ArcheryOS

Tovi

June 19, 2018

Contents

1 Introduction

ArcheryOS is a rolling release Arch based distribution with a distinct focus on Pentration testing, privacy, digital forensics and programming. Due to this ArcheryOS not only has full access to the standard Arch and AUR repositories ensuring that software is kept as up to date as possible but also boasts a well rounded toolkit for penetration testing.

1.1 Philosophy of ArcheryOS

- Simple

  ArcheryOS has a simple install process using a curses installer to make the arch install process as easy and painless as possible. ArcheryOS provides you with a choice of several minimal window managers all preconfigured to allow you to get started with next to no configuration required.

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

- Privacy

  Privacy is at the forefront of the philosophy of ArcheryOS and Firefox comes preconfigured to ensure that browser fingerprinting and the possibility of WebRTC IP leaks are minimized.

  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

  Unlike other penetration testing distributions that provide an overwhelming library of tools preinstalled the goal of ArcheryOS is to provide a perfect launching point for a personalized pentesting experience To make this possible ArcheryOS comes preinstalled with a suite of 61 essential tools and extensive documentation along with sane configs to ensure that you can get started immediately and expand your toolkit as required.

2 Installation

ArcheryOS is capable of being used in a liveboot setting as well as being installed onto your machine.
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.**

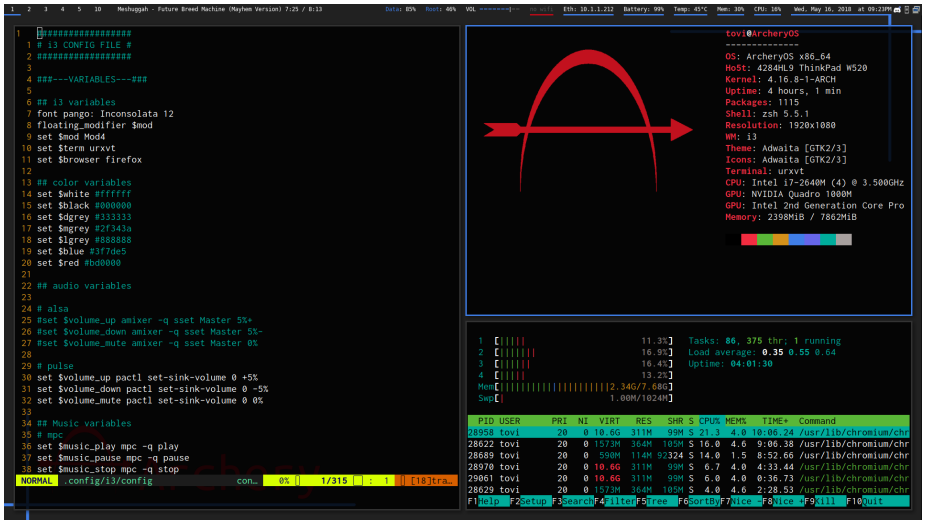

Figure 1: i3 running vim, htop, and neofetch in 3 terminals

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 licensei3 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
- **Mod+`** — Changes to a random wallpaper from /usr/share/wallpapers

3.2.2 Programs

- **Mod+Shift+a** — Audio (pavucontrol)
- **Mod+b** — Browser (firefox)
- **Mod+i** — System information (htop)
- **Mod+Shift+m** — Music (ncmpcpp)
- **Mod+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 of implementation is to 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
```

```
showstringspaces
```

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
```

showstringspaces

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'Otreppe
 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
```

showstringspaces

4.1.4 airdecloak-ng

airuncloak-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'Otreppe
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
```

showstringspaces

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'Otreppe
  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:

    -B       : activates the bitrate test

Source options:

    -i iface : capture packets from this interface
    -r file  : extract packets from this pcap file

Miscellaneous options:

    -R                  : disable /dev/rtc usage
    --ignore-negative-one : if the interface's channel can't be
        determined,
                          ignore the mismatch, needed for unpatched
                              cfg80211
    --deauth-rc rc      : Deauthentication reason code [0-254]
        (Default: 7)

Attack modes (numbers can still be used):

    --deauth     count : deauthenticate 1 or all stations (-0)
    --fakeauth   delay : fake authentication with AP (-1)
    --interactive     : interactive frame selection (-2)
    --arpreplay       : standard ARP-request replay (-3)
    --chopchop        : decrypt/chopchop WEP packet (-4)
    --fragment        : generates valid keystream (-5)
    --caffe-latte     : query a client for new IVs (-6)
    --cfrag           : fragments against a client (-7)
    --migmode         : attacks WPA migration mode (-8)
    --test            : tests injection and quality (-9)

    --help            : Displays this usage screen
```

showstringspaces

4.1.6 airmon-ng

Airmon-ng is a script can be used to enable monitor mode on wireless interfaces. It may also be used to go back from monitor mode to managed mode. Entering the airmon-ng command without parameters will show the interfaces status. It can also list/kill programs that can interfere with the wireless card operation.

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

usage: airmon-ng <start|stop|check> <interface> [channel or frequency]
```

```
showstringspaces
```

4.1.7 airodump-ng

airodump-ng is used for packet capturing of raw 802.11 frames for the intent of using them with aircrack-ng. If you have a GPS receiver connected to the 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'Otreppe
 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
```

showstringspaces

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

--export cowpatty <essid> <file> :
                Export to a cowpatty file.
```

showstringspaces

4.1.9 airserv-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 also supports multiple operating systems.

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

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

  Usage: airserv-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)
```

showstringspaces

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
```

showstringspaces

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
```