **if** the sysadmin will try setting his network to WEP or disable encryption.
- 

#### 4.1.42 medusa

Medusa is intended to be a speedy, massively parallel, modular, login brute-forcer. The goal is to support as many services which allow remote authentication as possible. The author considers following items to some of the key features of this application:

---

```
[root@ArcheryOS ~]# medusa --help
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks
<jmk@foofus.net>

medusa: invalid option -- '-'
CRITICAL: Unknown error processing command-line options.
ALERT: User logon information must be supplied.
```

Syntax: Medusa [-h host|-H file] [-u username|-U file] [-p password|-P file] [-C file] -M module [OPT]

- h [TEXT] : Target hostname or IP address
- H [FILE] : File containing target hostnames or IP addresses
- u [TEXT] : Username to **test**
- U [FILE] : File containing usernames to **test**
- p [TEXT] : Password to **test**
- P [FILE] : File containing passwords to **test**
- C [FILE] : File containing combo entries. See README **for** more information.
- O [FILE] : File to append log information to
- e [n/s/ns] : Additional password checks ([n] No Password, [s]

---

```

    Password = Username)
-M [TEXT] : Name of the module to execute (without the .mod
           extension)
-m [TEXT] : Parameter to pass to the module. This can be passed
           multiple times with a
           different parameter each time and they will all be sent
           to the module (i.e.
           -m Param1 -m Param2, etc.)
-d        : Dump all known modules
-n [NUM]  : Use for non-default TCP port number
-s        : Enable SSL
-g [NUM]  : Give up after trying to connect for NUM seconds (default
           3)
-r [NUM]  : Sleep NUM seconds between retry attempts (default 3)
-R [NUM]  : Attempt NUM retries before giving up. The total number
           of attempts will be NUM + 1.
-c [NUM]  : Time to wait in usec to verify socket is available
           (default 500 usec).
-t [NUM]  : Total number of logins to be tested concurrently
-T [NUM]  : Total number of hosts to be tested concurrently
-L        : Parallelize logins using one username per thread. The
           default is to process
           the entire username before proceeding.
-f        : Stop scanning host after first valid username/password
           found.
-F        : Stop audit after first valid username/password found on
           any host.
-b        : Suppress startup banner
-q        : Display module's usage information
-v [NUM]  : Verbose level [0 - 6 (more)]
-w [NUM]  : Error debug level [0 - 10 (more)]
-V        : Display version
-Z [TEXT] : Resume scan based on map of previous scan

```

---

#### 4.1.43 movfuscator

The M/o/Vfuscator (short 'o', sounds like "mobfuscator") compiles programs into "mov" instructions, and only "mov" instructions. Arithmetic, comparisons, jumps, function calls, and everything else a program needs are all performed through mov operations; there is no self-modifying code, no transport-triggered calculation, and no other form of non-mov cheating.

---

```

[root@ArcheryOS ~]# movcc --help
Usage: /usr/bin/ld [options] file...
Options:
  -a KEYWORD          Shared library control for HP/UX compatibility
  -A ARCH, --architecture ARCH
                      Set architecture

```

---

```

-b TARGET, --format TARGET Specify target for following input files
-c FILE, --mri-script FILE Read MRI format linker script
-d, -dc, -dp Force common symbols to be defined
--force-group-allocation Force group members out of groups
-e ADDRESS, --entry ADDRESS Set start address
-E, --export-dynamic Export all dynamic symbols
--no-export-dynamic Undo the effect of --export-dynamic
-EB Link big-endian objects
-EL Link little-endian objects
-f SHLIB, --auxiliary SHLIB Auxiliary filter for shared object symbol
  table
-F SHLIB, --filter SHLIB Filter for shared object symbol table
-g Ignored
-G SIZE, --gpsize SIZE Small data size (if no size, same as --shared)
-h FILENAME, -soname FILENAME
  Set internal name of shared library
-I PROGRAM, --dynamic-linker PROGRAM
  Set PROGRAM as the dynamic linker to use
--no-dynamic-linker Produce an executable with no program
  interpreter header
-l LIBNAME, --library LIBNAME
  Search for library LIBNAME
-L DIRECTORY, --library-path DIRECTORY
  Add DIRECTORY to library search path
--sysroot=<DIRECTORY> Override the default sysroot location
-m EMULATION Set emulation
-M, --print-map Print map file on standard output
-n, --nmagic Do not page align data
-N, --omagic Do not page align data, do not make text
  readonly
--no-omagic Page align data, make text readonly
-o FILE, --output FILE Set output file name
-O Optimize output file
--out-implib FILE Generate import library
-plugin PLUGIN Load named plugin
-plugin-opt ARG Send arg to last-loaded plugin
-flto Ignored for GCC LTO option compatibility
-flto-partition= Ignored for GCC LTO option compatibility
-fuse-ld= Ignored for GCC linker option compatibility
--map-whole-files Ignored for gold option compatibility
--no-map-whole-files Ignored for gold option compatibility
-Qy Ignored for SVR4 compatibility
-q, --emit-relocs Generate relocations in final output
-r, -i, --relocatable Generate relocatable output
-R FILE, --just-symbols FILE
  Just link symbols (if directory, same as
  --rpath)
-s, --strip-all Strip all symbols
-S, --strip-debug Strip debugging symbols
--strip-discarded Strip symbols in discarded sections

```

---

```

--no-strip-discarded    Do not strip symbols in discarded sections
-t, --trace             Trace file opens
-T FILE, --script FILE  Read linker script
--default-script FILE, -dT Read default linker script
-u SYMBOL, --undefined SYMBOL
                        Start with undefined reference to SYMBOL
--require-defined SYMBOL Require SYMBOL be defined in the final output
--unique [=SECTION]     Don't merge input [SECTION | orphan] sections
-Ur                     Build global constructor/destructor tables
-v, --version           Print version information
-V                     Print version and emulation information
-x, --discard-all      Discard all local symbols
-X, --discard-locals    Discard temporary local symbols (default)
--discard-none          Don't discard any local symbols
-y SYMBOL, --trace-symbol SYMBOL
                        Trace mentions of SYMBOL
-Y PATH                Default search path for Solaris compatibility
-(, --start-group       Start a group
-), --end-group         End a group
--accept-unknown-input-arch Accept input files whose architecture
                        cannot be determined
--no-accept-unknown-input-arch
                        Reject input files whose architecture is
                        unknown
--as-needed             Only set DT_NEEDED for following dynamic libs
                        if used
--no-as-needed          Always set DT_NEEDED for dynamic libraries
                        mentioned on
                        the command line
-assert KEYWORD         Ignored for SunOS compatibility
-Bdynamic, -dy, -call_shared
                        Link against shared libraries
-Bstatic, -dn, -non_shared, -static
                        Do not link against shared libraries
-Bsymbolic             Bind global references locally
-Bsymbolic-functions    Bind global function references locally
--check-sections        Check section addresses for overlaps (default)
--no-check-sections     Do not check section addresses for overlaps
--copy-dt-needed-entries Copy DT_NEEDED links mentioned inside DSOs
                        that follow
--no-copy-dt-needed-entries Do not copy DT_NEEDED links mentioned
                        inside DSOs that follow
--cref                 Output cross reference table
--defsym SYMBOL=EXPRESSION Define a symbol
--demangle [=STYLE]     Demangle symbol names [using STYLE]
--embedded-relocs       Generate embedded relocs
--fatal-warnings        Treat warnings as errors
--no-fatal-warnings     Do not treat warnings as errors (default)
-fini SYMBOL            Call SYMBOL at unload-time
--force-exe-suffix      Force generation of file with .exe suffix

```

---

```

--gc-sections          Remove unused sections (on some targets)
--no-gc-sections       Don't remove unused sections (default)
--print-gc-sections    List removed unused sections on stderr
--no-print-gc-sections Do not list removed unused sections
--gc-keep-exported     Keep exported symbols when removing unused
                        sections
--hash-size=<NUMBER>  Set default hash table size close to <NUMBER>
--help                Print option help
--init SYMBOL         Call SYMBOL at load-time
--Map FILE            Write a map file
--no-define-common    Do not define Common storage
--no-demangle         Do not demangle symbol names
--no-keep-memory      Use less memory and more disk I/O
--no-undefined        Do not allow unresolved references in object
                        files
--allow-shlib-undefined Allow unresolved references in shared
                        libraries
--no-allow-shlib-undefined Do not allow unresolved references in
                        shared libs
--allow-multiple-definition Allow multiple definitions
--no-undefined-version Disallow undefined version
--default-symver       Create default symbol version
--default-imported-symver Create default symbol version for imported
                        symbols
--no-warn-mismatch     Don't warn about mismatched input files
--no-warn-search-mismatch Don't warn on finding an incompatible library
--no-whole-archive     Turn off --whole-archive
--noinhibit-exec       Create an output file even if errors occur
--nostdlib            Only use library directories specified on
                        the command line
--oformat TARGET       Specify target of output file
--print-output-format  Print default output format
--print-sysroot        Print current sysroot
-qmagic               Ignored for Linux compatibility
--reduce-memory-overheads Reduce memory overheads, possibly taking
                        much longer
--relax                Reduce code size by using target specific
                        optimizations
--no-relax             Do not use relaxation techniques to reduce
                        code size
--retain-symbols-file FILE Keep only symbols listed in FILE
-rpath PATH           Set runtime shared library search path
-rpath-link PATH       Set link time shared library search path
-shared, -Bshareable   Create a shared library
-pie, --pic-executable Create a position independent executable
--sort-common [=ascending|descending]
                        Sort common symbols by alignment [in
                        specified order]
--sort-section name|alignment
                        Sort sections by name or maximum alignment

```



```

--spare-dynamic-tags COUNT How many tags to reserve in .dynamic section
--split-by-file [=SIZE] Split output sections every SIZE octets
--split-by-reloc [=COUNT] Split output sections every COUNT relocations
--stats Print memory usage statistics
--target-help Display target specific options
--task-link SYMBOL Do task level linking
--traditional-format Use same format as native linker
--section-start SECTION=ADDRESS
                        Set address of named section
-Tbss ADDRESS Set address of .bss section
-Tdata ADDRESS Set address of .data section
-Ttext ADDRESS Set address of .text section
-Ttext-segment ADDRESS Set address of text segment
-Trodata-segment ADDRESS Set address of rodata segment
-Tldata-segment ADDRESS Set address of ldata segment
--unresolved-symbols=<method>
                        How to handle unresolved symbols. <method> is:
                        ignore-all, report-all,
                        ignore-in-object-files,
                        ignore-in-shared-libs
--verbose [=NUMBER] Output lots of information during link
--version-script FILE Read version information script
--version-exports-section SYMBOL
                        Take export symbols list from .exports, using
                        SYMBOL as the version.
--dynamic-list-data Add data symbols to dynamic list
--dynamic-list-cpp-new Use C++ operator new/delete dynamic list
--dynamic-list-cpp-typeinfo Use C++ typeid dynamic list
--dynamic-list FILE Read dynamic list
--warn-common Warn about duplicate common symbols
--warn-constructors Warn if global constructors/destructors are
                        seen
--warn-multiple-gp Warn if the multiple GP values are used
--warn-once Warn only once per undefined symbol
--warn-section-align Warn if start of section changes due to
                        alignment
--warn-shared-textrel Warn if shared object has DT_TEXTREL
--warn-alternate-em Warn if an object has alternate ELF machine
                        code
--warn-unresolved-symbols Report unresolved symbols as warnings
--error-unresolved-symbols Report unresolved symbols as errors
--whole-archive Include all objects from following archives
--wrap SYMBOL Use wrapper functions for SYMBOL
--ignore-unresolved-symbol SYMBOL
                        Unresolved SYMBOL will not cause an error or
                        warning
--push-state Push state of flags governing input file
                        handling
--pop-state Pop state of flags governing input file
                        handling

```

```

--print-memory-usage      Report target memory usage
--orphan-handling =MODE   Control how orphan sections are handled.
@FILE                     Read options from FILE
/usr/bin/ld: supported targets: elf64-x86-64 elf32-i386 elf32-iamcu
    elf32-x86-64 a.out-i386-linux pei-i386 pei-x86-64 elf64-l1om
    elf64-k1om elf64-little elf64-big elf32-little elf32-big pe-x86-64
    pe-bigobj-x86-64 pe-i386 plugin srec symbolsrec verilog tekhex
    binary ihex
/usr/bin/ld: supported emulations: elf_x86_64 elf32_x86_64 elf_i386
    elf_iamcu i386linux elf_l1om elf_k1om i386pep i386pe
/usr/bin/ld: emulation specific options:
ELF emulations:
--ld-generated-unwind-info Generate exception handling info for PLT
--no-ld-generated-unwind-info
    Don't generate exception handling info for PLT
--build-id[=STYLE]        Generate build ID note
--compress-debug-sections=[none|zlib|zlib-gnu|zlib-gabi]
    Compress DWARF debug sections using zlib
    Default: none
-z common-page-size=SIZE  Set common page size to SIZE
-z max-page-size=SIZE     Set maximum page size to SIZE
-z defs                   Report unresolved symbols in object files.
-z muldefs                Allow multiple definitions
-z execstack              Mark executable as requiring executable stack
-z noexecstack            Mark executable as not requiring executable
    stack
-z globalaudit             Mark executable requiring global auditing
--audit=AUDITLIB           Specify a library to use for auditing
-Bgroup                   Selects group name lookup rules for DSO
--disable-new-dtags        Disable new dynamic tags
--enable-new-dtags         Enable new dynamic tags
--eh-frame-hdr             Create .eh_frame_hdr section
--no-eh-frame-hdr          Do not create .eh_frame_hdr section
--exclude-libs=LIBS        Make all symbols in LIBS hidden
--hash-style=STYLE         Set hash style to sysv, gnu or both
-P AUDITLIB, --depaudit=AUDITLIB
    Specify a library to use for auditing dependencies
-z combreloc               Merge dynamic relocs into one section and sort
-z nocombreloc             Don't merge dynamic relocs into one section
-z global                  Make symbols in DSO available for subsequently
    loaded objects
-z initfirst               Mark DSO to be initialized first at runtime
-z interpose               Mark object to interpose all DSOs but
    executable
-z lazy                    Mark object lazy runtime binding (default)
-z loadfltr                Mark object requiring immediate process
-z nocopyreloc             Don't create copy relocs
-z nodefaultlib            Mark object not to use default search paths
-z nodelete                Mark DSO non-deletable at runtime
-z nodlopen                Mark DSO not available to dlopen

```

```

-z nodump           Mark DSO not available to dldump
-z now             Mark object non-lazy runtime binding
-z origin          Mark object requiring immediate $ORIGIN
                   processing at runtime
-z relro           Create RELRO program header (default)
-z norelro         Don't create RELRO program header
-z separate-code   Create separate code program header
-z noseparate-code Don't create separate code program header
                   (default)
-z common          Generate common symbols with STT_COMMON type
-z nocommon        Generate common symbols with STT_OBJECT type
-z stack-size=SIZE Set size of stack segment
-z text            Treat DT_TEXTREL in shared object as error
-z notext          Don't treat DT_TEXTREL in shared object as
                   error
-z textoff         Don't treat DT_TEXTREL in shared object as
                   error
elf_x86_64:
-z noextern-protected-data Do not treat protected data symbol as
                   external
-z dynamic-undefined-weak Make undefined weak symbols dynamic
-z nodynamic-undefined-weak Do not make undefined weak symbols dynamic
-z noreloc-overflow  Disable relocation overflow check
-z call-nop=Padding Use Padding as 1-byte NOP for branch
-z ibtplt           Generate IBT-enabled PLT entries
-z ibt             Generate GNU_PROPERTY_X86_FEATURE_1_IBT
-z shstk           Generate GNU_PROPERTY_X86_FEATURE_1_SHSTK
-z bndplt           Always generate BND prefix in PLT entries
elf32_x86_64:
-z noextern-protected-data Do not treat protected data symbol as
                   external
-z dynamic-undefined-weak Make undefined weak symbols dynamic
-z nodynamic-undefined-weak Do not make undefined weak symbols dynamic
-z noreloc-overflow  Disable relocation overflow check
-z call-nop=Padding Use Padding as 1-byte NOP for branch
-z ibtplt           Generate IBT-enabled PLT entries
-z ibt             Generate GNU_PROPERTY_X86_FEATURE_1_IBT
-z shstk           Generate GNU_PROPERTY_X86_FEATURE_1_SHSTK
elf_i386:
-z noextern-protected-data Do not treat protected data symbol as
                   external
-z dynamic-undefined-weak Make undefined weak symbols dynamic
-z nodynamic-undefined-weak Do not make undefined weak symbols dynamic
-z call-nop=Padding Use Padding as 1-byte NOP for branch
-z ibtplt           Generate IBT-enabled PLT entries
-z ibt             Generate GNU_PROPERTY_X86_FEATURE_1_IBT
-z shstk           Generate GNU_PROPERTY_X86_FEATURE_1_SHSTK
elf_i386:
-z noextern-protected-data Do not treat protected data symbol as
                   external

```

```

-z dynamic-undefined-weak Make undefined weak symbols dynamic
-z nodynamic-undefined-weak Do not make undefined weak symbols dynamic
-z call-nop=Padding      Use Padding as 1-byte NOP for branch
elf_l1om:
-z noextern-protected-data Do not treat protected data symbol as
  external
-z dynamic-undefined-weak Make undefined weak symbols dynamic
-z nodynamic-undefined-weak Do not make undefined weak symbols dynamic
-z call-nop=Padding      Use Padding as 1-byte NOP for branch
elf_k1om:
-z noextern-protected-data Do not treat protected data symbol as
  external
-z dynamic-undefined-weak Make undefined weak symbols dynamic
-z nodynamic-undefined-weak Do not make undefined weak symbols dynamic
-z call-nop=Padding      Use Padding as 1-byte NOP for branch
i386pep:
--base_file <basefile>      Generate a base file for relocatable
  DLLs
--dll                        Set image base to the default for DLLs
--file-alignment <size>     Set file alignment
--heap <size>               Set initial size of the heap
--image-base <address>      Set start address of the executable
--major-image-version <number> Set version number of the executable
--major-os-version <number>  Set minimum required OS version
--major-subsystem-version <number> Set minimum required OS subsystem
  version
--minor-image-version <number> Set revision number of the executable
--minor-os-version <number>   Set minimum required OS revision
--minor-subsystem-version <number> Set minimum required OS subsystem
  revision
--section-alignment <size>   Set section alignment
--stack <size>               Set size of the initial stack
--subsystem <name>[:<version>] Set required OS subsystem [& version]
--support-old-code           Support interworking with old code
--[no-]leading-underscore   Set explicit symbol underscore prefix
  mode
--[no-]insert-timestamp     Use a real timestamp rather than zero.
  (default)
--add-stdcall-alias          This makes binaries non-deterministic
  Export symbols with and without @nn
--disable-stdcall-fixup      Don't link _sym to _sym@nn
--enable-stdcall-fixup       Link _sym to _sym@nn without warnings
--exclude-symbols sym,sym,... Exclude symbols from automatic export
--exclude-all-symbols        Exclude all symbols from automatic
  export
--exclude-libs lib,lib,...    Exclude libraries from automatic export
--exclude-modules-for-implib mod,mod,...
  Exclude objects, archive members from
  auto
  export, place into import library

```

	instead.
--export-all-symbols	Automatically <b>export</b> all globals to DLL
--kill-at	Remove @nn from exported symbols
--output-def <file>	Generate a .DEF file <b>for</b> the built DLL
--warn-duplicate-exports	Warn about duplicate exports.
--compat-implib	Create backward compatible import libs; create __imp_<SYMBOL> as well.
--enable-auto-image-base DLLs	Automatically choose image base <b>for</b> unless user specifies one
--disable-auto-image-base (default)	Do not auto-choose image base.
--dll-search-prefix=<string> without	When linking dynamically to a dll an importlib, use <string><basename>.dll <b>in</b> preference to lib<basename>.dll
--enable-auto-import	Do sophisticated linking of _sym to __imp_sym <b>for</b> DATA references
--disable-auto-import	Do not auto-import DATA items from DLLs
--enable-runtime-pseudo-reloc	Work around auto-import limitations by adding pseudo-relocations resolved at runtime.
--disable-runtime-pseudo-reloc <b>for</b>	Do not add runtime pseudo-relocations auto-imported DATA.
--enable-extra-pep-debug building	Enable verbose debug output when or linking to DLLs (esp. auto-import)
--enable-long-section-names	Use long COFF section names even <b>in</b> executable image files
--disable-long-section-names	Never use long COFF section names, even <b>in</b> object files
--high-entropy-va address space	Image is compatible with 64-bit layout randomization (ASLR)
--dynamicbase	Image base address may be relocated using address space layout randomization (ASLR)
--forceinteg	Code integrity checks are enforced
--nxcompat	Image is compatible with data execution prevention
--no-isolation image	Image understands isolation but <b>do</b> not isolate the
--no-seh	Image does not use SEH. No SE handler may be called <b>in</b> this image
--no-bind	Do not <b>bind</b> this image
--wdmdriver	Driver uses the WDM model
--tsaware	Image is Terminal Server aware
--build-id[=STYLE]	Generate build ID
i386pe:	
--base_file <basefile>	Generate a base file <b>for</b> relocatable

```

DLLs
--dll                               Set image base to the default for DLLs
--file-alignment <size>            Set file alignment
--heap <size>                      Set initial size of the heap
--image-base <address>             Set start address of the executable
--major-image-version <number>     Set version number of the executable
--major-os-version <number>        Set minimum required OS version
--major-subsystem-version <number> Set minimum required OS subsystem
                                   version
--minor-image-version <number>     Set revision number of the executable
--minor-os-version <number>        Set minimum required OS revision
--minor-subsystem-version <number> Set minimum required OS subsystem
                                   revision
--section-alignment <size>         Set section alignment
--stack <size>                    Set size of the initial stack
--subsystem <name>[:<version>]     Set required OS subsystem [& version]
--support-old-code                 Support interworking with old code
--[no-]leading-underscore          Set explicit symbol underscore prefix
                                   mode
--thumb-entry=<symbol>              Set the entry point to be Thumb
                                   <symbol>
--[no-]insert-timestamp            Use a real timestamp rather than zero
                                   (default).

                                   This makes binaries non-deterministic
--add-stdcall-alias                Export symbols with and without @nn
--disable-stdcall-fixup            Don't link _sym to _sym@nn
--enable-stdcall-fixup             Link _sym to _sym@nn without warnings
--exclude-symbols sym,sym,...      Exclude symbols from automatic export
--exclude-all-symbols             Exclude all symbols from automatic
                                   export
--exclude-libs lib,lib,...         Exclude libraries from automatic export
--exclude-modules-for-implib mod,mod,...
                                   Exclude objects, archive members from
                                   auto
                                   export, place into import library
                                   instead.
--export-all-symbols              Automatically export all globals to DLL
--kill-at                          Remove @nn from exported symbols
--output-def <file>               Generate a .DEF file for the built DLL
--warn-duplicate-exports           Warn about duplicate exports
--compat-implib                   Create backward compatible import libs;
                                   create __imp_<SYMBOL> as well.
--enable-auto-image-base[=<address>] Automatically choose image base
                                   for DLLs
                                   (optionally starting with address)
                                   unless
                                   specifically set with --image-base
--disable-auto-image-base          Do not auto-choose image base.
                                   (default)
--dll-search-prefix=<string>       When linking dynamically to a dll

```

Report bugs to <<https://bugs.archlinux.org/>>

msfconsole is a way to interact with the Metasploit Framework (MSF).

[illegible]

```

|_|  | | | _|_  | | / -\ _\ \ | |  | | \_/_| | | | _
    | / |___/ \___\ / \ \ \___/ \ /   \_|  | \ \___\

      =[ metasploit v4.16.58-dev                      ]
+ -- --=[ 1771 exploits - 1006 auxiliary - 307 post ]
+ -- --=[ 537 payloads - 41 encoders - 10 nops      ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >

```

---

#### 4.1.45 msfd

msfd provides an instance of msfconsole that remote clients can connect to.

```
[root@ArcheryOS ~]# msfd -h
```

Usage: msfd <options>

OPTIONS:

```

-A <opt> Specify list of hosts allowed to connect
-D <opt> Specify list of hosts not allowed to connect
-a <opt> Bind to this IP address instead of loopback
-f      Run the daemon in the foreground
-h      Help banner
-p <opt> Bind to this port instead of 55554
-q      Do not print the banner on startup
-s      Use SSL

```

---

#### 4.1.46 msfpc

MSFvenom Payload Creator (MSFPC) is a wrapper that generates multiple types of payloads, based on user-selected options.

```
[root@ArcheryOS ~]# msfpc --help
```

[\*] MSFvenom Payload Creator (MSFPC v1.4.4)

```

/usr/bin/msfpc <TYPE> (<DOMAIN/IP>) (<PORT>) (<CMD/MSF>)
                (<BIND/REVERSE>) (<STAGED/STAGELESS>) (<TCP/HTTP/HTTPS/FIND_PORT>)
                (<BATCH/LOOP>) (<VERBOSE>)

```

```

Example: /usr/bin/msfpc windows 192.168.1.10 # Windows & manual IP.
         /usr/bin/msfpc elf bind eth0 4444   # Linux, eth0's IP &
         manual port.
         /usr/bin/msfpc stageless cmd py https # Python, stageless
         command prompt.
         /usr/bin/msfpc verbose loop eth1     # A payload for every
         type, using eth1's IP.

```



```

/usr/bin/msfpc msf batch wan      # All possible
    Meterpreter payloads, using WAN IP.
/usr/bin/msfpc help verbose      # Help screen, with
    even more information.

```

```

<TYPE>:
+ APK
+ ASP
+ ASPX
+ Bash [.sh]
+ Java [.jsp]
+ Linux [.elf]
+ OSX [.macho]
+ Perl [.pl]
+ PHP
+ Powershell [.ps1]
+ Python [.py]
+ Tomcat [.war]
+ Windows [.exe // .exe // .dll]

```

Rather than putting <DOMAIN/IP>, you can **do** a interface and MSFPC will detect that IP address.

Missing <DOMAIN/IP> will default to the IP menu.

Missing <PORT> will default to 443.

<CMD> is a standard/native **command** prompt/terminal to interactive with.  
 <MSF> is a custom cross platform shell, gaining the full power of Metasploit.

Missing <CMD/MSF> will default to <MSF> where possible.

<BIND> opens a port on the target side, and the attacker connects to them. Commonly blocked with ingress firewalls rules on the target.  
 <REVERSE> makes the target connect back to the attacker. The attacker needs an open port. Blocked with engress firewalls rules on the target.

Missing <BIND/REVERSE> will default to <REVERSE>.

<STAGED> splits the payload into parts, making it smaller but dependent on Metasploit.

<STAGELESS> is the **complete** standalone payload. More 'stable' than <STAGED>.

Missing <STAGED/STAGELESS> will default to <STAGED> where possible.

<TCP> is the standard method to connecting back. This is the most compatible with TYPES as its RAW. Can be easily detected on IDSs.

<HTTP> makes the communication appear to be HTTP traffic (unencrypted). Helpful **for** packet inspection, which limit port access on protocol - e.g. TCP 80.

<HTTPS> makes the communication appear to be (encrypted) HTTP traffic

---

using as SSL. Helpful `for` packet inspection, which limit port access on protocol - e.g. TCP 443.

<FIND\_PORT> will attempt every port on the target machine, to find a way out. Useful with stick ingress/egress firewall rules. Will switch to 'allports' based on <TYPE>.

Missing <TCP/HTTP/HTTPS/FIND\_PORT> will default to <TCP>.

<BATCH> will generate as many combinations as possible: <TYPE>, <CMD + MSF>, <BIND + REVERSE>, <STAGED + STAGLESS> & <TCP + HTTP + HTTPS + FIND\_PORT>

<LOOP> will just create one of each <TYPE>.

<VERBOSE> will display more information.

---

#### 4.1.47 msfrpc

msfrpc connects to an RPC instance of Metasploit.

---

```
[root@ArcheryOS ~]# msfrpc -h
```

Usage: msfrpc <options>

OPTIONS:

```
-P <opt> Specify the password to access msfrpcd
-S      Disable SSL on the RPC socket
-U <opt> Specify the username to access msfrpcd
-a <opt> Connect to this IP address
-h      Help banner
-p <opt> Connect to the specified port instead of 55553
```

---

#### 4.1.48 msfrpcd

msfrpcd provides an RPC interface to Metasploit.

---

```
[root@ArcheryOS ~]# msfrpcd -h
```

Usage: msfrpcd <options>

OPTIONS:

```
-P <opt> Specify the password to access msfrpcd
-S      Disable SSL on the RPC socket
-U <opt> Specify the username to access msfrpcd
-a <opt> Bind to this IP address
-f      Run the daemon in the foreground
-h      Help banner
-n      Disable database
```

---

```
-p <opt> Bind to this port instead of 55553
-t <opt> Token Timeout (default 300 seconds)
-u <opt> URI for Web server
```

---

#### 4.1.49 msfvenom

msfvenom is a standalone Metasploit payload generator

---

```
[root@ArcheryOS ~]# msfvenom -h
MsfVenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>
```

##### Options:

-p, --payload	<payload>	Payload to use. Specify a '-' or stdin to use custom payloads
	--payload-options	List the payload's standard options
-l, --list	[type]	List a module type. Options are: payloads, encoders, nops, all
-n, --nopsled	<length>	Prepend a nopsled of [length] size on to the payload
-f, --format	<format>	Output format (use --help-formats for a list)
	--help-formats	List available formats
-e, --encoder	<encoder>	The encoder to use
-a, --arch	<arch>	The architecture to use
	--platform	<platform> The platform of the payload
	--help-platforms	List available platforms
-s, --space	<length>	The maximum size of the resulting payload
	--encoder-space	<length> The maximum size of the encoded payload (defaults to the -s value)
-b, --bad-chars	<list>	The list of characters to avoid example: '\x00\xff'
-i, --iterations	<count>	The number of times to encode the payload
-c, --add-code	<path>	Specify an additional win32 shellcode file to include
-x, --template	<path>	Specify a custom executable file to use as a template
-k, --keep		Preserve the template behavior and inject the payload as a new thread
-o, --out	<path>	Save the payload
-v, --var-name	<name>	Specify a custom variable name to use for certain output formats
	--smallest	Generate the smallest possible payload
-h, --help		Show this message

---

#### 4.1.50 mimikatz

mimikatz is a group of tools to exploit Windows security.

---

#### 4.1.51 miranda

miranda is a Python-based Universal Plug-N-Play client application designed to discover, query and interact with UPNP devices, particularly Internet Gateway Devices (aka, routers). It can be used to audit UPNP-enabled devices on a network for possible vulnerabilities. Some of its features include:

---

```
[root@ArcheryOS ~]# miranda -h
```

Command line usage: /usr/bin/miranda [OPTIONS]

```
-s <struct file> Load previous host data from struct file
-l <log file>    Log user-supplied commands to log file
-i <interface>  Specify the name of the interface to use (Linux only,
                requires root)
-u             Disable show-uniq-hosts-only option
-d             Enable debug mode
-v             Enable verbose mode
-h             Show help
```

---

#### 4.1.52 mitmf

MITMf aims to provide a one-stop-shop for Man-In-The-Middle and network attacks while updating and improving existing attacks and techniques.

Originally built to address the significant shortcomings of other tools (e.g Ettercap, Mallory), it's been almost completely re-written from scratch to provide a modular and easily extendible framework that anyone can use to implement their own MITM attack.

---

```
[root@ArcheryOS ~]# mitmf --help
```

```
usage: mitmf.py -i interface [mitmf options] [plugin name] [plugin
options]
```

MITMf v0.9.8 - 'The Dark Side'

optional arguments:

```
-h, --help            show this help message and exit
-v, --version          show program's version number and exit
```

MITMf:

Options for MITMf

```
--log-level {debug,info}
                        Specify a log level [default: info]
-i INTERFACE           Interface to listen on
-c CONFIG_FILE         Specify config file to use
-p, --preserve-cache  Don't kill client/server caching
-r READ_PCAP, --read-pcap READ_PCAP
                        Parse specified pcap for credentials and exit
-l PORT               Port to listen on (default 10000)
-f, --favicon          Substitute a lock favicon on secure requests.
-k, --killsessions    Kill sessions in progress.
-F FILTER [FILTER ...], --filter FILTER [FILTER ...]
                        Filter to apply to incoming traffic
```

Inject:

Inject arbitrary content into HTML content

```
--inject              Load plugin 'Inject'
--js-url JS_URL       URL of the JS to inject
--js-payload JS_PAYLOAD
                        JS string to inject
--js-file JS_FILE     File containing JS to inject
--html-url HTML_URL   URL of the HTML to inject
--html-payload HTML_PAYLOAD
                        HTML string to inject
--html-file HTML_FILE
                        File containing HTML to inject
--per-domain          Inject once per domain per client.
--rate-limit RATE_LIMIT
                        Inject once every RATE_LIMIT seconds per client.
--count-limit COUNT_LIMIT
                        Inject only COUNT_LIMIT times per client.
--white-ips IP        Inject content ONLY for these ips (comma seperated)
--black-ips IP        DO NOT inject content for these ips (comma
seperated)
--white-domains DOMAINS
                        Inject content ONLY for these domains (comma
```

```

                                seperated)
--black-domains DOMAINS
                                DO NOT inject content for these domains (comma
                                seperated)

HTA Drive-By:
Performs HTA drive-by attacks on clients

--hta                Load plugin 'HTA Drive-By'
--text TEXT          Text to display on notification bar
--hta-app HTA_APP    Path to HTA application [defaults to
                    config/hta_driveby/flash_setup.hta]

Replace:
Replace arbitrary content in HTML content

--replace            Load plugin 'Replace'

Upsidedowninternet:
Flips images 180 degrees

--upsidedowninternet Load plugin 'Upsidedowninternet'

BrowserProfiler:
Attempts to enumerate all browser plugins of connected clients

--browserprofiler    Load plugin 'BrowserProfiler'

FilePwn:
Backdoor executables being sent over http using bdfactory

--filepwn            Load plugin 'FilePwn'

SSLstrip+:
Enables SSLstrip+ for partial HSTS bypass

--hsts                Load plugin 'SSLstrip+'

Captive Portal:
Be a captive portal!

--captive            Load plugin 'Captive Portal'
--portalurl URL      Specify the URL where the portal is located, e.g.
                    http://example.com.
--portaldir LOCALDIR Specify a local path containg the portal files
                    served with a SimpleHTTPServer on a different port (see config).
--use-dns            Whether we use dns spoofing to serve from a
                    fancier portal URL captive.portal when used without options or
                    portaldir. Requires DNS for "captive.portal" to resolve, e.g. via
                    configured dns spoofing --dns.

```

**ScreenShotter:**

Uses HTML5 Canvas to render an accurate screenshot of a clients browser

```
--screen          Load plugin 'ScreenShotter'
--interval SECONDS Interval at which screenshots will be taken
                    (default 10 seconds)
```

**SMBAuth:**

Evoke SMB challenge-response auth attempts

```
--smbauth          Load plugin 'SMBAuth'
```

**Ferret-NG:**

Captures cookies and starts a proxy that will feed them to connected clients

```
--ferretng          Load plugin 'Ferret-NG'
--port PORT          Port to start Ferret-NG proxy on (default 10010)
--load-cookies FILE Load cookies from a log file
```

**JSKeylogger:**

Injects a javascript keylogger into clients webpages

```
--jskeylogger       Load plugin 'JSKeylogger'
```

**AppCachePoison:**

Performs App Cache Poisoning attacks

```
--appoison          Load plugin 'AppCachePoison'
```

**Responder:**

Poison LLMNR, NBT-NS and MDNS requests

```
--responder          Load plugin 'Responder'
--analyze             Allows you to see NBT-NS, BROWSER, LLMNR requests
                    without poisoning
--wreaddir            Enables answers for netbios wreaddir suffix queries
--nbtns              Enables answers for netbios domain suffix queries
--fingerprint         Fingerprint hosts that issued an NBT-NS or LLMNR
                    query
--lm                  Force LM hashing downgrade for Windows XP/2003 and
                    earlier
--wpad                Start the WPAD rogue proxy server
--forcewpaauth        Force NTLM/Basic authentication on wpad.dat file
                    retrieval (might cause a login prompt)
--basic               Return a Basic HTTP authentication. If not set, an
                    NTLM authentication will be returned
```

**ImageRandomizer:**

Replaces images with a random one from a specified directory

```
--imgrand          Load plugin 'ImageRandomizer'
--img-dir DIRECTORY Directory with images
```

Spoof:

Redirect/Modify traffic using ICMP, ARP, DHCP or DNS

```
--spoof           Load plugin 'Spoof'
--arp             Redirect traffic using ARP spoofing
--icmp            Redirect traffic using ICMP redirects
--dhcp            Redirect traffic using DHCP offers
--dns             Proxy/Modify DNS queries
--netmask NETMASK The netmask of the network
--shellshock PAYLOAD Trigger the Shellshock vuln when spoofing DHCP,
                  and execute specified command
--gateway GATEWAY Specify the gateway IP
--gatewaymac GATEWAYMAC
                  Specify the gateway MAC [will auto resolve if
                  omitted]
--targets TARGETS Specify host/s to poison [if omitted will default
                  to subnet]
--ignore IGNORE   Specify host/s not to poison
--arpmode {rep,req} ARP Spoofing mode: replies (rep) or requests (req)
                  [default: rep]
```

SMBTrap:

Exploits the SMBTrap vulnerability on connected clients

```
--smbtrap          Load plugin 'SMBTrap'
```

BrowserSniper:

Performs drive-by attacks on clients with out-of-date browser plugins

```
--browsersniper    Load plugin 'BrowserSniper'
```

Use wisely, young Padawan.

---

#### 4.1.53 nbtscan

NBTScan is a program for scanning IP networks for NetBIOS name information (similar to what the Windows nbtstat tool provides against single hosts). It sends a NetBIOS status query to each address in a supplied range and lists received information in human readable form. For each responded host it lists IP address, NetBIOS computer name, logged-in user name and MAC address.

---

```
[root@ArcheryOS ~]# nbtscan
```



NBTscan version 1.5.1. Copyright (C) 1999-2003 Alla Bezroutchko.  
 This is a free software and it comes with absolutely no warranty.  
 You can use, distribute and modify it under terms of GNU GPL.

Usage:

```
nbtscan [-v] [-d] [-e] [-l] [-t timeout] [-b bandwidth] [-r] [-q] [-s
separator] [-m retransmits] (-f filename)|(<scan_range>)
-v    verbose output. Print all names received
      from each host
-d    dump packets. Print whole packet contents.
-e    Format output in /etc/hosts format.
-l    Format output in lmhosts format.
      Cannot be used with -v, -s or -h options.
-t    timeout wait timeout milliseconds for response.
      Default 1000.
-b    bandwidth Output throttling. Slow down output
      so that it uses no more that bandwidth bps.
      Useful on slow links, so that outgoing queries
      don't get dropped.
-r    use local port 137 for scans. Win95 boxes
      respond to this only.
      You need to be root to use this option on Unix.
-q    Suppress banners and error messages,
-s    separator Script-friendly output. Don't print
      column and record headers, separate fields with separator.
-h    Print human-readable names for services.
      Can only be used with -v option.
-m    retransmits Number of retransmits. Default 0.
-f    filename Take IP addresses to scan from file filename.
      -f - makes nbtscan take IP addresses from stdin.
<scan_range> what to scan. Can either be single IP
      like 192.168.1.1 or
      range of addresses in one of two forms:
      xxx.xxx.xxx.xxx/xx or xxx.xxx.xxx.xxx-xxx.
```

Examples:

```
nbtscan -r 192.168.1.0/24
  Scans the whole C-class network.
nbtscan 192.168.1.25-137
  Scans a range from 192.168.1.25 to 192.168.1.137
nbtscan -v -s : 192.168.1.0/24
  Scans C-class network. Prints results in script-friendly
  format using colon as field separator.
  Produces output like that:
  192.168.0.1:NT_SERVER:00U
  192.168.0.1:MY_DOMAIN:00G
  192.168.0.1:ADMINISTRATOR:03U
  192.168.0.2:OTHER_BOX:00U
  ...
nbtscan -f iplist
  Scans IP addresses specified in file iplist.
```

#### 4.1.54 ncrack

Ncrack is an open source tool for network authentication cracking. It was designed for high-speed parallel cracking using a dynamic engine that can adapt to different network situations. Ncrack can also be extensively fine-tuned for special cases, though the default parameters are generic enough to cover almost every situation. It is built on a modular architecture that allows for easy extension to support additional protocols. Ncrack is designed for companies and security professionals to audit large networks for default or weak passwords in a rapid and reliable way. It can also be used to conduct fairly sophisticated and intensive brute force attacks against individual services.

---

```
[root@ArcheryOS ~]# ncrack --help
Ncrack 0.6 ( http://ncrack.org )
Usage: ncrack [Options] {target and service specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iX <inputfilename>: Input from Nmap's -oX XML output format
  -iN <inputfilename>: Input from Nmap's -oN Normal output format
  -iL <inputfilename>: Input from list of hosts/networks
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
SERVICE SPECIFICATION:
  Can pass target specific services in <service>://target (standard)
  notation or
  using -p which will be applied to all hosts in non-standard notation.
  Service arguments can be specified to be host-specific, type of
  service-specific
  (-m) or global (-g). Ex: ssh://10.0.0.10,at=10,cl=30 -m ssh:at=50 -g
  cd=3000
  Ex2: ncrack -p ssh,ftp:3500,25 10.0.0.10 scanme.nmap.org
  google.com:80,ssl
  -p <service-list>: services will be applied to all non-standard
  notation hosts
  -m <service>:<options>: options will be applied to all services of
  this type
  -g <options>: options will be applied to every service globally
Misc options:
  ssl: enable SSL over this service
  path <name>: used in modules like HTTP ('=' needs escaping if used)
  db <name>: used in modules like MongoDB to specify the database
  domain <name>: used in modules like WinRM to specify the domain
TIMING AND PERFORMANCE:
  Options which take <time> are in seconds, unless you append 'ms'
  (milliseconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
```

Service-specific options:

- cl (min connection limit): minimum number of concurrent parallel connections
- CL (max connection limit): maximum number of concurrent parallel connections
- at (authentication tries): authentication attempts per connection
- cd (connection delay): delay <time> between each connection initiation
- cr (connection retries): caps number of service connection attempts to (time-out): maximum cracking <time> for service, regardless of success so far
- T<0-5>: Set timing template (higher is faster)
- connection-limit <number>: threshold for total concurrent connections
- stealthy-linear: try credentials using only one connection against each specified host
- until you hit the same host again. Overrides all other timing options.

AUTHENTICATION:

- U <filename>: username file
- P <filename>: password file
- user <username\_list>: comma-separated username list
- pass <password\_list>: comma-separated password list
- passwords-first: Iterate password list for each username. Default is opposite.
- pairwise: Choose usernames and passwords in pairs.

OUTPUT:

- oN/-oX <file>: Output scan in normal and XML format, respectively, to the given filename.
- oA <basename>: Output in the two major formats at once
- v: Increase verbosity level (use twice or more for greater effect)
- d[level]: Set or increase debugging level (Up to 10 is meaningful)
- nsock-trace <level>: Set nsock trace level (Valid range: 0 - 10)
- log-errors: Log errors/warnings to the normal-format output file
- append-output: Append to rather than clobber specified output files

MISC:

- resume <file>: Continue previously saved session
- save <file>: Save restoration file with specific filename
- f: quit cracking service after one found credential
- 6: Enable IPv6 cracking
- sL or --list: only list hosts and services
- datadir <dirname>: Specify custom Ncrack data file location
- proxy <type>://proxy:port>: Make connections via socks4, 4a, http.
- V: Print version number
- h: Print this help summary page.

MODULES:

SSH, RDP, FTP, Telnet, HTTP(S), POP3(S), IMAP, SMB, VNC, SIP, Redis, PostgreSQL, MySQL, MSSQL, MongoDB, Cassandra, WinRM, OWA

EXAMPLES:

```
ncrack -v --user root localhost:22
ncrack -v -T5 https://192.168.0.1
```

---

```
ncrack -v -iX ~/nmap.xml -g CL=5,to=1h
SEE THE MAN PAGE (http://nmap.org/ncrack/man.html) FOR MORE OPTIONS AND
EXAMPLES
```

---

#### 4.1.55 netdiscover

netdiscover is an active/passive arp reconnaissance tool, initially developed to gain information about wireless networks without dhcp servers in wardriving scenarios. It can also be used on switched networks.

---

```
[root@ArcheryOS ~]# netdiscover -h
Netdiscover 0.3-pre-beta7 [Active/passive arp reconnaissance tool]
Written by: Jaime Penalba <jpenalbae@gmail.com>
```

Usage: netdiscover [-i device] [-r range | -l file | -p] [-m file] [-s time] [-n node] [-c count] [-f] [-d] [-S] [-P] [-c]

- i device: your network device
- r range: scan a given range instead of auto scan.  
192.168.6.0/24,/16,/8
- l file: scan the list of ranges contained into the given file
- p passive mode: do not send anything, only sniff
- m file: scan the list of known MACs and host names
- F filter: Customize pcap filter expression (default: "arp")
- s time: time to sleep between each arp request (milliseconds)
- n node: last ip octet used for scanning (from 2 to 253)
- c count: number of times to send each arp request (for nets with packet loss)
- f enable fastmode scan, saves a lot of time, recommended for auto
- d ignore home config files for autoscan and fast mode
- S enable sleep time suppression between each request (hardcore mode)
- P print results in a format suitable for parsing by another program
- N Do not print header. Only valid when -P is enabled.
- L in parsable output mode (-P), continue listening after the active scan is completed

If -r, -l or -p are not enabled, netdiscover will scan for common lan addresses.

---

#### 4.1.56 nikto

nikto examines a web server to find potential problems and security vulnerabilities, including:

- Server and software misconfigurations
- Default files and programs
- Insecure files and programs

- Outdated servers and programs

---

```
[root@ArcheryOS ~]# nikto -Help
```

```
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
  -Cgидirs+            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
  -Format+            Save file (-o) format:
                        csv   Comma-separated-value
                        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               Extended help information
  -host+              Target host
```

```

-404code      Ignore these HTTP codes as negative responses
               (always). Format is "302,301".
-404string    Ignore this string in response body content as
               negative response (always). Can be a regular expression.
-id+         Host authentication to use, format is id:pass or
               id:pass:realm
-key+        Client certificate key file
-list-plugins List all available plugins, perform no testing
-maxtime+    Maximum testing time per host (e.g., 1h, 60m,
               3600s)
-mutate+     Guess additional file names:
               1   Test all files with all root
                   directories
               2   Guess for password file names
               3   Enumerate user names via Apache
                   (/~user type requests)
               4   Enumerate user names via cgiwrap
                   (/cgi-bin/cgiwrap/~user type requests)
               5   Attempt to brute force sub-domain
                   names, assume that the host name is the
                   parent domain
               6   Attempt to guess directory names from
                   the supplied dictionary file
-mutate-options Provide information for mutates
-nointeractive Disables interactive features
-nolookup      Disables DNS lookups
-nossl        Disables the use of SSL
-no404         Disables nikto attempting to guess a 404 page
-Option       Over-ride an option in nikto.conf, can be issued
               multiple times
-output+      Write output to this file ('. ' for auto-name)
-Pause+       Pause between tests (seconds, integer or float)
-Plugins+     List of plugins to run (default: ALL)
-port+        Port to use (default 80)
-RSAcert+     Client certificate file
-root+        Prepend root value to all requests, format is
               /directory
-Save         Save positive responses to this directory ('. '
               for auto-name)
-ssl          Force ssl mode on port
-Tuning+      Scan tuning:
               1   Interesting File / Seen in logs
               2   Misconfiguration / Default File
               3   Information Disclosure
               4   Injection (XSS/Script/HTML)
               5   Remote File Retrieval - Inside Web Root
               6   Denial of Service
               7   Remote File Retrieval - Server Wide
               8   Command Execution / Remote Shell
               9   SQL Injection

```

---

	0	File Upload
	a	Authentication Bypass
	b	Software Identification
	c	Remote Source Inclusion
	d	WebService
	e	Administrative Console
	x	Reverse Tuning Options (i.e., include all except specified)
-timeout+		Timeout <b>for</b> requests (default 10 seconds)
-Userdbs		Load only user databases, not the standard databases
	all	Disable standard dbs and load only user dbs
	tests	Disable only db_tests and load udb_tests
-useragent		Over-rides the default useragent
-until		Run <b>until</b> the specified time or duration
-update		Update databases and plugins from CIRT.net
-useproxy		Use the proxy defined <b>in</b> nikto.conf, or argument http://server:port
-Version		Print plugin and database versions
-vhost+		Virtual host ( <b>for</b> Host header)
		+ requires a value

---

#### 4.1.57 nmap

Nmap (Network Mapper) is an open source tool for network exploration and security auditing. It was designed to rapidly scan large networks, although it works fine against single hosts. Nmap uses raw IP packets in novel ways to determine what hosts are available on the network, what services (application name and version) those hosts are offering, what operating systems (and OS versions) they are running, what type of packet filters/firewalls are in use, and dozens of other characteristics.

---

```
[root@ArcheryOS ~]# nmap --help
Nmap 7.70 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
```

```

-PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given
    ports
-PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
-PO[protocol list]: IP Protocol Ping
-n/-R: Never do DNS resolution/Always resolve [default: sometimes]
--dns-servers <serv1[,serv2],...>: Specify custom DNS servers
--system-dns: Use OS's DNS resolver
--traceroute: Trace hop path to each host
SCAN TECHNIQUES:
-sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
-sU: UDP Scan
-sN/sF/sX: TCP Null, FIN, and Xmas scans
--scanflags <flags>: Customize TCP scan flags
-sI <zombie host[:probeport]>: Idle scan
-sY/sZ: SCTP INIT/COOKIE-ECHO scans
-sO: IP protocol scan
-b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
-p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
--exclude-ports <port ranges>: Exclude the specified ports from
    scanning
-F: Fast mode - Scan fewer ports than the default scan
-r: Scan ports consecutively - don't randomize
--top-ports <number>: Scan <number> most common ports
--port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
-sV: Probe open ports to determine service/version info
--version-intensity <level>: Set from 0 (light) to 9 (try all probes)
--version-light: Limit to most likely probes (intensity 2)
--version-all: Try every single probe (intensity 9)
--version-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
-sC: equivalent to --script=default
--script=<Lua scripts>: <Lua scripts> is a comma separated list of
    directories, script-files or script-categories
--script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts
--script-args-file=filename: provide NSE script args in a file
--script-trace: Show all data sent and received
--script-updatedb: Update the script database.
--script-help=<Lua scripts>: Show help about scripts.
    <Lua scripts> is a comma-separated list of script-files or
    script-categories.
OS DETECTION:
-O: Enable OS detection
--osscan-limit: Limit OS detection to promising targets
--osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
Options which take <time> are in seconds, or append 'ms'
    (milliseconds),

```



's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).  
 -T<0-5>: Set timing template (higher is faster)  
 --min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes  
 --min-parallelism/max-parallelism <numprobes>: Probe parallelization  
 --min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <time>: Specifies probe round trip time.  
 --max-retries <tries>: Caps number of port scan probe retransmissions.  
 --host-timeout <time>: Give up on target after this long  
 --scan-delay/--max-scan-delay <time>: Adjust delay between probes  
 --min-rate <number>: Send packets no slower than <number> per second  
 --max-rate <number>: Send packets no faster than <number> per second

FIREWALL/IDS EVASION AND SPOOFING:

-f; --mtu <val>: fragment packets (optionally w/given MTU)  
 -D <decoy1,decoy2[,ME],...>: Cloak a scan with decoys  
 -S <IP\_Address>: Spoof [source](#) address  
 -e <iface>: Use specified interface  
 -g/--[source](#)-port <portnum>: Use given port number  
 --proxies <url1,[url2],...>: Relay connections through HTTP/SOCKS4 proxies  
 --data <hex string>: Append a custom payload to sent packets  
 --data-string <string>: Append a custom ASCII string to sent packets  
 --data-length <num>: Append random data to sent packets  
 --ip-options <options>: Send packets with specified ip options  
 --ttl <val>: Set IP time-to-live field  
 --spoof-mac <mac address/prefix/vendor name>: Spoof your MAC address  
 --badsum: Send packets with a bogus TCP/UDP/SCTP checksum

OUTPUT:

-oN/-oX/-oS/-oG <file>: Output scan [in](#) normal, XML, s|<riPt kIdDi3, and Grepable format, respectively, to the given filename.  
 -oA <basename>: Output [in](#) the three major formats at once  
 -v: Increase verbosity level (use -vv or more [for](#) greater effect)  
 -d: Increase debugging level (use -dd or more [for](#) greater effect)  
 --reason: Display the reason a port is [in](#) a particular state  
 --open: Only show open (or possibly open) ports  
 --packet-trace: Show all packets sent and received  
 --iflist: Print host interfaces and routes ([for](#) debugging)  
 --append-output: Append to rather than clobber specified output files  
 --resume <filename>: Resume an aborted scan  
 --stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML  
 --webxml: Reference stylesheet from Nmap.Org [for](#) more portable XML  
 --no-stylesheet: Prevent associating of XSL stylesheet w/XML output

MISC:

-6: Enable IPv6 scanning  
 -A: Enable OS detection, version detection, script scanning, and traceroute  
 --datadir <dirname>: Specify custom Nmap data file location  
 --send-eth/--send-ip: Send using raw ethernet frames or IP packets  
 --privileged: Assume that the user is fully privileged  
 --unprivileged: Assume the user lacks raw socket privileges  
 -V: Print version number

---

```
-h: Print this help summary page.
EXAMPLES:
nmap -v -A scanme.nmap.org
nmap -v -sn 192.168.0.0/16 10.0.0.0/8
nmap -v -iR 10000 -Pn -p 80
SEE THE MAN PAGE (https://nmap.org/book/man.html) FOR MORE OPTIONS AND
EXAMPLES
```

---

#### 4.1.58 openvas

OpenVAS is a framework of several services and tools offering a comprehensive and powerful vulnerability scanning and vulnerability management solution.

---

```
# Setting up
[root@ArcheryOS ~]# systemctl restart redis
[root@ArcheryOS ~]# openvas-manage-certs -a
[root@ArcheryOS ~]# greenbone-nvt-sync
[root@ArcheryOS ~]# greenbone-scaphdata-sync
[root@ArcheryOS ~]# greenbone-certdata-sync
[root@ArcheryOS ~]# systemctl start openvas-scanner
[root@ArcheryOS ~]# openvasmd --rebuild --progress
[root@ArcheryOS ~]# openvasmd --create-user=admin --role=Admin
[root@ArcheryOS ~]# openvasmd -p 9390 -a 127.0.0.1
# Start the Greenbone Security Assistant WebUI (optional)
[root@ArcheryOS ~]# gsad -f --listen=127.0.0.1 --mlisten=127.0.0.1
--mport=9390
# Then open 127.0.0.1:9390 in firefox
```

---

#### 4.1.59 ophcrack

Ophcrack is a free open source program that cracks Windows log-in passwords by using LM hashes through rainbow tables. The program includes the ability to import the hashes from a variety of formats, including dumping directly from the SAM files of Windows.

---

```
[root@ArcheryOS ~]# ophcrack -h
ophcrack 3.8.0 by Objectif Securite (http://www.objectif-securite.ch)
```

```
Usage: ophcrack [OPTIONS]
Cracks Windows passwords with Rainbow tables
```

```
-a          disable audit mode (default)
-A          enable audit mode
-b          disable bruteforce
-B          enable bruteforce (default)
-c config_file specify the config file to use
-D          display (lots of!) debugging information
```

---

```

-d dir      specify tables base directory
-e          do not display empty passwords
-f file     load hashes from the specified file (pwdump or session)
-g          disable GUI
-h          display this information
-i          hide usernames
-I          show usernames (default)
-l file     log all output to the specified file
-n num      specify the number of threads to use
-o file     write cracking output to file in pwdump format
-p num      preload (0 none, 1 index, 2 index+end, 3 all default)
-q          quiet mode
-r          launch the cracking when ophcrack starts (GUI only)
-s          disable session auto-saving
-S session_file specify the file to use to automatically save the
           progress of the search
-u          display statistics when cracking ends
-t table1[,a[,b,...]][:table2[,a[,b,...]]]
           specify which table to use in the directory given by -d
-v          verbose
-w dir      load hashes from encrypted SAM file in directory dir
-x file     export data in CSV format to file

```

Example: `ophcrack -g -d /path/to/tables -t xp_free_fast,0,3:vista_free -f in.txt`

Launch ophcrack in command line using tables 0 and 3 in  
 /path/to/tables/xp\_free\_fast and all tables in  
 /path/to/tables/vista\_free  
 and cracks hashes from pwdump file in.txt

---

#### 4.1.60 piCrypt

piCrypt is a small tool used to encrypt small files or folders.

---

```
[root@ArcheryOS ~]# ./piCrypt --help
```

Usage: piCrypt [options] [arguments]

Options:

```

-h, --help: Shows this help screen.
-e, --encrypt: Sets program to encrypt files.
-d, --decrypt: Sets program to decrypt files.
-r, --read: Prints output of file to screen instead of
            writing to file (to be used -d, --decrypt).

```

Arguments:

---

Arguments passed are file or folder to be encrypted or decrypted.  
 If a folder is passed into the encryption option of piCrypt,  
 it will be archived into a .tar.gz file first, and [then](#) encrypted.

---

#### 4.1.61 pixiewps

pixiewps is a tool written in C used to bruteforce offline the WPS PIN exploiting the low or non-existing entropy of some software implementations, the so-called "pixie-dust attack" discovered by Dominique Bongard in summer 2014.

---

```
[root@ArcheryOS ~]# pixiewps -h
```

```
Pixiewps 1.4 WPS pixie-dust attack tool
Copyright (c) 2015-2017, wiire <wi7ire@gmail.com>
```

```
Usage: pixiewps <arguments>
```

Required arguments:

```
-e, --pke      : Enrollee public key
-r, --pkr      : Registrar public key
-s, --e-hash1  : Enrollee hash-1
-z, --e-hash2  : Enrollee hash-2
-a, --authkey  : Authentication session key
-n, --e-nonce  : Enrollee nonce
```

Optional arguments:

```
-m, --r-nonce  : Registrar nonce
-b, --e-bssid  : Enrollee BSSID
-v, --verbosity : Verbosity level 1-3, 1 is quietest      [3]
-o, --output    : Write output to file
-j, --jobs      : Number of parallel threads to use        [Auto]

-h              : Display this usage screen
--help         : Verbose help and more usage examples
-V, --version   : Display version
```

```
--mode N[,... N] : Mode selection, comma separated      [Auto]
--start [mm/]yyyy : Starting date                        (only mode 3) [+1 day]
--end  [mm/]yyyy : Ending date                          (only mode 3) [-1 day]
-f, --force       : Bruteforce full range                (only mode 3)
```

Miscellaneous arguments:

```
-7, --m7-enc    : Recover encrypted settings from M7 (only mode 3)
-5, --m5-enc    : Recover secret nonce from M5          (only mode 3)
```

Example (use `--help` for more):

```
pixiewps -e <pke> -r <pk> -s <e-hash1> -z <e-hash2> -a <authkey> -n
<e-nonce>
```

---

#### 4.1.62 powersploit

PowerSploit is a collection of Microsoft PowerShell modules that can be used to aid penetration testers during all phases of an assessment.

---

```
# The PowerSploit modules/scripts are located in /opt/powersploit
```

---

#### 4.1.63 pyrit

Pyrit allows you to create massive databases of pre-computed WPA/WPA2-PSK authentication phase in a space-time-tradeoff. By using the computational power of Multi-Core CPUs and other platforms through ATI-Stream, Nvidia CUDA and OpenCL, it is currently by far the most powerful attack against one of the world's most used security-protocols.

---

```
[root@Archery0S ~]# pyrit -h
Pyrit 0.5.0 (C) 2008-2011 Lukas Lueg - 2015 John Mora
https://github.com/JPaulMora/Pyrit
This code is distributed under the GNU General Public License v3+
```

Usage: pyrit [options] `command`

Recognized options:

```
-b          : Filters AccessPoint by BSSID
-e          : Filters AccessPoint by ESSID
-h          : Print help for a certain command
-i          : Filename for input ('-' is stdin)
-o          : Filename for output ('-' is stdout)
-r          : Packet capture source in pcap-format
-u          : URL of the storage-system to use
--all-handshakes : Use all handshakes instead of the best one
--aes       : Use AES
```

Recognized commands:

```
analyze          : Analyze a packet-capture file
attack_batch     : Attack a handshake with PMKs/passwords from
the db
attack_cowpatty  : Attack a handshake with PMKs from a
cowpatty-file
attack_db        : Attack a handshake with PMKs from the db
attack_passthrough : Attack a handshake with passwords from a file
batch            : Batchprocess the database
```

---

benchmark	: Determine performance of available cores
benchmark_long (5 minutes)	: Longer and more accurate version of benchmark
check_db	: Check the database <b>for</b> errors
create_essid	: Create a new ESSID
delete_essid	: Delete a ESSID from the database
eval results	: Count the available passwords and matching
export_cowpatty	: Export results to a new cowpatty file
export_hashdb	: Export results to an airolib database
export_passwords	: Export passwords to a file
help	: Print general <b>help</b>
import_passwords	: Import passwords from a file-like <b>source</b>
import_unique_passwords <b>source</b>	: Import unique passwords from a file-like
list_cores	: List available cores
list_essids results	: List all ESSIDs but don't count matching
passthrough	: Compute PMKs and write results to a file
relay	: Relay a storage-url via RPC
selftest results	: Test hardware to ensure it computes correct
serve	: Serve <b>local</b> hardware to other Pyrit clients
strip packets	: Strip packet-capture files to the relevant
stripLive capture- <b>source</b>	: Capture relevant packets from a live
verify	: Verify 10% of the results by recomputation

---

#### 4.1.64 radare2

Radare2 is a complete framework for reverse-engineering and analyzing binaries; composed of a set of small utilities that can be used together or independently from the command line.

---

```
[root@ArcheryOS ~]# radare2 -h
Usage: r2 [-ACdfLMnQstuvwzX] [-P patch] [-p prj] [-a arch] [-b bits]
      [-i file]
      [-s addr] [-B baddr] [-m maddr] [-c cmd] [-e k=v]
      file|pid|--|=
--      run radare2 without opening any file
-      same as 'r2 malloc://512'
=      read file from stdin (use -i and -c to run cmds)
-=     perform !=! command to run all commands remotely
-0     print \x00 after init and every command
-2     close stderr file descriptor (silent warning messages)
-a [arch] set asm.arch
-A     run 'aaa' command to analyze all referenced code
```

---

```

-b [bits]    set asm.bits
-B [baddr]   set base address for PIE binaries
-c 'cmd..'    execute radare command
-C          file is host:port (alias for -c+=http://%s/cmd/)
-d          debug the executable 'file' or running process 'pid'
-D [backend] enable debug mode (e cfg.debug=true)
-e k=v       evaluate config var
-f          block size = file size
-F [binplug] force to use that rbin plugin
-h, -hh      show help message, -hh for long
-H ([var])   display variable
-i [file]    run script file
-I [file]    run script file before the file is opened
-k [OS/kern] set asm.os (linux, macos, w32, netbsd, ...)
-l [lib]     load plugin file
-L          list supported IO plugins
-m [addr]    map file at given address (loadaddr)
-M          do not demangle symbol names
-n, -nn      do not load RBin info (-nn only load bin structures)
-N          do not load user settings and scripts
-q          quiet mode (no prompt) and quit after -i
-Q          quiet mode (no prompt) and quit faster (quickLeak=true)
-p [prj]     use project, list if no arg, load if no file
-P [file]    apply rapatch file and quit
-r [rarun2]  specify rarun2 profile to load (same as -e dbg.profile=X)
-R [rr2rule] specify custom rarun2 directive
-s [addr]    initial seek
-S          start r2 in sandbox mode
-t          load rabin2 info in thread
-u          set bin.filter=false to get raw sym/sec/cls names
-v, -V      show radare2 version (-V show lib versions)
-w          open file in write mode
-x          open without exec-flag (asm.emu will not work), See io.exec
-X          same as -e bin.usextr=false (useful for dyldcache)
-z, -zz     do not load strings or load them even in raw

```

---

#### 4.1.65 reaver

Reaver implements a brute force attack against WiFi Protected Setup which can crack the WPS pin of an access point in a matter of hours and subsequently recover the WPA/WPA2 passphrase.

---

```
[root@ArcheryOS ~]# reaver --help
```

```

Reaver v1.6.5 WiFi Protected Setup Attack Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner
<cheffner@tacnetsol.com>

```

## Required Arguments:

- i, --interface=<wlan> Name of the monitor-mode interface to use
- b, --bssid=<mac> BSSID of the target AP

## Optional Arguments:

- m, --mac=<mac> MAC of the host system
- e, --essid=<ssid> ESSID of the target AP
- c, --channel=<channel> Set the 802.11 channel for the interface  
(implies -f)
- s, --session=<file> Restore a previous session file
- C, --exec=<command> Execute the supplied command upon  
successful pin recovery
- f, --fixed Disable channel hopping
- 5, --5ghz Use 5GHz 802.11 channels
- v, --verbose Display non-critical warnings (-vv or  
-vvv for more)
- q, --quiet Only display critical messages
- h, --help Show help

## Advanced Options:

- p, --pin=<wps pin> Use the specified pin (may be arbitrary  
string or 4/8 digit WPS pin)
- d, --delay=<seconds> Set the delay between pin attempts [1]
- l, --lock-delay=<seconds> Set the time to wait if the AP locks WPS  
pin attempts [60]
- g, --max-attempts=<num> Quit after num pin attempts
- x, --fail-wait=<seconds> Set the time to sleep after 10  
unexpected failures [0]
- r, --recurring-delay=<x:y> Sleep for y seconds every x pin attempts
- t, --timeout=<seconds> Set the receive timeout period [10]
- T, --m57-timeout=<seconds> Set the M5/M7 timeout period [0.40]
- A, --no-associate Do not associate with the AP  
(association must be done by another application)
- N, --no-nacks Do not send NACK messages when out of  
order packets are received
- S, --dh-small Use small DH keys to improve crack speed
- L, --ignore-locks Ignore locked state reported by the  
target AP
- E, --eap-terminate Terminate each WPS session with an EAP  
FAIL packet
- J, --timeout-is-nack Treat timeout as NACK (DIR-300/320)
- F, --ignore-fcs Ignore frame checksum errors
- w, --win7 Mimic a Windows 7 registrar [False]
- K, --pixie-dust Run pixiedust attack
- Z Run pixiedust attack

## Example:

```
reaver -i wlan0mon -b 00:90:4C:C1:AC:21 -vv
```



### 4.1.66 rkhunter

rkhunter is a shell script which carries out various checks on the local system to try and detect known rootkits and malware. It also performs checks to see if commands have been modified, if the system startup files have been modified, and various checks on the network interfaces, including checks for listening applications.

---

```
[root@ArcheryOS ~]# rkhunter --help
```

```
Usage: rkhunter [--check | --unlock | --update | --versioncheck |
--propupd [{filename | directory | package name},...] |
--list [{tests | {lang | languages} | rootkits | perl |
propfiles}] |
--config-check | --version | --help} [options]
```

Current options are:

--append-log	Append to the logfile, <b>do not</b>
overwrite	
--bindir <directory>...	Use the specified <b>command</b> directories
-c, --check	Check the <b>local</b> system
-C, --config-check	Check the configuration file(s),
<b>then exit</b>	
--cs2, --color-set2	Use the second color <b>set for</b> output
--configfile <file>	Use the specified configuration file
--cronjob	Run as a cron job
	(implies -c, --sk and --nocolors
	options)
--dbdir <directory>	Use the specified database directory
--debug	Debug mode
	(Do not use unless asked to <b>do so</b> )
--disable <test>[,<test>...]	Disable specific tests
	(Default is to disable no tests)
--display-logfile	Display the logfile at the end
--enable <test>[,<test>...]	Enable specific tests
	(Default is to <b>enable</b> all tests)
--hash {MD5   SHA1   SHA224   SHA256   SHA384   SHA512	
NONE   <command>}	Use the specified file <b>hash function</b>
	(Default is SHA256)
-h, --help	Display this <b>help</b> menu, <b>then exit</b>
--lang, --language <language>	Specify the language to use
	(Default is English)
--list [tests   languages	List the available <b>test</b> names,
languages,	
rootkits   perl	rootkit names, perl module status
propfiles]	or file properties database, <b>then</b>
<b>exit</b>	
-l, --logfile [file]	Write to a logfile
	(Default is /var/log/rkhunter.log)
--noappend-log	Do not append to the logfile,

---

overwrite it	
--nocf	Do not use the configuration file
entries	
	for disabled tests (only valid with --disable)
--nocolors	Use black and white output
--nolog	Do not write to a logfile
--nomow, --no-mail-on-warning occur	Do not send a message if warnings occur
--ns, --nosummary results	Do not show the summary of check results
--novl, --no-verbose-logging	No verbose logging
--pkgmgr {RPM   DPKG   BSD   BSDng   SOLARIS   NONE}	Use the specified package manager to obtain or verify file property values. (Default is NONE)
--propupd [file   directory   database, package]...	Update the entire file properties database, or just for the specified entries
-q, --quiet	Quiet mode (no output at all)
--rwo, --report-warnings-only	Show only warning messages
--sk, --skip-keypress test	Don't wait for a keypress after each test
--summary results	Show the summary of system check results
	(This is the default)
--syslog [facility.priority] times	Log the check start and finish times to syslog
	(Default level is authpriv.notice)
--tmpdir <directory>	Use the specified temporary directory
--unlock	Unlock (remove) the lock file
--update	Check for updates to database files
--vl, --verbose-logging	Use verbose logging (on by default)
-V, --version	Display the version number, then exit
--versioncheck	Check for latest version of program
-x, --autox	Automatically detect if X is in use
-X, --no-autox	Do not automatically detect if X is in use

---

#### 4.1.67 social-engineer-toolkit

The Social-Engineer Toolkit is an open-source penetration testing framework designed for social engineering. SET has a number of custom attack vectors that allow you to make a believable attack quickly. SET is a product of TrustedSec, LLC an information security consulting firm located in Cleveland, Ohio.

---

```
[root@ArcheryOS ~]# setoolkit
```

---

### 4.1.68 sslscan

sslscan queries SSL/TLS services, such as HTTPS, in order to determine the ciphers that are supported.

---

```
[root@ArcheryOS ~]# sslscan --help
```

```

      _
  _ _ _ _ _ | | _ _ _ _ _ _ _ _
 / _ _ / _ _ | / _ _ / _ _ ' | ' _ \
 \ _ _ \ _ _ \ _ _ \ ( _ | ( _ | | | |
 | _ _ / _ _ / _ _ _ \ _ _ \ _ _ , _ | | |

```

SSLScan is a fast SSL port scanner. SSLScan connects to SSL ports and determines what ciphers are supported, which are the servers preferred ciphers, which SSL protocols are supported and returns the SSL certificate. Client certificates / private key can be configured and output is to text / XML.

Command:

```
sslscan [Options] [host:port | host]
```

Options:

```

--targets=<file>  A file containing a list of hosts to
                  check. Hosts can be supplied with
                  ports (i.e. host:port).
--localip=<ip>    Local IP from which connection should be made
--no-failed       List only accepted ciphers (default
                  is to listing all ciphers).
--tls1            Only check TLSv1 ciphers.
--tls11           Only check TLSv11 ciphers.
--tls12           Only check TLSv12 ciphers.
--pk=<file>       A file containing the private key or
                  a PKCS#12 file containing a private
                  key/certificate pair (as produced by
                  MSIE and Netscape).
--pkpass=<password> The password for the private key or
                  PKCS#12 file.
--certs=<file>    A file containing PEM/ASN1 formatted
                  client certificates.
--renegotiation   Attempt TLS renegotiation
--starttls-ftp    STARTTLS setup for FTP
--starttls-imap   STARTTLS setup for IMAP
--starttls-pop3   STARTTLS setup for POP3
--starttls-smtp   STARTTLS setup for SMTP
--starttls-xmpp   STARTTLS setup for XMPP
--http            Test a HTTP connection.
--bugs            Enable SSL implementation bug work-
                  arounds.

```

---

```
--xml=<file>      Output results to an XML file.
--version          Display the program version.
--verbose          Display verbose output.
--help            Display the help text you are now
                  reading.
```

Example:

```
sslscan 127.0.0.1
```

---

#### 4.1.69 steghide

Steghide is a steganography program that is able to hide data in various kinds of image- and audio-files. The color- respectively sample-frequencies are not changed thus making the embedding resistant against first-order statistical tests.

---

```
[root@ArcheryOS ~]# steghide embed -cf picture.jpg -ef secret.txt
Enter passphrase:
Re-Enter passphrase:
embedding "secret.txt" in "picture.jpg"... done
[root@ArcheryOS ~]# steghide extract -sf picture.jpg
Enter passphrase:
the file "secret.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "secret.txt".
```

---

#### 4.1.70 smbmap

SMBMap allows users to enumerate samba share drives across an entire domain. List share drives, drive permissions, share contents, upload/download functionality, file name auto-download pattern matching, and even execute remote commands. This tool was designed with pen testing in mind, and is intended to simplify searching for potentially sensitive data across large networks.

---

```
[root@ArcheryOS ~]# smbmap --help
usage: smbmap [-h] (-H HOST | --host-file FILE) [-u USERNAME] [-p
      PASSWORD]
           [-s SHARE] [-d DOMAIN] [-P PORT] [-x COMMAND] [-L | -R
           [PATH] |
           -r [PATH]] [-A PATTERN] [-q] [--depth DEPTH] [-F PATTERN]
           [--search-path PATH] [--download PATH] [--upload SRC DST]
           [--delete PATH TO FILE] [--skip]
```

SMBMap - Samba Share Enumerator | Shawn Evans - ShawnDEvans@gmail.com

optional arguments:

```
-h, --help          show this help message and exit
```

Main arguments:

```
-H HOST             IP of host
```

```

--host-file FILE      File containing a list of hosts
-u USERNAME          Username, if omitted null session assumed
-p PASSWORD          Password or NTLM hash
-s SHARE             Specify a share (default C$), ex 'C$'
-d DOMAIN            Domain name (default WORKGROUP)
-P PORT              SMB port (default 445)

```

## Command Execution:

Options for executing commands on the specified host

```

-x COMMAND           Execute a command ex. 'ipconfig /all'

```

## Filesystem Search:

Options for searching/enumerating the filesystem of the specified host

```

-L                  List all drives on the specified host
-R [PATH]           Recursively list dirs, and files (no share\path
                    lists
                    ALL shares), ex. 'C$\Finance'
-r [PATH]           List contents of directory, default is to list
                    root of
                    all shares, ex. -r 'C$\Documents and
                    Settings\Administrator\Documents'
-A PATTERN          Define a file name pattern (regex) that auto
                    downloads
                    a file on a match (requires -R or -r), not case
                    sensitive, ex '(web|global).(aspx|config)'
-q                 Disable verbose output. Only shows shares you have
                    READ/WRITE on, and suppresses file listing when
                    performing a search (-A).
--depth DEPTH       Traverse a directory tree to a specific depth

```

## File Content Search:

Options for searching the content of files

```

-F PATTERN          File content search, -F '[Pp]assword' (requires
                    admin
                    access to execute commands, and powershell on
                    victim
                    host)
--search-path PATH  Specify drive/path to search (used with -F, default
                    C:\Users), ex 'D:\HR\'

```

## Filesystem interaction:

Options for interacting with the specified host's filesystem

```

--download PATH     Download a file from the remote system,
                    ex. 'C$\temp\passwords.txt'
--upload SRC DST     Upload a file to the remote system ex.
                    '/tmp/payload.exe C$\temp\payload.exe'

```

```
--delete PATH TO FILE
                        Delete a remote file, ex. 'C$\temp\msf.exe'
--skip                  Skip delete file confirmation prompt
```

Examples:

```
$ python smbmap.py -u jsmith -p password1 -d workgroup -H 192.168.0.1
$ python smbmap.py -u jsmith -p
  'aad3b435b51404eeaad3b435b51404ee:da76f2c4c96028b7a6111aef4a50a94d'
  -H 172.16.0.20
$ python smbmap.py -u 'apadmin' -p 'asdf1234!' -d ACME -H 10.1.3.30 -x
  'net group "Domain Admins" /domain'
```

#### 4.1.71 sqlmap

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

```
[root@ArcheryOS ~]# sqlmap --help
---
__H__
--- ["]----- {1.2.5#stable}
|_ -| . ["] | .'| . |
|___|_ [D]_|_|_|_|_|_|_|_|
      |_|V          |_| http://sqlmap.org
```

Usage: python sqlmap.py [options]

Options:

```
-h, --help          Show basic help message and exit
-hh                 Show advanced help message and exit
--version            Show program's version number and exit
-v VERBOSE          Verbosity level: 0-6 (default 1)
```

Target:

At least one of these options has to be provided to define the target(s)

```
-u URL, --url=URL Target URL (e.g.
  "http://www.site.com/vuln.php?id=1")
-g GOOGLEDORK       Process Google dork results as target URLs
```

Request:

These options can be used to specify how to connect to the target URL

```
--data=DATA      Data string to be sent through POST
--cookie=COOKIE  HTTP Cookie header value
--random-agent   Use randomly selected HTTP User-Agent header value
--proxy=PROXY    Use a proxy to connect to the target URL
--tor            Use Tor anonymity network
--check-tor      Check to see if Tor is used properly
```

#### Injection:

These options can be used to specify which parameters to **test for**, provide custom injection payloads and optional tampering scripts

```
-p TESTPARAMETER Testable parameter(s)
--dbms=DBMS       Force back-end DBMS to this value
```

#### Detection:

These options can be used to customize the detection phase

```
--level=LEVEL    Level of tests to perform (1-5, default 1)
--risk=RISK       Risk of tests to perform (1-3, default 1)
```

#### Techniques:

These options can be used to tweak testing of specific SQL injection techniques

```
--technique=TECH SQL injection techniques to use (default "BEUSTQ")
```

#### Enumeration:

These options can be used to enumerate the back-end database management system information, structure and data contained **in** the tables. Moreover you can run your own SQL statements

```
-a, --all          Retrieve everything
-b, --banner       Retrieve DBMS banner
--current-user     Retrieve DBMS current user
--current-db       Retrieve DBMS current database
--passwords        Enumerate DBMS users password hashes
--tables           Enumerate DBMS database tables
--columns           Enumerate DBMS database table columns
--schema           Enumerate DBMS schema
--dump             Dump DBMS database table entries
--dump-all         Dump all DBMS databases tables entries
-D DB              DBMS database to enumerate
-T TBL             DBMS database table(s) to enumerate
-C COL            DBMS database table column(s) to enumerate
```

#### Operating system access:

These options can be used to access the back-end database management system underlying operating system

---

```
--os-shell      Prompt for an interactive operating system shell
--os-pwn        Prompt for an OOB shell, Meterpreter or VNC
```

**General:**

These options can be used to set some general working parameters

```
--batch        Never ask for user input, use the default behavior
--flush-session Flush session files for current target
```

**Miscellaneous:**

```
--sqlmap-shell Prompt for an interactive sqlmap shell
--wizard        Simple wizard interface for beginner users
```

---

**4.1.72 swaks**

swaks' primary design goal is to be a flexible, scriptable, transaction-oriented SMTP test tool. It handles SMTP features and extensions such as TLS, authentication, and pipelining; multiple version of the SMTP protocol including SMTP, ESMTP, and LMTP; and multiple transport methods including unix-domain sockets, internet-domain sockets, and pipes to spawned processes.

---

```
[root@ArcheryOS ~]# swaks --help
```

---

**4.1.73 thc-ipv6**

The Hacker Choice's IPv6 Attack Toolkit (aka thc-ipv6)

---

```
[root@ArcheryOS ~]# thc-ipv6-setup.sh
[root@ArcheryOS ~]# thcping6
[root@ArcheryOS ~]# thcsyn6
```

---

**4.1.74 twofi**

twofi uses Twitter to help generate those lists based on searches for keywords related to the list that is being cracked. I've expanded this idea into twofi which will take multiple search terms and return a word list sorted by most common first.

---

```
[root@ArcheryOS ~]# twofi --help
twofi 1.0 Robin Wood (robin@digininja.org) (www.digininja.org)
twofi - Twitter Words Of Interest
```

---

Usage: twofi [OPTIONS]

```
--help, -h: show help
--count, -c: include the count with the words
```



---

```
--min_word_length, -m: minimum word length
--term_file, -T file: a file containing a list of terms
--terms, -t: comma separated usernames
    quote words containing spaces, no space after commas
--user_file, -U file: a file containing a list of users
--users, -u: comma separated search terms
    quote words containing spaces, no space after commas
--verbose, -v: verbose
```

---

#### 4.1.75 unicornscan

Unicornscan is an attempt at a User-land Distributed TCP/IP stack for information gathering and correlation. It is intended to provide a researcher a superior interface for introducing a stimulus into and measuring a response from a TCP/IP enabled device or network. Some of its features include asynchronous stateless TCP scanning with all variations of TCP flags, asynchronous stateless TCP banner grabbing, and active/passive remote OS, application, and component identification by analyzing responses. Like Scanrand, it isn't for the faint of heart.

---

```
[root@ArcheryOS ~]# unicornscan --help
unicornscan (version 0.4.7)
usage: unicornscan [options]
    'b:B:cd:De:EFG:hHi:Ij:l:L:m:M:o:p:P:q:Qr:R:s:St:T:u:Uw:W:vVzZ:' ]
    X.X.X.X/YY:S-E
-b, --broken-crc    *set broken crc sums on [T]ransport layer,
                    [N]etwork layer, or both[TN]
-B, --source-port  *set source port? or whatever the scan module
                    expects as a number
-c, --proc-duplicates process duplicate replies
-d, --delay-type   *set delay type (numeric value, valid options are
                    '1:tsc 2:gtod 3:sleep')
-D, --no-defpayload no default Payload, only probe known protocols
-e, --enable-module *enable modules listed as arguments (output and
                    report currently)
-E, --proc-errors  for processing 'non-open' responses (icmp errors,
                    tcp rst's...)
-F, --try-frags
-G, --payload-group *payload group (numeric) for tcp/udp type payload
                    selection (default all)
-h, --help         help
-H, --do-dns       resolve hostnames during the reporting phase
-i, --interface    *interface name, like eth0 or fxp1, not normally
                    required
-I, --immediate     immediate mode, display things as we find them
-j, --ignore-seq   *ignore 'A'll, 'R'etset sequence numbers for tcp
                    header validation
-l, --logfile       *write to this file not my terminal
```

---

```

-L, --packet-timeout *wait this long for packets to come back
    (default 7 secs)
-m, --mode *scan mode, tcp (syn) scan is default, U for udp T
    for tcp 'sf' for tcp connect scan and A for arp
    for -mT you can also specify tcp flags following
    the T like -mTsFpU for example
    that would send tcp syn packets with (NO
    Syn|FIN|NO Push|URG)
-M, --module-dir *directory modules are found at (defaults to
    /usr/lib/unicornscan/modules)
-o, --format *format of what to display for replies, see man
    page for format specification
-p, --ports global ports to scan, if not specified in target
    options
-P, --pcap-filter *extra pcap filter string for reciever
-q, --covertness *covertness value from 0 to 255
-Q, --quiet dont use output to screen, its going somewhere
    else (a database say...)
-r, --pps *packets per second (total, not per host, and as
    you go higher it gets less accurate)
-R, --repeats *repeat packet scan N times
-s, --source-addr *source address for packets 'r' for random
-S, --no-shuffle do not shuffle ports
-t, --ip-ttl *set TTL on sent packets as in 62 or 6-16 or
    r64-128
-T, --ip-tos *set TOS on sent packets
-u, --debug *debug mask
-U, --no-openclosed dont say open or closed
-w, --safefile *write pcap file of recieved packets
-W, --fingerprint *OS fingerprint 0=cisco(def) 1=openbsd 2=WindowsXP
    3=p0fsendsyn 4=FreeBSD 5=nmap
    6=linux 7:strangetcp
-v, --verbose verbose (each time more verbose so -vvvvv is
    really verbose)
-V, --version display version
-z, --sniff sniff alike
-Z, --drone-str *drone String
*: options with '*' require an argument following them

```

address ranges are cidr like 1.2.3.4/8 for all of 1.?.?.?

if you omit the cidr mask then /32 is implied

port ranges are like 1-4096 with 53 only scanning one port, a for all  
65k and p for 1-1024

example: unicornscan -i eth1 -Ir 160 -E 192.168.1.0/24:1-4000 gateway:a

---

#### 4.1.76 urlcrazy

Generate and test domain typos and variations to detect and perform typo squatting, URL hijacking, phishing, and corporate espionage.

---

```
[root@ArcheryOS ~]# urlcrazy --help
URLCrazy version 0.5
by Andrew Horton (urbanadventurer)
http://www.morningstarsecurity.com/research/urlcrazy
```

Generate and **test** domain typos and variations to detect and perform typo squatting, URL hijacking, phishing, and corporate espionage.

Supports the following domain variations:  
Character omission, character repeat, adjacent character swap, adjacent character replacement, double character replacement, adjacent character insertion, missing dot, strip dashes, singular or pluralise, common misspellings, vowel swaps, homophones, bit flipping (cosmic rays), homoglyphs, wrong top level domain, and wrong second level domain.

Usage: /usr/bin/urlcrazy [options] domain

##### Options

```
-k, --keyboard=LAYOUT Options are: qwerty, azerty, qwertz, dvorak
    (default: qwerty)
-p, --popularity Check domain popularity with Google
-r, --no-resolve Do not resolve DNS
-i, --show-invalid Show invalid domain names
-f, --format=TYPE Human readable or CSV (default: human readable)
-o, --output=FILE Output file
-h, --help This help
-v, --version Print version information. This version is 0.5
```

---

#### 4.1.77 unix-privesc-check

unix-priv-esc is a script to run post exploitation to find misconfigurations and security vulnerabilities, to help find a way to elevate privilege.

---

```
# Move to victim computer, and run:
[root@ArcheryOS ~]# ./unix-privesc-check
```

---

#### 4.1.78 Veil-Evasion

Veil-Evasion is a tool designed to generate metasploit payloads that bypass common anti-virus solutions.

---

```
[root@ArcheryOS ~]# Veil-Evasion-Setup
[root@ArcheryOS ~]# veil-evasion
=====
Veil-Evasion | [Version]: 2.28.2
=====
[Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=====

Main Menu

    51 payloads loaded

Available Commands:

    use          Use a specific payload
    info         Information on a specific payload
    list         List available payloads
    update       Update Veil-Evasion to the latest version
    clean        Clean out payload folders
    checkvt      Check payload hashes vs. VirusTotal
    exit         Exit Veil-Evasion

[menu>>]:
```

---

#### 4.1.79 veracrypt

VeraCrypt is a fork of the discontinued TrueCrypt project. Many security improvements have been implemented and issues raised by TrueCrypt code audits have been fixed. VeraCrypt features optimized implementations of cryptographic hash functions and ciphers which boost performance on modern CPUs.

---

```
[root@ArcheryOS ~]# veracrypt
```

---

#### 4.1.80 wapiti

Wapiti allows you to audit the security of your web applications. It performs "black-box" scans, i.e. it does not study the source code of the application but will scan the webpages of the deployed webapp, looking for scripts and forms where it can inject data. Once it gets this list, Wapiti acts like a fuzzer, injecting payloads to see if a script is vulnerable. Wapiti is useful only to discover vulnerabilities : it is not an exploitation tools. Some well known applications can be used for the exploitation part like the recommended sqlmap.

---

```
[root@ArcheryOS ~]# wapiti --help
```

---

```

Wapiti-3.0.0 (wapiti.sourceforge.net)
usage: wapiti [-h] [-u URL] [--scope {page,folder,domain,url}]
              [-m MODULES_LIST] [--list-modules] [-l LEVEL] [-p PROXY_URL]
              [-a CREDENTIALS] [--auth-type {basic,digest,kerberos,ntlm}]
              [-c COOKIE_FILE] [--skip-crawl] [--resume-crawl]
              [--flush-attacks] [--flush-session] [-s URL] [-x URL]
              [-r PARAMETER] [--skip PARAMETER] [-d DEPTH]
              [--max-links-per-page MAX] [--max-files-per-dir MAX]
              [--max-scan-time MINUTES] [--max-parameters MAX] [-S FORCE]
              [-t SECONDS] [-H HEADER] [-A AGENT] [--verify-ssl {0,1}]
              [--color] [-v LEVEL] [-f FORMAT] [-o OUPUT_PATH]
              [--no-bugreport] [--version]

```

Wapiti-3.0.0: Web application vulnerability scanner

optional arguments:

```

-h, --help            show this help message and exit
-u URL, --url URL     The base URL used to define the scan scope (default
                      scope is folder)
--scope {page,folder,domain,url}
                      Set scan scope
-m MODULES_LIST, --module MODULES_LIST
                      List of modules to load
--list-modules        List Wapiti attack modules and exit
-l LEVEL, --level LEVEL
                      Set attack level
-p PROXY_URL, --proxy PROXY_URL
                      Set the HTTP(S) proxy to use. Supported: http(s)
                      and
                      socks proxies
-a CREDENTIALS, --auth-cred CREDENTIALS
                      Set HTTP authentication credentials
--auth-type {basic,digest,kerberos,ntlm}
                      Set the authentication type to use
-c COOKIE_FILE, --cookie COOKIE_FILE
                      Set a JSON cookie file to use
--skip-crawl          Don't resume the scanning process, attack URLs

```

```

        scanned
        during a previous session
--resume-crawl some      Resume the scanning process (if stopped) even if
                           attacks were previously performed
--flush-attacks          Flush attack history and vulnerabilities for the
                           current session
--flush-session          Flush everything that was previously found for this
                           target (crawled URLs, vulns, etc)
-s URL, --start URL      Adds an url to start scan with
-x URL, --exclude URL    Adds an url to exclude from the scan
-r PARAMETER, --remove PARAMETER
                           Remove this parameter from urls
--skip PARAMETER          Skip attacking given parameter(s)
-d DEPTH, --depth DEPTH  Set how deep the scanner should explore the website
--max-links-per-page MAX Set how many (in-scope) links the scanner should
                           extract for each page
--max-files-per-dir MAX  Set how many pages the scanner should explore per
                           directory
--max-scan-time MINUTES  Set how many minutes you want the scan to last
                           (floats
                           accepted)
--max-parameters MAX    MAX URLs and forms having more than MAX input
                           parameters
                           will be erased before attack.
-S FORCE, --scan-force FORCE
                           Easy way to reduce the number of scanned and
                           attacked
                           URLs. Possible values: paranoid, sneaky, polite,
                           normal, aggressive, insane
-t SECONDS, --timeout SECONDS
                           Set timeout for requests
-H HEADER, --header HEADER
                           Set a custom header to use for every requests
-A AGENT, --user-agent AGENT
                           Set a custom user-agent to use for every requests
--verify-ssl {0,1}      Set SSL check (default is no check)
--color                  Colorize output
-v LEVEL, --verbose LEVEL
                           Set verbosity level (0: quiet, 1: normal, 2:
                           verbose)
-f FORMAT, --format FORMAT
                           Set output format. Supported: json, html (default),
                           txt, openvas, vulneranet, xml
-o OUPUT_PATH, --output OUPUT_PATH

```

---

	Output file or folder
--no-bugreport	Don't send automatic bug report when an attack
module	
	fails
--version	Show program's version number and <code>exit</code>

---

#### 4.1.81 websploit

Websploit is an automatic vulnerability assessment, web crawler and exploiter tool. It is an open source command line utility that composed on modular structure.

---

```
[root@ArcheryOS ~]# websploit
```

---

#### 4.1.82 wfuzz

Wfuzz is a tool designed for bruteforcing Web Applications, it can be used for finding resources not linked (directories, servlets, scripts, etc), bruteforce GET and POST parameters for checking different kind of injections (SQL, XSS, LDAP,etc), bruteforce Forms parameters (User/Password), Fuzzing,etc.

---

```
[root@ArcheryOS ~]# wfuzz --help
*****
* Wfuzz 2.2.9 - The Web Fuzzer *
* *
* Version up to 1.4c coded by: *
* Christian Martorella (cmartorella@edge-security.com) *
* Carlos del ojo (deepbit@gmail.com) *
* *
* Version 1.4d to 2.2.9 coded by: *
* Xavier Mendez (xmendez@edge-security.com) *
*****
```

Usage: wfuzz [options] -z payload,params <url>

FUZZ, ..., FUZZnZ wherever you put these keywords wfuzz will replace them with the values of the specified payload.

FUZZ{baseline\_value} FUZZ will be replaced by baseline\_value. It will be the first request performed and could be used as a base for filtering.

Options:

```
-h/--help      : This help
--help         : Advanced help
--version      : Wfuzz version details
```

```

-e <type>          : List of available
                    encoders/payloads/iterators/printers/scripts

--recipe <filename> : Reads options from a recipe
--dump-recipe <filename> : Prints current options as a recipe
--oF <filename>      : Saves fuzz results to a file. These can be
                    consumed later using the wfuzz payload.

-c                : Output with colors
-v                : Verbose information.
-f filename,printer : Store results in the output file using the
                    specified printer (raw printer if omitted).
-o printer        : Show results using the specified printer.
--interact        : (beta) If selected, all key presses are captured.
                    This allows you to interact with the program.
--dry-run         : Print the results of applying the requests without
                    actually making any HTTP request.
--prev            : Print the previous HTTP requests (only when using
                    payloads generating fuzzresults)

-p addr           : Use Proxy in format ip:port:type. Repeat option
                    for using various proxies.
                    Where type could be SOCKS4, SOCKS5 or HTTP if omitted.

-t N              : Specify the number of concurrent connections (10
                    default)
-s N              : Specify time delay between requests (0 default)
-R depth          : Recursive path discovery being depth the maximum
                    recursion level.
-L,--follow       : Follow HTTP redirections
-Z               : Scan mode (Connection errors will be ignored).
--req-delay N     : Sets the maximum time in seconds the request is
                    allowed to take (CURLOPT_TIMEOUT). Default 90.
--conn-delay N    : Sets the maximum time in seconds the
                    connection phase to the server to take (CURLOPT_CONNECTTIMEOUT).
                    Default 90.

-A               : Alias for --script=default -v -c
--script=        : Equivalent to --script=default
--script=<plugins> : Runs script's scan. <plugins> is a comma
                    separated list of plugin-files or plugin-categories
--script-help=<plugins> : Show help about scripts.
--script-args n1=v1,... : Provide arguments to scripts. ie.
                    --script-args grep.regex=""

-u url           : Specify a URL for the request.
-m iterator      : Specify an iterator for combining payloads
                    \(product by default\)
-z payload       : Specify a payload for each FUZZ keyword used in
                    the form of name\[,parameter\]\[,encoder\].

```



---

```

A list of encoders can be used, ie. md5-sha1. Encoders
    can be chained, ie. md5@sha1.
Encoders category can be used. ie. url
    Use help as a payload to show payload
        plugin's details (you can filter using
        --slice)
--zP <params>      : Arguments for the specified payload (it must be
    preceded by -z or -w).
--slice <filter>    : Filter payload's elements using the specified
    expression. It must be preceded by -z.
-w wordlist         : Specify a wordlist file (alias for -z
    file,wordlist).
-V alltype          : All parameters bruteforcing (allvars and allpost).
    No need for FUZZ keyword.
-X method           : Specify an HTTP method for the request, ie. HEAD
    or FUZZ

-b cookie           : Specify a cookie for the requests. Repeat option
    for various cookies.
-d postdata         : Use post data (ex: "id=FUZZ&catalogue=1")
-H header           : Use header (ex:"Cookie:id=1312321&user=FUZZ").
    Repeat option for various headers.
--basic/ntlm/digest auth : in format "user:pass" or "FUZZ:FUZZ" or
    "domain\FUZZ:FUZZ"

--hc/hl/hw/hh N[,N]+ : Hide responses with the specified
    code/lines/words/chars (Use BBB for taking values from baseline)
--sc/sl/sw/sh N[,N]+ : Show responses with the specified
    code/lines/words/chars (Use BBB for taking values from baseline)
--ss/hs regex       : Show/hide responses with the specified regex
    within the content
--filter <filter>    : Show/hide responses using the specified filter
    expression (Use BBB for taking values from baseline)
--prefilter <filter> : Filter items before fuzzing using the
    specified expression.

```

---

#### 4.1.83 whois

WHOIS is a query and response protocol that is widely used for querying databases that store the registered users or assignees of an Internet resource, such as a domain name, an IP address block, or an autonomous system, but is also used for a wider range of other information.

---

```

[root@ArcheryOS ~]# whois --help
Usage: whois [OPTION]... OBJECT...

```

```

-h HOST, --host HOST connect to server HOST
-p PORT, --port PORT connect to PORT

```

---

```
-H          hide legal disclaimers
--verbose  explain what is being done
--help     display this help and exit
--version  output version information and exit
```

These flags are supported by whois.ripe.net and some RIPE-like servers:

```
-l          find the one level less specific match
-L          find all levels less specific matches
-m          find all one level more specific matches
-M          find all levels of more specific matches
-c          find the smallest match containing a mnt-irt
attribute
-x          exact match
-b          return brief IP address ranges with abuse contact
-B          turn off object filtering (show email addresses)
-G          turn off grouping of associated objects
-d          return DNS reverse delegation objects too
-i ATTR[,ATTR]... do an inverse look-up for specified ATTRIBUTES
-T TYPE[,TYPE]... only look for objects of TYPE
-K          only primary keys are returned
-r          turn off recursive look-ups for contact information
-R          force to show local copy of the domain object even
            if it contains referral
-a          also search all the mirrored databases
-s SOURCE[,SOURCE]... search the database mirrored from SOURCE
-g SOURCE:FIRST-LAST find updates from SOURCE from serial FIRST to LAST
-t TYPE     request template for object of TYPE
-v TYPE     request verbose template for object of TYPE
-q [version|sources|types] query specified server info
```

---

#### 4.1.84 wifite

wifite is used to attack multiple WEP, WPA, and WPS encrypted networks in a row.

---

```
[root@ArcheryOS ~]# wifite --help
```

```
.,'          ';;
.,' ,,'      ';; ,,'  WiFite v2 (r87)
.,' ,,' ,,'  ';; ,,'
:: :: : ( ) : :: :: automated wireless auditor
'. .'. '. /_\ ,,' ,,'
'. .'. /___\ ,,' ,,' designed for Linux
'. .'. /-----\ ,,'
      /-----\
      \
```

```
usage: wifite [-h] [--check CHECK] [--cracked] [--reack] [--all]
              [-i INTERFACE] [--mac] [--mon-iface MONITOR_INTERFACE]
              [-c CHANNEL] [-e ESSID] [-b BSSID] [--showb] [--nodeauth]
```

```

[--power POWER] [--tx TX] [--quiet] [--update] [--wpa]
[--wpat WPAT] [--wpadt WPADT] [--strip] [--crack] [--dict
DIC]
[--aircrack] [--pyrit] [--tshark] [--cowpatty] [--wep]
[--pps PPS] [--wept WEPT] [--chopchop] [--arpreplay]
[--fragment] [--caffelatte] [--p0841] [--hirte]
[--nofakeauth]
[--wepca WEPCA] [--wepsave WEPSAVE] [--wps] [--pixie]
[--wpst WPST] [--wpsratio WPSRATIO] [--wpsretry WPSRETRY]

```

## optional arguments:

```
-h, --help          show this help message and exit
```

## COMMAND:

```

--check CHECK      Check capfile [file] for handshakes.
--cracked          Display previously cracked access points.
--recrack          Include already cracked networks in targets.

```

## GLOBAL:

```

--all              Attack all targets.
--i INTERFACE      Wireless interface for capturing.
--mac              Anonymize MAC address.
--mon-iface MONITOR_INTERFACE
                  Interface already in monitor mode.
--c CHANNEL        Channel to scan for targets.
--e ESSID          Target a specific access point by ssid (name).
--b BSSID          Target a specific access point by bssid (mac).
--showb           Display target BSSIDs after scan.
--nodeauth         Do not deauthenticate clients while scanning
--power POWER      Attacks any targets with signal strength > [pow].
--tx TX            Set adapter TX power level.
--quiet            Do not print list of APs during scan.
--update           Check and update Wifite.

```

## WPA:

```

--wpa              Only target WPA networks (works with --wps --wep).
--wpat WPAT        Time to wait for WPA attack to complete (seconds).
--wpadt WPADT      Time to wait between sending deauth packets
                  (seconds).
--strip            Strip handshake using tshark or pyrit.
--crack            Crack WPA handshakes using [dic] wordlist file.
--dict DIC         Specificy dictionary to use when cracking WPA.
--aircrack         Verify handshake using aircrack.
--pyrit            Verify handshake using pyrit.
--tshark           Verify handshake using tshark.
--cowpatty         Verify handshake using cowpatty.

```

## WEP:

```

--wep              Only target WEP networks.
--pps PPS          Set the number of packets per second to inject.

```

---

```
--wept WEPT      Sec to wait for each attack, 0 implies endless.
--chopchop      Use chopchop attack.
--arpreplay     Use arpreplay attack.
--fragment      Use fragmentation attack.
--caffelatte    Use caffe-latte attack.
--p0841         Use P0842 attack.
--hirte         Use hirte attack.
--nofakeauth    Stop attack if fake authentication fails.
--wepca WEPCA   Start cracking when number of IVs surpass [n].
--wepsave WEPSAVE Save a copy of .cap files to this directory.
```

## WPS:

```
--wps          Only target WPS networks.
--pixie        Only use the WPS PixieDust attack
--wpst WPST    Max wait for new retry before giving up (0: never).
--wpsratio WPSRATIO Min ratio of successful PIN attempts/total retries.
--wpsretry WPSRETRY Max number of retries for same PIN before giving
up.
```

---

## 4.1.85 wireshark-gtk

Wireshark is a GUI network protocol analyzer. It lets you interactively browse packet data from a live network or from a previously saved capture file.

---

```
[root@ArcheryOS ~]# wireshark-gtk
```

---

## 4.1.86 wpscan

WPScan is a black box WordPress vulnerability scanner that can be used to scan remote WordPress installations to find security issues.

---

```
[root@ArcheryOS ~]# wpscan --help
```

---

```

      _ _ _ _ _
     / /   / /   / /
    / /   / /   / /
   / /   / /   / /
  / /   / /   / /
 / /   / /   / /
/ /   / /   / /

```

WordPress Security Scanner by the WPScan Team

Version 2.9.3

Sponsored by Sucuri - <https://sucuri.net>

@\_WPScan\_, @ethicalhack3r, @erwan\_lr, pvd1, @\_FireFart\_

---

Help :

Some values are settable [in](#) a config file, see the `example.conf.json`

```
--update                                Update the database to the latest
    version.
--url      | -u <target url>           The WordPress URL/domain to scan.
--force     | -f                       Forces WPScan to not check if the
    remote site is running WordPress.
--enumerate | -e [option(s)]          Enumeration.
    option :
        u      usernames from id 1 to 10
        u[10-20] usernames from id 10 to 20 (you must write [] chars)
        p      plugins
        vp     only vulnerable plugins
        ap     all plugins (can take a long time)
        tt     timthumbs
        t      themes
        vt     only vulnerable themes
        at     all themes (can take a long time)
Multiple values are allowed : "-e tt,p" will enumerate timthumbs and
    plugins
If no option is supplied, the default is "vt,tt,u,vp"

--exclude-content-based "<regexp or string>"
    Used with the enumeration option, will
    exclude all occurrences based on
    the regexp or string supplied.
    You do not need to provide the regexp
    delimiters, but you must write the
    quotes (simple or double).

--config-file | -c <config file>       Use the specified config file, see the
    example.conf.json.
--user-agent | -a <User-Agent>         Use the specified User-Agent.
--cookie <string>                     String to read cookies from.
--random-agent | -r                   Use a random User-Agent.
--follow-redirection                  If the target url has a redirection, it
    will be followed without asking if you wanted to do so or not
--batch                               Never ask for user input, use the
    default behaviour.
--no-color                             Do not use colors in the output.
--log [filename]                      Creates a log.txt file with WPScan's
    output if no filename is supplied. Otherwise the filename is used
    for logging.
--no-banner                           Prevents the WPScan banner from being
    displayed.
--disable-accept-header               Prevents WPScan sending the Accept HTTP
    header.
--disable-referer                     Prevents setting the Referer header.
--disable-tls-checks                  Disables SSL/TLS certificate
    verification.
--wp-content-dir <wp content dir>     WPScan try to find the content
```

directory (ie wp-content) by scanning the index page, however you can specify it.

Subdirectories are allowed.

--wp-plugins-dir <wp plugins dir> Same thing than --wp-content-dir but [for](#) the plugins directory.

If not supplied, WPScan will use wp-content-dir/plugins.

Subdirectories are allowed

--proxy <[protocol://]host:port> Supply a proxy. HTTP, SOCKS4 SOCKS4A and SOCKS5 are supported.

If no protocol is given (format host:port), HTTP will be used.

--proxy-auth <username:password> Supply the proxy [login](#) credentials.

--basic-auth <username:password> Set the HTTP Basic authentication.

--wordlist | -w <wordlist> Supply a wordlist [for](#) the password brute forcer.

--username | -U <username> Only brute force the supplied username.

--usernames <path-to-file> Only brute force the usernames from the file.

--cache-dir <cache-directory> Set the cache directory.

--cache-ttl <cache-ttl> Typhoeus cache TTL.

--request-timeout <request-timeout> Request Timeout.

--connect-timeout <connect-timeout> Connect Timeout.

--threads | -t <number of threads> The number of threads to use when multi-threading requests.

--max-threads <max-threads> Maximum Threads.

--throttle <milliseconds> Milliseconds to [wait](#) before doing another web request. If used, the --threads should be [set](#) to 1.

--help | -h This [help](#) screen.

--verbose | -v Verbose output.

--version Output the current version and [exit](#).

Examples :

-Further [help](#) ...

```
ruby /opt/wpscan/wpscan.rb --help
```

-Do 'non-intrusive' checks ...

```
ruby /opt/wpscan/wpscan.rb --url www.example.com
```

-Do wordlist password brute force on enumerated users using 50 threads

```
...
ruby /opt/wpscan/wpscan.rb --url www.example.com --wordlist darkc0de.lst
--threads 50
```

-Do wordlist password brute force on the 'admin' username only ...

```
ruby /opt/wpscan/wpscan.rb --url www.example.com --wordlist darkc0de.lst
--username admin
```

```
-Enumerate installed plugins ...
ruby /opt/wpscan/wpscan.rb --url www.example.com --enumerate p

-Enumerate installed themes ...
ruby /opt/wpscan/wpscan.rb --url www.example.com --enumerate t

-Enumerate users ...
ruby /opt/wpscan/wpscan.rb --url www.example.com --enumerate u

-Enumerate installed timthumbs ...
ruby /opt/wpscan/wpscan.rb --url www.example.com --enumerate tt

-Use a HTTP proxy ...
ruby /opt/wpscan/wpscan.rb --url www.example.com --proxy 127.0.0.1:8118

-Use a SOCKS5 proxy ... (cURL >= v7.21.7 needed)
ruby /opt/wpscan/wpscan.rb --url www.example.com --proxy
    socks5://127.0.0.1:9000

-Use custom content directory ...
ruby /opt/wpscan/wpscan.rb -u www.example.com --wp-content-dir
    custom-content

-Use custom plugins directory ...
ruby /opt/wpscan/wpscan.rb -u www.example.com --wp-plugins-dir
    wp-content/custom-plugins

-Update the DB ...
ruby /opt/wpscan/wpscan.rb --update

-Debug output ...
ruby /opt/wpscan/wpscan.rb --url www.example.com --debug-output
    2>debug.log
```

---

#### 4.1.87 zaproxy

zaproxy can help you automatically find security vulnerabilities in your web applications while you are developing and testing your applications. Its also a great tool for experienced pentesters to use for manual security testing.

---

```
[root@Archery0S ~]# zaproxy
```

---

#### 4.1.88 zmap

ZMap is a network tool for scanning the entire Internet (or large samples). ZMap is capable of scanning the entire Internet in around 45 minutes on a gigabit network connection, reaching 98% theoretical line speed.

---

```
[root@ArcheryOS ~]# zmap --help
```

```
zmap 2.1.1
```

A fast Internet-wide scanner.

Usage: zmap [OPTIONS]... [SUBNETS]...

Basic arguments:

```
-p, --target-port=port    port number to scan (for TCP and UDP scans)
-o, --output-file=name    Output file
-b, --blacklist-file=path  File of subnets to exclude, in CIDR
                           notation,
                           e.g. 192.168.0.0/16
-w, --whitelist-file=path  File of subnets to constrain scan to, in
                           CIDR
                           notation, e.g. 192.168.0.0/16
```

Scan options:

```
-r, --rate=pps            Set send rate in packets/sec
-B, --bandwidth=bps       Set send rate in bits/second (supports
                           suffixes
                           G, M and K)
-n, --max-targets=n       Cap number of targets to probe (as a number
                           or
                           a percentage of the address space)
-t, --max-runtime=ses     Cap length of time for sending packets
-N, --max-results=n       Cap number of results to return
-P, --probes=n            Number of probes to send to each IP
                           (default='1')
-c, --cooldown-time=secs  How long to continue receiving after sending
                           last probe (default='8')
-e, --seed=n              Seed used to select address permutation
--retries=n               Max number of times to try to send packet if
                           send fails (default='10')
-d, --dryrun              Don't actually send packets
--shards=N                Set the total number of shards (default='1')
--shard=n                 Set which shard this scan is (0 indexed)
                           (default='0')
```

Network options:

```
-s, --source-port=port|range Source port(s) for scan packets
-S, --source-ip=ip|range    Source address(es) for scan packets
-G, --gateway-mac=addr      Specify gateway MAC address
--source-mac=addr           Source MAC address
-i, --interface=name        Specify network interface to use
-X, --vpn                   Sends IP packets instead of Ethernet (for
                           VPNs)
```

Probe Modules:



-M, --probe-module=name Select probe module (default='tcp\_synscan')  
 --probe-args=args Arguments to pass to probe module  
 --list-probe-modules List available probe modules

## Data Output:

-f, --output-fields=fields Fields that should be output in result set  
 -O, --output-module=name Select output module (default='default')  
 --output-args=args Arguments to pass to output module  
 --output-filter=filter Specify a filter over the response fields to  
     limit what responses get sent to the  
     output  
     module  
 --list-output-modules List available output modules  
 --list-output-fields List all fields that can be output by  
     selected  
     probe module

## Logging and Metadata:

-v, --verbosity=n Level of log detail (0-5) (default='3')  
 -l, --log-file=name Write log entries to file  
 -L, --log-directory=directory Write log entries to a timestamped file  
     in this  
     directory  
 -m, --metadata-file=name Output file for scan metadata (JSON)  
 -u, --status-updates-file=name Write scan progress updates to CSV file  
 -q, --quiet Do not print status updates  
 --disable-syslog Disables logging messages to syslog  
 --notes=notes Inject user-specified notes into scan  
     metadata  
 --user-metadata=json Inject user-specified JSON metadata into  
     scan  
     metadata

## Additional options:

-C, --config=filename Read a configuration file, which can specify  
     any of these options  
     (default='/etc/zmap/zmap.conf')  
 --max-sendto-failures=n Maximum NIC sendto failures before scan is  
     aborted (default='-1')  
 --min-hitratescan Minimum hitrate that scan can hit before  
     is aborted (default='0.0')  
 -T, --sender-threads=n Threads used to send packets (default='1')  
 --cores=STRING Comma-separated list of cores to pin to  
 --ignore-invalid-hosts Ignore invalid hosts in whitelist/blacklist  
     file  
 -h, --help Print help and exit  
 -V, --version Print version and exit

## Examples:

```
zmap -p 80 (scan the Internet for hosts on tcp/80 and output to
          stdout)
zmap -N 5 -B 10M -p 80 (find 5 HTTP servers, scanning at 10 Mb/s)
zmap -p 80 10.0.0.0/8 192.168.0.0/16 -o (scan both subnets on tcp/80)
zmap -p 80 1.2.3.4 10.0.0.3 (scan 1.2.3.4, 10.0.0.3 on tcp/80)
```

## Probe-module (tcp\_synscan) Help:

Probe module that sends a TCP SYN packet to a specific port. Possible classifications are: synack and rst. A SYN-ACK packet is considered a success and a reset packet is considered a failed response.

## Output-module (csv) Help:

By default, ZMap prints out unique, successful IP addresses (e.g., SYN-ACK from a TCP SYN scan) in ASCII form (e.g., 192.168.1.5) to stdout or the specified output file. Internally this is handled by the "csv" output module and is equivalent to running `zmap --output-module=csv --output-fields=saddr --output-filter="success = 1 && repeat = 0"`

---

## 5 Known bugs and issues

### 5.1 Screen size

When first starting ArcheryOS in virtualbox, the screen size will be quite small. To fix this just logout and log back in again (**Mod+Shift+s** then **e**).

### 5.2 Ranger image preview

The default file manager in ArcheryOS, ranger, does not preview images when run as root.

### 5.3 Found more bugs?

If you have found bugs or have any issue with ArcheryOS, please submit an issue to the ArcheryOS github repo.

## **6 Thanks**

Thank you to Icar-u5 [1], who created the wallpaper. See the bibliography for his github account.

## References

- [1] Icar-u5. *Wallpaper*. URL: <https://github.com/Icar-u5>.
- [2] Michael Stapelberg. *i3wm*. 2009-2017. URL: <https://i3wm.org/>.