Last updated: 25.04.2013

FULL LIST
"an overview with details"

Kali Linux Tools

Digital Forensics
Penetration Testing

@Aleks_Cudars

NB!

- This reference guide describes every tool one by one and is aimed at anyone who wants to get familiar with digital forensics and penetration testing or refresh their knowledge in these areas with tools available in Kali Linux
- Note! I've tried to gather as much information as possible, however, even despite that, some entries don't have information, which I might update if I get more information. Also, mistakes are inevitable
- The purpose was to create the most detailed source of every tool in Kali Linux for quick reference and better understanding
- Some tools fall under several categories, which means that duplicate entries exist in the full ~670 pages long source
- The information about every tool usually consists of: DESCRIPTION, USAGE, EXAMPLE and sometimes OPTIONS and TIPs
- Kali Linux tools are **not limited to Kali Linux** / Backtrack (most can be installed on other Linux distributions taking into consideration all the necessary dependencies. Additionally, some tools are also available on other types of operating systems such as Windows and Mac OS)
- Kali Linux is a new and developing OS – some tools may be added, some - updated, some – removed over time
- It is assumed that all tools are **run as root** (or as administrator) **(in Kali Linux you are root by default)**
- All the information gathered about each tool has been found freely on the Internet and is **publicly available**
- Sources of information are **referenced at the end**
- Most command line tools include **options**, however, due to space considerations, only some tools have options listed (search the internet for options, **read documentation/manual**, **use –h** or **--help**)
- For more information on each tool - search the internet, click on links or check the references at the end

- **PLEASE DO NOT USE KALI LINUX AND THE TOOLS LISTED HERE FOR ANY ILLEGAL OPERATION!**
- **Tools which are specifically aimed at DOS, DDOS or anonymity are rarely used in legitimate engagements, and are therefore not installed by default in Kali Linux**

[01] INFORMATION GATHERING - DNS ANALYSIS

- dnsdict6
- dnsenum
- dnsmap
- dnsrecon
- dnsrevenum6
- dnstracer
- dnswalk
- fierce
- maltego
- nmap
- urlcrazy

dnsdict6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

The tool is used to enumerate domain to get the IPv6 address , if it exists. It is a parallized DNS IPv6 dictionary bruteforcer.

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

USAGE dnsdict6 <url>

USAGE dnsdict6 [-d46] [-s|-m|-l|-x] [-t THREADS] [-D] domain [dictionary-file]

EXAMPLE dnsdict6 google.com

dnsenum

DESCRIPTION The purpose of **dnsenum** is to gather as much information as possible about a domain. The program currently performs the following operations:

- Get the host's address (A record) / get name servers (threaded) / get the MX record (threaded).
- Perform axfr queries on name servers and get BIND versions(threaded).
- Get extra names and subdomains via google scraping (google query = "allinurl: -www site:domain").
- Brute force subdomains from file, can also perform recursion on subdomain that have NS records (all threaded).
- Calculate C class domain network ranges and perform whois queries on them (threaded).
- Perform reverse lookups on network ranges (C class or/and whois netranges) (threaded).
- Write to domain_ips.txt file ip-blocks.

USAGE dnsenum.pl [options] <domain>

EXAMPLE ./dnsenum.pl -p 1 -s 1 google.com

dnsmap

DESCRIPTION The tool enables to discover all subdomains associated to a given domain (e.g. from google.com, it is possible to discover mail.google.com, earth.google.com, sketchup.google.com, desktop.google.com, ...).

USAGE ./dnsmap <target-domain> [options]

EXAMPLE ./dnsmap google.com

dnsrecon

DESCRIPTION **dnsrecon** enables to gather DNS-oriented information on a given target.

At the time of this writing (version 1.6), the tool supports following types:

- Brute force hostnames and subdomains of a given target domain using a wordlist.
- Standard Record Enumeration for a given domain (A, NS, SOA and MX).
- Top Leven Domain Expansion for a given domain.
- Zone Transfer against all NS records of a given domain.
- Reverse Lookup against a given IP Range given a start and end IP.
- SRV Record enumeration

USAGE ./dnsrecon.rb -t <type> -d <target> [options]

EXAMPLE ./dnsrecon.rb -t std -d google.com (Standard (-t std))
EXAMPLE ./dnsrecon.rb -t tld -d aldeid (Top Level Domain (-t tld))
EXAMPLE ./dnsrecon.rb -t axfr -d ??????club.net (Zone transfer (-t axfr))
EXAMPLE ./dnsrecon.rb -t rvs -i 66.249.92.100,66.249.92.150 (Reverse Record Enumeration (-t rvs))

dnsrevenum6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

Simple and fast Reverse DNS Enumerator for IPv6

- detects wildcard DNS servers
- adapts to lossy/slow DNS server
- fast but non-flooding
- specify the reverse domain as 2001:db8::/56 or 0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

USAGE dnsrevenum6 <url>

EXAMPLE dnsrevenum6 google.com

dnstracer

DESCRIPTION **dnstracer** enables to trace a chain of DNS servers to the source. It determines where a given Domain Name Server (DNS) gets its information from, and follows the chain of DNS servers back to the servers which know the data.

USAGE dnstracer [options] name

EXAMPLE dnstracer www.mavetju.org (Search for the A record of www.mavetju.org on your local nameserver)
EXAMPLE dnstracer "-s" . "-q" mx mavetju.or (Search for the MX record of mavetju.org on the root-nameservers)
EXAMPLE dnstracer "-q" ptr 141.230.204.212.in-addr.arpa (Search for the PTR record (hostname) of 212.204.230.141)
EXAMPLE dnstracer "-q" ptr "-s" . "-o“ 2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.6.4.0.2.0.0.0.0.8.b.0.e.f.f.3.ip6.int (for IPv6 addresses)

dnswalk

DESCRIPTION **Dnswalk** is a DNS database debugger. It performs zone transfers of specified domains, and checks the database in numerous ways for internal consistency, as well as for correctness according to accepted practices with the Domain Name System.

The domain name specified on the command line MUST end with a '.'. You can specify a forward domain, such as ***dnswalk*** *podunk.edu. or a reverse domain, such as* ***dnswalk*** *3.2.1.in-addr.arpa.*

USAGE dnswalk [-adilrfFm] <domain>.

EXAMPLE dnswalk google.com

fierce

DESCRIPTION **fierce** is a semi-lightweight enumeration scanner that helps penetration testers locate non-contiguous IP space and hostnames for a specified domains using things like DNS, Whois and ARIN. It's really meant as a pre-cursor to active testing tools via something like: **nmap**, **unicornscan**, **nessus**, **nikto**, etc, since all of those require that you already know what IP space you are looking for. **Fierce** does not perform exploitation and does not scan the whole internet indiscriminately. It is meant specifically to locate likely targets both inside and outside a corporate network.

Since it uses DNS primarily you will often find mis-configured networks that leak internal address space.

USAGE fierce {target options} [OPTIONS]

EXAMPLE fierce -dns company.com (Standard Fierce scan)
EXAMPLE fierce -dns company.com –wide (Standard Fierce scan and search all class c ranges found for PTR names that match the domain)
EXAMPLE fierce -dns company.com -only zt (Fierce scan that only checks for zone transfer)
EXAMPLE fierce -dns company.com –ztstop (Fierce scan that does not perform bruteforcing if a zone transfer is found)
EXAMPLE fierce -dns company.com –wildcstop (Fierce scan that does not perform bruteforcing if a wildcard is found)

maltego

DESCRIPTION **Maltego** is a unique platform developed to deliver a clear threat picture to the environment that an organization owns and operates. **Maltego** can locate, aggregate and visualize this information. **Maltego** is a program that can be used to determine the relationships and real world links between people, groups of people (social networks), companies, organizations, web sites, phrases, affiliations, documents and files, internet infrastructure (domains, DNS names, netblocks, IP addresses).

USAGE n/a, GUI tool

EXAMPLE n/a, GUI tool

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

urlcrazy

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

- Detect typo squatters profiting from typos on your domain name
- Protect your brand by registering popular typos
- Identify typo domain names that will receive traffic intended for another domain
- Conduct phishing attacks during a penetration test

USAGE ./urlcrazy [options] <domain>

EXAMPLE ./urlcrazy example.com

[02] INFORMATION GATHERING - IDS/IPS IDENTIFICATION

- fragroute
- fragrouter
- wafw00f

fragroute

DESCRIPTION **fragroute** intercepts, modifies, and rewrites egress traffic destined for a specified host.

It features a simple ruleset language to delay, duplicate, drop, fragment, overlap, print, reorder, segment, source-route, or otherwise monkey with all outbound packets destined for a target host, with minimal support for randomized or probabilistic behaviour.

This tool was written in good faith to aid in the testing of network intrusion detection systems, firewalls, and basic TCP/IP stack behaviour.

Unlike **fragrouter**, this program only affects packets originating from the local machine destined for a remote host. Do not enable IP forwarding on the local machine.

USAGE fragroute [-f file] <host>

EXAMPLE fragroute 192.168.123.233

fragrouter

DESCRIPTION **Fragrouter** is a network intrusion detection evasion toolkit. It implements most of the attacks described in the Secure Networks "Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection" paper of January 1998.

This program was written in the hopes that a more precise testing methodology might be applied to the area of network intrusion detection, which is still a black art at best.

*To test your firewall(s) using **fragrouter** , you will need two systems in addition to your firewall/packet filter. This is because **fragrouter** cannot by design be run on the same system from which you're testing (according to the documentation, this is to prevent abuse).*

USAGE fragrouter [options]

EXAMPLE fragrouter -F1

wafw00f

DESCRIPTION

Web Application Firewalls (WAFs) can be detected through stimulus/response testing scenarios. Here is a short listing of possible detection methods:

- Cookies: Some WAF products add their own cookie in the HTTP communication.
- Server Cloaking: Altering URLs and Response Headers
- Response Codes: Different error codes for hostile pages/parameters values
- Drop Action: Sending a FIN/RST packet (technically could also be an IDS/IPS)
- Pre Built-In Rules: Each WAF has different negative security signatures

WafW00f is based on these assumptions to determine remote WAFs.

USAGE python wafw00f.py <url>

EXAMPLE python wafw00f.py google.com

[03] INFORMATION GATHERING - LIVE HOST IDENTIFICATION

- alive6
- arping
- cdpsnarf
- detect-new-ip-6
- detect-sniffer6
- dmitry
- dnmap-client
- dnmap-server
- fping
- hping3
- inverse_lookup6
- miranda
- ncat
- netdiscover
- nmap
- passive_discovery6
- thcping6
- wol-e
- xprobe2

alive6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

alive6 shows alive addresses in the segment. If you specify a remote router, the packets are sent with a routing header prefixed by fragmentation.

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

USAGE alive6 [-dlmrS] [-W TIME] [-i FILE] [-o FILE] [-s NUMBER] interface [unicast-or-multicast-address [remote-router]]

EXAMPLE alive6 eth1

arping

DESCRIPTION **arping** pings a destination by sending ARP REQUEST packets to a neighbour host, using a given source address.

USAGE arping [-fqbDUAV] [-c count] [-w timeout] [-I device] [-s source] destination

EXAMPLE arping -f -c 1 -I wlan0 192.168.100.1 (Host 192.168.100.1 is alive -> **Received 1 response(s))**
EXAMPLE arping -f -c 1 -I eth0 192.168.100.2 (Host 192.168.100.2 isn't alive -> **Received 0 response(s))**

cdpsnarf

DESCRIPTION **CDPSnarf** if a network sniffer exclusively written to extract information from CDP packets. It provides all the information a “show cdp neighbors detail” command would return on a Cisco router and even more.
Features: Time intervals between CDP advertisements, Source MAC address, CDP Version, TTL, Checksum, Device ID, Software version, Platform, Addresses, Port ID, Capabilities, Duplex, Save packets in PCAP dump file format, Read packets from PCAP dump files, Debugging information (using the "-d" flag), Tested with IPv4 and IPv6

USAGE cdpsnarf -i <device>

OPTIONS cdpsnarf -h

EXAMPLE ./cdpsnarf eth2

detect-new-ip-6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

This tool detects new IPv6 addresses joining the local network. If script is supplied, it is executed with the detected IPv6 address as option.

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

USAGE detect-new-ip6 <interface> [script]

EXAMPLE detect-new-ip6 eth0

detect-sniffer6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

detect-sniffer6 - tests if systems on the local LAN are sniffing. Works against Windows, Linux, OS/X and *BSD. If no target is given, the link-local-all-nodes address is used, which however rarely works.

USAGE detect-sniffer6 interface [target6]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

DMitry

DESCRIPTION **DMitry** has the ability to gather as much information as possible about a host. Base functionality is able to gather possible subdomains, email addresses, uptime information, TCP port scan, whois lookups, and more. The information are gathered with following methods:

- Perform an Internet Number whois lookup.
- Retrieve possible uptime data, system and server data.
- Perform a SubDomain search on a target host.
- Perform an E-Mail address search on a target host.
- Perform a TCP Portscan on the host target.
- A Modular program allowing user specified modules

USAGE dmitry [options] <file> <url>

EXAMPLE dmitry –help (DMitry help)
EXAMPLE man dmitry (DMitry complete documentation)
EXAMPLE dmitry -iwns -o example.out google.com

dnmap

DESCRIPTION **dnmap** is a framework to distribute **nmap** scans among several clients. It reads an already created file with **nmap** commands and send those commands to each client connected to it.

The framework use a client/server architecture. The server knows what to do and the clients do it. All the logic and statistics are managed in the server. **Nmap** output is stored on both server and client.

Usually you would want this if you have to scan a large group of hosts and you have several different internet connections (or friends that want to help you).

- Clients can be run on any computer on Internet. Do not have to be on a local cluster or anything.
- It uses the TLS protocol for encryption.

BASIC USAGE

1. Put some **nmap** commands on a file like **commands.txt**
2. ./dnmap_server -f commands.txt (Start the dnmap_server)
3. ./dnmap_client -s <server-ip> -a <alias> (Start any number of clients)

dnmap-client

DESCRIPTION

- If the server gets down, it keeps connecting to it until it gets up again.
- Strip strange characters from the command sent by the server. Tries to avoid command injection vulns.
- It only executes the nmap command. It deletes the command send by the server and changes it by the known and trusted nmap binary on the system.
- You can select an alias for your user.
- You can change which port the client connects to.
- If the command sent by the server does not have a -oA option, the client add it anyway to the command, so it will always have a local copy of the output.

USAGE ./dnmap_client -s <server-ip> -a <alias> (start any number of clients)

EXAMPLE (see dnmap)

dnmap-server

DESCRIPTION

- If the server gets down, clients continue trying to connect until the server gets back online.
- If the server gets down, when you put it up again it will send commands starting from the last command given before the shutdown. You do not need to remember where it was.
- You can add new commands to the original file without having to stop the server. The server will read them automatically.
- If some client goes down, the server will remember which command it was executing and it will re-schedule it for later.
- It will store every detail of the operations in a log file.
- It shows real time statistics about the operation of each client

You can choose which port to use. Defaults to 46001. Only the Online clients are shown in the running stats.

USAGE ./dnmap_server -f commands.txt (start dnmap server)

EXAMPLE (see dnmap)

fping

DESCRIPTION **fping** is a program like **ping** which uses the Internet Control Message Protocol (ICMP) echo request to determine if a target host is responding.

Fping differs from **ping** in that you can specify any number of targets on the command line, or specify a file containing the lists of targets to ping. Instead of sending to one target until it times out or replies, **fping** will send out a ping packet and move on to the next target in a round-robin fashion.

In the default mode, if a target replies, it is noted and removed from the list of targets to check; if a target does not respond within a certain time limit and/or retry limit it is designated as unreachable. **Fping** also supports sending a specified number of pings to a target, or looping indefinitely (as in **ping**).

Unlike **ping**, **fping** is meant to be used in scripts, so its output is designed to be easy to parse.

USAGE fping [options] [targets...]

EXAMPLE fping 192.168.100.1 (Responding host -> **192.168.100.1 is alive**)

EXAMPLE fping 192.168.100.13 (Non-responding host -> **192.168.100.13 is unreachable**)

hping3

DESCRIPTION **hping3** is a network tool able to send custom ICMP/UDP/TCP packets and to display target replies like ping do with ICMP replies. **Hping3** handles fragmentation, arbitrary packet body and size and can be used in order to transfer files under supported protocols.

Hping3 can be used, among other things to: Test firewall rules, [spoofed] port scanning, test net performance using differents protocols, packet size, TOS (type of service) and fragmentation, path MTU discovery, files transferring even between really fascist firewall rules, traceroute like under different protocols, firewalk like usage, remote OS fingerprint, TCP/IP stack auditing

USAGE hping3 <host> [options]

EXAMPLE hping3 192.168.100.1 -c 1 -I wlan0 -S -p 22 (Following command checks the status of port 22/tcp with a TCP SYN scan)

EXAMPLE hping3 192.168.100.1 -c 1 -I wlan0 -S -p 81 (Following command sends a TCP SYN packet to port 81/tcp on host 192.168.100.1)

EXAMPLE hping3 192.168.100.1 -I wlan0 -S --scan 20,21,22,80,8080 -V (Scan mode)

inverse_lookup6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

inverse_lookup6 - performs an inverse address query, to get the IPv6 addresses that are assigned to a MAC address. Note that only few systems support this yet.

USAGE inverse_lookup6 interface mac-address

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

miranda

DESCRIPTION **Miranda** is a tool that uses the UPnP(universal plug and play) protocol to enumerate the target modem (if you found some routers and firewalls running the UPnP IGD protocol are vulnerable to attack).
*Before working with **Miranda** you should have moderate knowledge of UPnP.*

BASIC USAGE

1. root@root:/pentest/enumeration/miranda#
2. # ./miranda.py
3. upnp> msearch (search for that device with the UPnP port open)
4. upnp> host info 0 (this command will tell you various information about your target – name, protocol, server type, UPnP server)
5. upnp> host get 0 (enumerates targets if possible)
6. upnp> host summary 0 (get full details of your target after you have enumerated it)
7. upnp> host info 0 devicelist WANConnectionDevice services WANPPPConnection actions (this command will tell you about the services that are running on the TARGET)
8. upnp> host send 0 WANConnectionDevice WANPPPConnection ForceTermination (terminate the internet all oevr the network)
9. upnp> host send 0 WANConnectionDevice WANPPPConnection RequestConnection (re-enable internet)

ncat

DESCRIPTION **ncat** is a general-purpose command-line tool for reading, writing, redirecting, and encrypting data across a network. It aims to be your network Swiss Army knife, handling a wide variety of security testing and administration tasks.

Ncat can:

- Act as a simple TCP/UDP/SCTP/SSL client for interacting with web/telnet/mail/TCP/IP servers and services
- Act as a simple TCP/UDP/SCTP/SSL server for offering services to clients, or simply to understand what existing clients are up to by capturing every byte they send.
- Redirect or proxy TCP/UDP/SCTP traffic to other ports or hosts.
- Encrypt communication with SSL, and transport it over IPv4 or IPv6.
- Act as a network gateway for execution of system commands, with I/O redirected to the network.
- Act as a connection broker, allowing two (or far more) clients to connect to each other through a third (brokering) server.

USAGE ncat [options] <url>

EXAMPLE ncat -C mail.example.com 25 (sending email to an SMTP server. Read manual for further steps)

EXAMPLE ncat -l localhost 143 --sh-exec "ncat --ssl imap.example.com 993“ (connecting to an IMPA server that requires SSL . Read manual for further steps)

netdiscover

DESCRIPTION **Netdiscover** is an active/passive address reconnaissance tool, mainly developed for those wireless networks without dhcp server, when you are wardriving. It can be also used on hub/switched networks.

Built on top of **libnet** and **libpcap**, it can passively detect online hosts, or search for them, by actively sending arp requests, it can also be used to inspect your network arp traffic, or find network addresses using auto scan mode, which will scan for common local networks.

USAGE netdiscover [-i device] [-r range | -p] [-s time] [-n node] [-c count] [-f] [-S]

EXAMPLE netdiscover -i wlan0 -r 192.168.1.0/24 (Scan a class C network, to see which hosts are up)
EXAMPLE netdiscover -i wlan0 -r 192.168.0.0/16 (Scanning /16 network, trying to find online boexes)
EXAMPLE netdiscover -i wlan0 -r 10.0.0.0/8 (Scan a class A network, trying to find network addresses)
EXAMPLE netdiscover -i wlan0 (Auto scan common networks)
EXAMPLE netdiscover -i wlan0 -p (Don't send arp requests, listen only)

TIP

(If you want to change your mac address for the scan)
 # ifconfig wlan0 down
 # ifconfig wlan0 hw ether 00:11:22:33:44:55
 # ifconfig wlan0 up
 # netdiscover -i wlan0 [options]

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a Swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

passive_discovery6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

passive_discovery6 - passivly sniffs the network and dump all client's IPv6 addresses detected. Note that in a switched environment you get better results when additionally\nstarting parasite6, however this will impact the network. If a script name is specified after the interface, it is called with the\ndetected ipv6 address as first and the interface as second option.

USAGE passive_discovery6 [-Ds] [-m maxhop] [-R prefix] interface [script]

OPTIONS

-D do also dump destination addresses (does not work with -m)
-s do only print the addresses, no other output
-m maxhop the maximum number of hops a target which is dumped may be away.
0 means local only, the maximum amount to make sense is usually 5
-R prefix exchange the defined prefix with the link local prefix

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

thcping6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

With **thcping6** we can carft a custom ICMPv6 packet, with being able to configure almost any field in the header, at least the most important ones. You can put an "x" into src6, srcmac and dstmac for an automatic value.

USAGE thcping6 <interface> <source-ipv6> <destination-ipv6>

USAGE [-af] [-H o:s:v] [-D o:s:v] [-F dst] [-t ttl] [-c class] [-l label] [-d size] [-S port|-U port] interface src6 dst6 [srcmac [dstmac [data]]]

OPTIONS https://github.com/mmoya/thc-ipv6/blob/master/thcping6.c

EXAMPLE thcping6 eth0 2002:5cf9:8214:e472:a00:27ff:fe37:b032 2002:5cf9:8214:e472:290:a9ff:feb0:cac6

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

wol-e

DESCRIPTION **WOL-E** is a suite of tools for the Wake on LAN feature of network attached computers, this is now enabled by default on many Apple computers. These tools include bruteforcing the MAC address to wake up clients, sniffing WOL attempts and passwords, scanning for Apple devices and more.

If you do not specify a broadcast address or port, wol-e will set the following as defaults for you:

- *Port: 9*
- *Broadcast: 255.255.255.255*

If a password is required use the -k 00:12:34:56:78:90 at the end of the above command.

USAGE python wol-e.py -f

EXAMPLE ./wol-e.py -m 00:12:34:56:78:90 -b 192.168.1.255 -p 9 (To wake up a single computer)
EXAMPLE ./wol-e.py -s -i eth0 (To sniff the network for WOL traffic)
EXAMPLE ./wol-e.py –a (To bruteforce the network)
EXAMPLE ./wol-e.py –f (If you want to scan the network for Apple devices on your subnet)
EXAMPLE wol-e.py –fa (If you want to attempt to wake all targets found from using -f)

xprobe2

DESCRIPTION **xprobe2** is a remote active operating system fingerprinting tool. **Xprobe2** relies on fuzzy signature matching, probabilistic guesses, multiple matches simultaneously, and a signature database.

USAGE xprobe2 [-v] [-r] [-p proto:portnum:state] [-c configfile] [-o logfile] [-p port] [-t receive_timeout] [-m numberofmatches] [-D modnum] [-F] [-X] [-B] [-A] [-T port spec] [-U port spec] host

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (Will launch an OS fingerprinting attempt targeting 192.168.1.10. Modules 1 and 2, which are reachability tests, will be disabled, so probes will be sent even if target is down. Output will be verbose.)

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (Will launch an OS fingerprint attempt targeting 192.168.1.20. The UDP destination port is set to 53, and the output will be verbose.)

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (Will only enable TCP handshake module (number 11) to probe the target, very useful when all ICMP traffic is filtered.)

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (Will cause TCP handshake module to try blindly guess open port on the target by sequentially sending TCP packets to the most likely open ports (80, 443, 23, 21, 25, 22, 139, 445 and 6000).)

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (Will enable portscanning module, which will scan TCP ports starting from 1 to 1024 on 127.0.0.1)

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (If remote target has TCP port 139 open, the command line above will enable application level SMB module (if remote target has TCP port 445 open, substitute 139 in the command line with 445).)

EXAMPLE xprobe2 -v -D 1 -D 2 192.168.1.10 (Will enable SNMPv2c application level module, which will try to retrieve sysDescr.0 OID using community strings taken from xprobe2.conf file.)

[04] INFORMATION GATHERING - NETWORK SCANNERS

- dmitry
- dnmap-client
- dnmap-server
- netdiscover
- nmap

DMitry

DESCRIPTION **DMitry** has the ability to gather as much information as possible about a host. Base functionality is able to gather possible subdomains, email addresses, uptime information, TCP port scan, whois lookups, and more. The information are gathered with following methods:

- Perform an Internet Number whois lookup.
- Retrieve possible uptime data, system and server data.
- Perform a SubDomain search on a target host.
- Perform an E-Mail address search on a target host.
- Perform a TCP Portscan on the host target.
- A Modular program allowing user specified modules

USAGE dmitry [options] <file> <url>

EXAMPLE dmitry –help (DMitry help)
EXAMPLE man dmitry (DMitry complete documentation)
EXAMPLE dmitry -iwns -o example.out google.com

dnmap

DESCRIPTION **dnmap** is a framework to distribute nmap scans among several clients. It reads an already created file with nmap commands and send those commands to each client connected to it.

The framework use a client/server architecture. The server knows what to do and the clients do it. All the logic and statistics are managed in the server. Nmap output is stored on both server and client.

Usually you would want this if you have to scan a large group of hosts and you have several different internet connections (or friends that want to help you).

- Clients can be run on any computer on Internet. Do not have to be on a local cluster or anything.
- It uses the TLS protocol for encryption.

BASIC USAGE

1. Put some **nmap** commands on a file like **commands.txt**
2. ./dnmap_server -f commands.txt (Start the dnmap_server)
3. ./dnmap_client -s <server-ip> -a <alias> (Start any number of clients)

dnmap-client

DESCRIPTION

- If the server gets down, it keeps connecting to it until it gets up again.
- Strip strange characters from the command sent by the server. Tries to avoid command injection vulns.
- It only executes the nmap command. It deletes the command send by the server and changes it by the known and trusted nmap binary on the system.
- You can select an alias for your user.
- You can change which port the client connects to.
- If the command sent by the server does not have a -oA option, the client add it anyway to the command, so it will always have a local copy of the output.

USAGE ./dnmap_client -s <server-ip> -a <alias> (start any number of clients)

EXAMPLE (see dnmap)

dnmap-server

DESCRIPTION

- If the server gets down, clients continue trying to connect until the server gets back online.
- If the server gets down, when you put it up again it will send commands starting from the last command given before the shutdown. You do not need to remember where it was.
- You can add new commands to the original file without having to stop the server. The server will read them automatically.
- If some client goes down, the server will remember which command it was executing and it will re-schedule it for later.
- It will store every detail of the operations in a log file.
- It shows real time statistics about the operation of each client

You can choose which port to use. Defaults to 46001. Only the Online clients are shown in the running stats.

USAGE ./dnmap_server -f commands.txt (start dnmap server)

EXAMPLE (see dnmap)

netdiscover

DESCRIPTION **Netdiscover** is an active/passive address reconnaissance tool, mainly developed for those wireless networks without dhcp server, when you are wardriving. It can be also used on hub/switched networks.

Built on top of **libnet** and **libpcap**, it can passively detect online hosts, or search for them, by actively sending arp requests, it can also be used to inspect your network arp traffic, or find network addresses using auto scan mode, which will scan for common local networks.

USAGE netdiscover [-i device] [-r range | -p] [-s time] [-n node] [-c count] [-f] [-S]

EXAMPLE netdiscover -i wlan0 -r 192.168.1.0/24 (Scan a class C network, to see which hosts are up)
EXAMPLE netdiscover -i wlan0 -r 192.168.0.0/16 (Scanning /16 network, trying to find online boexes)
EXAMPLE netdiscover -i wlan0 -r 10.0.0.0/8 (Scan a class A network, trying to find network addresses)
EXAMPLE netdiscover -i wlan0 (Auto scan common networks)
EXAMPLE netdiscover -i wlan0 -p (Don't send arp requests, listen only)

TIP

(If you want to change your mac address for the scan)
 # ifconfig wlan0 down
 # ifconfig wlan0 hw ether 00:11:22:33:44:55
 # ifconfig wlan0 up
 # netdiscover -i wlan0 [options]

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a Swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

[05] INFORMATION GATHERING - OS FINGERPRINTING

- dnmap-client
- dnmap-server
- miranda
- nmap

dnmap

DESCRIPTION **dnmap** is a framework to distribute **nmap** scans among several clients. It reads an already created file with **nmap** commands and send those commands to each client connected to it.

The framework use a client/server architecture. The server knows what to do and the clients do it. All the logic and statistics are managed in the server. **Nmap** output is stored on both server and client.

Usually you would want this if you have to scan a large group of hosts and you have several different internet connections (or friends that want to help you).

- Clients can be run on any computer on Internet. Do not have to be on a local cluster or anything.
- It uses the TLS protocol for encryption.

BASIC USAGE

1. Put some **nmap** commands on a file like **commands.txt**
2. ./dnmap_server -f commands.txt (Start the dnmap_server)
3. ./dnmap_client -s <server-ip> -a <alias> (Start any number of clients)

dnmap-client

DESCRIPTION

- If the server gets down, it keeps connecting to it until it gets up again.
- Strip strange characters from the command sent by the server. Tries to avoid command injection vulns.
- It only executes the **nmap** command. It deletes the command send by the server and changes it by the known and trusted **nmap** binary on the system.
- You can select an alias for your user.
- You can change which port the client connects to.
- If the command sent by the server does not have a -oA option, the client add it anyway to the command, so it will always have a local copy of the output.

USAGE ./dnmap_client -s <server-ip> -a <alias> (start any number of clients)

EXAMPLE (see dnmap)

dnmap-server

DESCRIPTION

- If the server gets down, clients continue trying to connect until the server gets back online.
- If the server gets down, when you put it up again it will send commands starting from the last command given before the shutdown. You do not need to remember where it was.
- You can add new commands to the original file without having to stop the server. The server will read them automatically.
- If some client goes down, the server will remember which command it was executing and it will re-schedule it for later.
- It will store every detail of the operations in a log file.
- It shows real time statistics about the operation of each client

You can choose which port to use. Defaults to 46001. Only the Online clients are shown in the running stats.

USAGE ./dnmap_server -f commands.txt (start dnmap server)

EXAMPLE (see dnmap)

miranda

DESCRIPTION **Miranda** is a tool that uses the UPnP (universal plug and play) protocol to enumerate the target modem (if you found some routers and firewalls running the UPnP IGD protocol are vulnerable to attack).
*Before working with **Miranda** you should have moderate knowledge of UPnP.*

BASIC USAGE

1. root@root:/pentest/enumeration/miranda#
2. # ./miranda.py
3. upnp> msearch (search for that device with the UPnP port open)
4. upnp> host info 0 (this command will tell you various information about your target – name, protocol, server type, UPnP server)
5. upnp> host get 0 (enumerates targets if possible)
6. upnp> host summary 0 (get full details of your target after you have enumerated it)
7. upnp> host info 0 devicelist WANConnectionDevice services WANPPPConnection actions (this command will tell you about the services that are running on the TARGET)
8. upnp> host send 0 WANConnectionDevice WANPPPConnection ForceTermination (terminate the internet all oevr the network)
9. upnp> host send 0 WANConnectionDevice WANPPPConnection RequestConnection (re-enable internet)

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a Swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

[06] INFORMATION GATHERING - OSINT ANALYSIS

- casefile
- creepy
- dmitry
- jigsaw
- maltego
- metagoofil
- theharvester
- twofi
- urlcrazy

casefile

DESCRIPTION **CaseFile** gives you the ability to quickly add, link and analyse data having the same graphing flexibility and performance as **Maltego** without the use of transforms. Combining **Maltego's** fantastic graph and link analysis this tool allows for analysts to examine links between manually added data to mind map your information.

- **CaseFile** is a visual intelligence application that can be used to determine the relationships and real world links between hundreds of different types of information.
- It gives you the ability to quickly view second, third and n-th order relationships and find links otherwise undiscoverable with other types of intelligence tools.
- **CaseFile** comes bundled with many different types of entities that are commonly used in investigations allowing you to act quickly and efficiently. **CaseFile** also has the ability to add custom entity types allowing you to extend the product to your own data sets.

USAGE n/a; GUI tool

EXAMPLE n/a, GUI tool

creepy

DESCRIPTION **creepy** is an application that allows you to gather geolocation related information about users from social networking platforms and image hosting services. The information is presented in a map inside the application where all the retrieved data is shown accompanied with relevant information (i.e. what was posted from that specific location) to provide context to the presentation. As you can see Cree.py is just that – CREEPY, but what a great tool to gather information and building profiles on targets.

USAGE n/a, GUI tool

EXAMPLE n/a, GUI tool

DMitry

DESCRIPTION **DMitry** has the ability to gather as much information as possible about a host. Base functionality is able to gather possible subdomains, email addresses, uptime information, TCP port scan, whois lookups, and more. The information are gathered with following methods:

- Perform an Internet Number whois lookup.
- Retrieve possible uptime data, system and server data.
- Perform a SubDomain search on a target host.
- Perform an E-Mail address search on a target host.
- Perform a TCP Portscan on the host target.
- A Modular program allowing user specified modules

USAGE dmitry [options] <file> <url>

EXAMPLE dmitry –help (DMitry help)
EXAMPLE man dmitry (DMitry complete documentation)
EXAMPLE dmitry -iwns -o example.out google.com

jigsaw

DESCRIPTION **jigsaw** is a simple ruby script for enumerating information about a company's employees. It is useful for Social Engineering or Email Phishing.

USAGE jigsaw [options] <url>

EXAMPLE jigsaw -s Google
EXAMPLE ./jigsaw.rb -i 215043 -r google -d google.com

maltego

DESCRIPTION **Maltego** is a unique platform developed to deliver a clear threat picture to the environment that an organization owns and operates. **Maltego** can locate, aggregate and visualize this information. **Maltego** is a program that can be used to determine the relationships and real world links between people, groups of people (social networks), companies, organizations, web sites, phrases, affiliations, documents and files, internet infrastructure (domains, DNS names, netblocks, IP addresses).

USAGE n/a, GUI tool

EXAMPLE n/a, GUI tool

metagoofil

DESCRIPTION **Metagoofil** is an information gathering tool designed for extracting metadata of public/indexed documents (pdf,doc,xls,ppt,odp,ods) available in the target/victim websites.
The output is a file that can reveal:

- relevant metadata information
- usernames (potential targets for brute force attacks on open services like ftp, pop3, auths in web apps, ...)
- list of disclosed paths in the metadata

USAGE python metagoofil.py <option>

OPTIONS

- **-d <domain>** Domain to search
- **-f <type>** Filetype to download (all,pdf,doc,xls,ppt,odp,ods, etc)
- **-l <number>** Limit of results to work with (default 100)
- **-o <path>** Output file (html format)
- **-t <path>** Target directory to download files

EXAMPLE python metagoofil.py \ -d ******club.net \ -l 100 \ -f all \ -o output.html \ -t output-files

theharvester

DESCRIPTION **TheHarvester** aims at gathering e-mail accounts and subdomain names from:

- google (www.google.com)
- bing (search.msn.com)
- pgp (pgp.rediris.es)

USAGE theharvester [options]

OPTIONS

- **-d <domain>** domain to search or company name
- **-b <src>** data source (google,bing,pgp,linkedin)
- **-s <start>** start in result number X (default 0)
- **-v** verify host name via DNS resolution
- **-l <limit>** limit the number of results to work with (bing goes from 50 to 50 results, Google 100 to 100, and pgp doesn't use this option)

EXAMPLE ./theHarvester.py -d microsoft.com -l 500 -b bing

twofi

DESCRIPTION **T**witter **W**ords **O**f **I**nterest - **twofi** uses Twitter to help generate lists based on searches for keywords related to the list that is being cracked. An expanded idea is being used in **twofi** which will take multiple search terms and return a word list sorted by most common first. Also given a list of twitter usernames the script will bring back approximately the last 500 tweets for each user and use those to create the list.

USAGE term1,term2,term3 ,...(no spaces)
USAGE username1,username2,username3 ,...(no spaces and no @)

OPTIONS text
--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 search terms quote words containing spaces, no space after commas
--user_file, -U file: a file containing a list of users
--users, -u: comma separated usernames quote words containing spaces, no space after commas
--verbose, -v: verbose

urlcrazy

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

- Detect typo squatters profiting from typos on your domain name
- Protect your brand by registering popular typos
- Identify typo domain names that will receive traffic intended for another domain
- Conduct phishing attacks during a penetration test

USAGE ./urlcrazy [options] <domain>

EXAMPLE ./urlcrazy example.com

[07] INFORMATION GATHERING - ROUTE ANALYSIS

- dnmap-client
- dnmap-server
- intrace
- netmask
- trace6

dnmap

DESCRIPTION **dnmap** is a framework to distribute nmap scans among several clients. It reads an already created file with nmap commands and send those commands to each client connected to it.

The framework use a client/server architecture. The server knows what to do and the clients do it. All the logic and statistics are managed in the server. Nmap output is stored on both server and client.

Usually you would want this if you have to scan a large group of hosts and you have several different internet connections (or friends that want to help you).

- Clients can be run on any computer on Internet. Do not have to be on a local cluster or anything.
- It uses the TLS protocol for encryption.

BASIC USAGE

1. Put some **nmap** commands on a file like **commands.txt**
2. ./dnmap_server -f commands.txt (Start the dnmap_server)
3. ./dnmap_client -s <server-ip> -a <alias> (Start any number of clients)

dnmap-client

DESCRIPTION

- If the server gets down, it keeps connecting to it until it gets up again.
- Strip strange characters from the command sent by the server. Tries to avoid command injection vulns.
- It only executes the nmap command. It deletes the command send by the server and changes it by the known and trusted nmap binary on the system.
- You can select an alias for your user.
- You can change which port the client connects to.
- If the command sent by the server does not have a -oA option, the client add it anyway to the command, so it will always have a local copy of the output.

USAGE ./dnmap_client -s <server-ip> -a <alias> (start any number of clients)

EXAMPLE (see dnmap)

dnmap-server

DESCRIPTION

- If the server gets down, clients continue trying to connect until the server gets back online.
- If the server gets down, when you put it up again it will send commands starting from the last command given before the shutdown. You do not need to remember where it was.
- You can add new commands to the original file without having to stop the server. The server will read them automatically.
- If some client goes down, the server will remember which command it was executing and it will re-schedule it for later.
- It will store every detail of the operations in a log file.
- It shows real time statistics about the operation of each client

You can choose which port to use. Defaults to 46001. Only the Online clients are shown in the running stats.

USAGE ./dnmap_server -f commands.txt (start dnmap server)

EXAMPLE (see dnmap)

intrace

DESCRIPTION **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. It could be useful for network reconnaissance and firewall bypassing. The difference between **traceroute** and **InTrace** is that **InTrace** will make use of an existing TCP connection, and piggyback its packets on this connection, effectively bypassing any firewall rules that block them, and quite often giving you more internal information than you expected.

USAGE intrace [options] <url>

EXAMPLE ./intrace --h www.freescale.com (Locally initiated TCP connection)

EXAMPLE ./intrace -i eth0 -h 217.17.34.18 (Remotely initiated TCP connection)

EXAMPLE ./intrace -h paypal.com -p 80 (instead of port 80, you can use any other port such as 21 for FTP, or 22 for SSH)

netmask

DESCRIPTION **Netmask** is a netmask generation and conversion program. It accepts and produces a variety of common network address and netmask formats. Not only can it convert address and netmask notations, but it will optimize the masks to generate the smallest list of rules. This is very handy if you've ever configured a firewall or router and some nasty network administrator before you decided that base 10 numbers were good places to start and end groups of machines.

USAGE netmask [options] spec [spec...]

OPTIONS http://www.linuxcertif.com/man/1/netmask/

EXAMPLE netmask aldeid.com
EXAMPLE netmask -s aldeid.com
EXAMPLE netmask -s 192.168.100.1:192.168.100.20
EXAMPLE netmask 192.168.100.1:192.168.100.20

trace6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

trace6 - a basic but very fast traceroute6 program. If no port is specified, ICMP6 Ping requests are used, otherwise TCP SYN packets to the specified port.

USAGE trace6 [-abdt] [-s src6] interface targetaddress [port]

OPTIONS

-a insert a hop-by-hop header with router alert option.
-b instead of an ICMP6 Ping, use TooBig (you will not see the target)
-d resolves the IPv6 addresses to DNS.
-t enables tunnel detection
-s src6 specifies the source IPv6 address

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

[08] INFORMATION GATHERING - SERVICE FINGERPRINTING

- dnmap-client
- dnmap-server
- implementation6
- implementation6d
- ncat
- sslscan
- sslyze
- tlssled

dnmap

DESCRIPTION **dnmap** is a framework to distribute nmap scans among several clients. It reads an already created file with nmap commands and send those commands to each client connected to it.

The framework use a client/server architecture. The server knows what to do and the clients do it. All the logic and statistics are managed in the server. Nmap output is stored on both server and client.

Usually you would want this if you have to scan a large group of hosts and you have several different internet connections (or friends that want to help you).

- Clients can be run on any computer on Internet. Do not have to be on a local cluster or anything.
- It uses the TLS protocol for encryption.

BASIC USAGE

1. Put some **nmap** commands on a file like **commands.txt**
2. ./dnmap_server -f commands.txt (Start the dnmap_server)
3. ./dnmap_client -s <server-ip> -a <alias> (Start any number of clients)

dnmap-client

DESCRIPTION

- If the server gets down, it keeps connecting to it until it gets up again.
- Strip strange characters from the command sent by the server. Tries to avoid command injection vulns.
- It only executes the nmap command. It deletes the command send by the server and changes it by the known and trusted nmap binary on the system.
- You can select an alias for your user.
- You can change which port the client connects to.
- If the command sent by the server does not have a -oA option, the client add it anyway to the command, so it will always have a local copy of the output.

USAGE ./dnmap_client -s <server-ip> -a <alias> (start any number of clients)

EXAMPLE (see dnmap)

dnmap-server

DESCRIPTION

- If the server gets down, clients continue trying to connect until the server gets back online.
- If the server gets down, when you put it up again it will send commands starting from the last command given before the shutdown. You do not need to remember where it was.
- You can add new commands to the original file without having to stop the server. The server will read them automatically.
- If some client goes down, the server will remember which command it was executing and it will re-schedule it for later.
- It will store every detail of the operations in a log file.
- It shows real time statistics about the operation of each client

You can choose which port to use. Defaults to 46001. Only the Online clients are shown in the running stats.

USAGE ./dnmap_server -f commands.txt (start dnmap server)

EXAMPLE (see dnmap)

implementation6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

implementation6 - tests various IPv6 specific options for their implementations. This can also be used to test firewalls, check what it passes. A sniffer on the other side of the firewall or running **implementation6d** shows you what got through. Performs some ipv6 implementation checks, can be used to test some firewall features too. Takes approx. 2 minutes to complete.

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbor solitications which are sent to a non-existing mac, and are therefore very easy to detect). If you dont want this, change the code.

USAGE implementation6 [-p] [-s sourceip6] interface destination [test-case-number]

OPTIONS https://github.com/mmoya/thc-ipv6/blob/master/implementation6.c

EXAMPLE n/a

implementation6d

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

implementation6d - identifies test packets by the **implementation6** tool, useful to check what packets passed a firewall

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

USAGE **implementation6d** interface

EXAMPLE n/a

ncat

DESCRIPTION **ncat** is a general-purpose command-line tool for reading, writing, redirecting, and encrypting data across a network. It aims to be your network Swiss Army knife, handling a wide variety of security testing and administration tasks.

Ncat can:

- Act as a simple TCP/UDP/SCTP/SSL client for interacting with web/telnet/mail/TCP/IP servers and services
- Act as a simple TCP/UDP/SCTP/SSL server for offering services to clients, or simply to understand what existing clients are up to by capturing every byte they send.
- Redirect or proxy TCP/UDP/SCTP traffic to other ports or hosts.
- Encrypt communication with SSL, and transport it over IPv4 or IPv6.
- Act as a network gateway for execution of system commands, with I/O redirected to the network.
- Act as a connection broker, allowing two (or far more) clients to connect to each other through a third (brokering) server.

USAGE ncat [options] <url>

EXAMPLE ncat -C mail.example.com 25 (sending email to an SMTP server. Read manual for further steps)

EXAMPLE ncat -l localhost 143 --sh-exec "ncat --ssl imap.example.com 993“ (connecting to an IMPA server that requires SSL . Read manual for further steps)

sslscan

DESCRIPTION **SSLScan** is a fast SSL port scanner. It 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.

USAGE 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)
--no-failed List only accepted ciphers (default is to listing all ciphers)
--ssl2 Only check SSLv2 ciphers
--ssl3 Only check SSLv3 ciphers
--tls1 Only check TLSv1 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.--starttlsIf a STARTTLS is required to kick an SMTP service into action.
--http Test a HTTP connection.
--bugs Enable SSL implementation bug workarounds.
--xml=<file> Output results to an XML file.
--version Display the program version.
--help Display the help text you are now reading.

EXAMPLE sslscan 209.85.146.17

sslyze

DESCRIPTION Fast and full-featured SSL scanner. **SSLyze** is a Python tool that can analyze the SSL configuration of a server by connecting to it. It is designed to be fast and comprehensive, and should help organizations and testers identify misconfigurations affecting their SSL servers.

More info: https://github.com/iSECPartners/sslyze/wiki

Key features include:

- *SSL 2.0/3.0 and TLS 1.0/1.1/1.2 compatibility*
- *Performance testing: session resumption and TLS tickets support*
- *Security testing: weak cipher suites, insecure renegation, CRIME and THC-SSL DOS attacks*
- *Server certificate validation*
- *Support for StartTLS with SMTP and XMPP, and traffic tunneling through an HTTPS proxy*
- *Client certificate support for servers performing mutual authentication*
- *Scan results can be written to an XML file for further processing*

USAGE python sslyze.py [options] www.target1.com www.target2.com:443

EXAMPLE python sslyze.py --regular www.isecpartners.com:443 www.google.com

tlssled

DESCRIPTION **TLSSLed** is a Linux shell script whose purpose is to evaluate the security of a target SSL/TLS (HTTPS) web server implementation. It is based on **sslscan**, a thorough SSL/TLS scanner that is based on the openssl library, and on the "openssl s_client" command line tool. The current tests include checking if the target supports the SSLv2 protocol, the NULL cipher, weak ciphers based on their key length (40 or 56 bits), the availability of strong ciphers (like AES), if the digital certificate is MD5 signed, and the current SSL/TLS renegotiation capabilities.

USAGE **TLSSLed** <url> <port>

EXAMPLE ./TLSSLed.sh www.owasp.org 443

[09] INFORMATION GATHERING - SMB ANALYSIS

- accheck
- nbtscan
- nmap

accheck

DESCRIPTION no info

USAGE no info

EXAMPLE no info

nbtscan

DESCRIPTION **Nbtscan** is a program for scanning IP networks for NetBIOS name information.

It sends Net-BIOS status query to each address in 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 (such as Ethernet).

USAGE nbtscan [-v] [-d] [-e] [-l] [-t timeout] [-b bandwidth] [-r] [-q] [-s separator] [-m retransmits] (-f filename)|(<scan_range>)

EXAMPLE nbtscan 10.1.1.2

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

[10] INFORMATION GATHERING - SMTP ANALYSIS

- nmap
- smtp-user-enum
- swalks

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

smtp-user-enum

DESCRIPTION **smtp-user-enum** is a tool for enumerating OS-level user accounts on Solaris via the SMTP service (sendmail). Enumeration is performed by inspecting the responses to VRFY, EXPN and RCPT TO commands.

smtp-user-enum simply needs to be passed a list of users and at least one target running an SMTP service.

USAGE smtp-user-enum.pl [options] (-u username|-U file-of-usernames) (-t host|-T file-of-targets)

EXAMPLE smtp-user-enum.pl -M **VRFY** -U users.txt -t 10.0.0.1
EXAMPLE smtp-user-enum.pl -M **EXPN** -U users.txt -t 10.0.0.1
EXAMPLE smtp-user-enum.pl -M **RCPT** -U users.txt -t 10.0.0.1
EXAMPLE ./smtp-user-enum.pl -D example.com -M **RCPT** -U users.txt -t 10.0.0.1

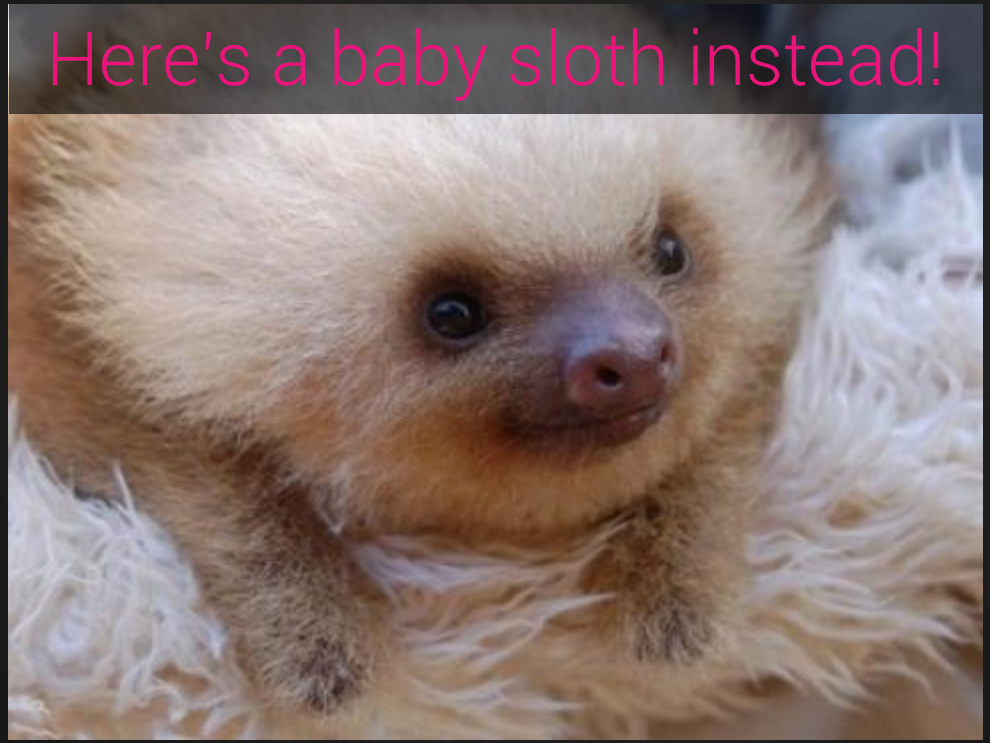
```
$ head users.txt
root
Bin
daemon
adm
lp
sync
shutdown
halt
mail
news
```

swalks

DESCRIPTION no info

USAGE no info

EXAMPLE no info



[11] INFORMATION GATHERING - SNMP ANALYSIS

- braa
- cisco-auditing-tool
- cisco-torch
- copy-router-config
- merge-router-config
- nmap
- onesixtyone

braa

DESCRIPTION **Braa** is a tool for making SNMP queries. It is able to query hundreds or thousands of hosts simultaneously, while being completely single-threaded. It does not need any SNMP libraries, as it is equipped with its own SNMP engine. However, it's good to have a complete SNMP package including "snmptranslate" installed somewhere, because for speed reasons, there is no ASN.1 parser in **Braa**, and all the SNMP OIDs need to be specified numerically.

USAGE braa [-2] [-v] [-t <s>] [-f <file>] [-a <time>] [-r <retries>] [-d <delay>] [querylist1] [querylist2] ...

EXAMPLE braa 10.253.101.1-10.253.101.50:.1.3.6.1.2.1.1.6.0 (query 50 hosts; specify a host range instead a single host in the query list specification)

cisco-auditing-tool

DESCRIPTION **Cisco Auditing Tool** - Perl script which scans cisco routers for common vulnerabilities. Checks for default passwords, easily guessable community names, and the IOS history bug. Includes support for plugins and scanning multiple hosts.

USAGE ./CAT [options]

OPTIONS

-h hostname (for scanning single hosts)
-f hostfile (for scanning multiple hosts)
-p port # (default port is 23)
-w wordlist (wordlist for community name guessing)
-a passlist (wordlist for password guessing)
-i [ioshist] (Check for IOS History bug)
-l logfile (file to log to, default screen)
-q quiet mode (no screen output)

EXAMPLE ./CAT -h 192.168.1.100 -w wordlist -a passwords -i

EXAMPLE ./CAT -h 192.168.1.22 -a lists/passwords -w lists/community (Audit Cisco Telnet Password & SNMP Community String)

cisco-torch

DESCRIPTION **Cisco Torch** was designed as a mass scanning, fingerprinting, and exploitation tool. **Cisco Torch** is unlike other tools in that it utilises multiple threads, (forking techniques), to launch scanning processes. It also uses several methods to simultaneously carry out application layer fingerprinting. **Cisco Torch** can be used for launching dictionary based password attacks against the services and discovering hosts running the following services: Telnet, SSH, Web, NTP, SNMP.

USAGE ./cisco-torch.pl <options> <IP,hostname,network>

USAGE ./cisco-torch.pl <options> -F <hostlist>

OPTIONS check http://www.vulnerabilityassessment.co.uk/torch.htm

EXAMPLE ./cisco-torch.pl -A 10.10.0.0/16

EXAMPLE ./cisco-torch.pl -s -b -F sshtocheck.txt

EXAMPLE ./cisco-torch.pl -w -z 10.10.0.0/16

EXAMPLE ./cisco-torch.pl -j -b -g -F tftptocheck.txt

copy-router-config

DESCRIPTION This tool is good for copying a Cisco router's running configuration with SNMP to a TFTP server, if we have the RW community string. This can be discovered for example with the Cisco Auditing Tool.

USAGE ./copy-router-config.pl [config] <router IP> <server IP> <community>

EXAMPLE ./copy-router-config.pl 192.168.1.1 192.168.0.200 public
EXAMPLE root@bt:/pentest/cisco/copy-router-config# ./merge-router-config.pl 192.168.80.137 192.168.80.128 private
EXAMPLE ./copy-router-config.pl 192.168.1.1 192.168.1.5 datest

merge-router-config

DESCRIPTION The **merge-router-config** menu item allows you to make changes to a Cisco router configuration file and merge those changes to a Cisco router. You should be extremely careful with this script as it will make changes to the target Cisco router.

USAGE ./merge-copy-config.pl <router-ip> <tftp-serverip> <community>

EXAMPLE ./merge-router-config.pl 192.168.1.22 192.168.1.88 private

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

onesixtyone

DESCRIPTION **onesixtyone** takes advantage of the fact that SNMP is a connectionless protocol and sends all SNMP requests as fast as it can. Then the scanner waits for responses to come back and logs them, in a fashion similar to **Nmap** ping sweeps. By default **onesixtyone** waits for 10 milliseconds between sending packets, which is adequate for 100MBs switched networks. The user can adjust this value via the -w command line option. If set to 0, the scanner will send packets as fast as the kernel would accept them, which may lead to packet drop.

USAGE onesixtyone [options] <host> <community>

OPTIONS

-c <communityfile> file with community names to try

-i <inputfile> file with target hosts

-o <outputfile> output log

-d debug mode, use twice for more information

-w <n> wait n milliseconds (1/1000 of a second) between sending packets (default 10)

-q quiet mode, do not print log to stdout, use with –l

EXAMPLE onesixtyone 192.168.100.51

[12] INFORMATION GATHERING - SSL ANALYSIS

- sslcaudit
- sslddump
- sslh
- sslscan
- sslsniff
- sslstrip
- sslyze
- stunnel4
- tlssled

sslcaudit

DESCRIPTION The goal of sslcaudit project is to develop a utility to automate testing SSL/TLS clients for resistance against MITM attacks. It might be useful for testing a thick client, a mobile application, an appliance, pretty much anything communicating over SSL/TLS over TCP.

Full documentation at: http://www.gremwell.com/sslcaudit_files/doc/sslcaudit-user-guide-1.0.pdf

USAGE sslcaudit [Options]

EXAMPLE ./sslcaudit

EXAMPLE ./sslcaudit --server 62.213.200.252:443

EXAMPLE ./sslcaudit --server 62.213.200.252:443 \
--user-cert test/certs/www.example.com-cert.pem
--user-key test/certs/www.example.com-key.pem

ssldump

DESCRIPTION **Ssldump** is an network protocol analyzer specially for SSLv3/TLS. The main purpose of this tool is to identify TCP connections on the selected network interface and interpret them as SSLv3/TLS traffic. It decodes SSLv3/TLS traffic records and uses text form to display them. It can also decrypt the connections and display the application data traffic in some situation. Unlike **tcpdump** this tool needs to see both sides of data transmission so there may be some trouble using it with network taps.

More info: http://www.rtfm.com/ssldump/Ssldump.html

USAGE ssldump [-vtaTnsAxXhHVNdq] [-r dumpfile] [-i interface]

[-k keyfile] [-p password] [expression]

EXAMPLE ssldump -i eth0 port 443 (listen to traffic on interface *eth0* port *443*)

EXAMPLE ssldump -i le0 port 443 and host romeo (ssldump -i le0 port 443 and host romeo)

EXAMPLE ssldump -Ad -k ~/server.pem -p foobar -i le0 host romeo (decrypt traffic to to host romeo server.pem and the password foobar)

EXAMPLE ssldump -nr /var/tmp/www-ssl-client.cap (displays all of the SSL record messages found in the **tcpdump** capture file named **www-ssl-client.cap**)

sslh

DESCRIPTION **sslh** - ssl/ssh multiplexer. **sslh** accepts connections on specified ports, and forwards them further based on tests performed on the first data packet sent by the remote client.

Probes for HTTP, SSL, SSH, OpenVPN, tinc, XMPP are implemented, and any other protocol that can be tested using a regular expression, can be recognised. A typical use case is to allow serving several services on port 443 (e.g. to connect to ssh from inside a corporate firewall, which almost never block port 443) while still serving HTTPS on that port.

Hence **sslh** acts as a protocol demultiplexer, or a switchboard. Its name comes from its original function to serve SSH and HTTPS on the same port.

USAGE sslh [-t num] [-p listening address] [-l target address for SSL] [-s target address for SSH] [-u username] [-P pidfile] [-v] [-i] [-V] [-f]

OPTIONS http://rpm.pbone.net/index.php3/stat/45/idpl/20655622/numer/8/nazwa/sslh

EXAMPLE n/a configure the script and start/stop/restart script

sslscan

DESCRIPTION **SSLScan** is a fast SSL port scanner. It 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.

USAGE 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)
--no-failed List only accepted ciphers (default is to listing all ciphers)
--ssl2 Only check SSLv2 ciphers
--ssl3 Only check SSLv3 ciphers
--tls1 Only check TLSv1 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.--starttlsIf a STARTTLS is required to kick an SMTP service into action.
--http Test a HTTP connection.
--bugs Enable SSL implementation bug workarounds.
--xml=<file> Output results to an XML file.
--version Display the program version.
--help Display the help text you are now reading.

EXAMPLE sslscan 209.85.146.17

sslsniff

DESCRIPTION It is designed to MITM all SSL connections on a LAN, and dynamically generates certificates for the domains that are being accessed on the fly. The new certificates are constructed in a certificate chain that is signed by any certificate that you provide.
More info: https://github.com/moxie0/sslsniff

USAGE sslsniff -a -c <path/to/your/certificate> -f ios -h <httpPort> -s <sslPort> -w iphone.log
USAGE ./sslsniff -t -s <$listenPort> -w <$logFile> -m IPSCACLASEA1.crt \ -c <$certDir>

EXAMPLE

Assuming we want to intercept SSL traffic from 172.17.10.36, we need to trick that host into thinking that we're the router. Using arpspoof, we can convince the target that the router's MAC address is our MAC address.

arpspoof -i eth0 -t 172.17.10.36 172.17.8.1

or

arp-sk -r -S 172.17.8.1 -D 172.17.10.36

At this point, any SSL traffic should get proxied by sslsniff and logged to a file.

First, arpspoof convinces a host that our MAC address is the router's MAC address, and the target begins to send us all its network traffic. The kernel forwards everything along except for traffic destined to port 443, which it redirects to $listenPort (10000, for example).
At this point, sslsniff receives the client connection, makes a connection to the real SSL site, and looks at the information in the server's certificate. sslsniff then generates a new certificate with an identical Distinguished Name and signs it with the end-entity certificate in $certificateFile. sslsniff uses the generated certificate chain to do a SSL handshake with the client and proxy data between both hosts (while logging it, of course).

sslstrip

DESCRIPTION **sslstrip** provides a demonstration of the HTTPS stripping attacks. It will transparently hijack HTTP traffic on a network, watch for HTTPS links and redirects, then map those links into either look-alike HTTP links or homograph-similar HTTPS links. It also supports modes for supplying a favicon which looks like a lock icon, selective logging, and session denial. First, **arpspoof** convinces a host that our MAC address is the router's MAC address, and the target begins to send us all its network traffic. The kernel forwards everything along except for traffic destined to port 80, which it redirects to $listenPort (10000, for example). At this point, **sslstrip** receives the traffic and does its magic.

USAGE sslstrip.py -l <listenPort>

EXAMPLE

Flip your machine into forwarding mode.

```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Setup iptables to redirect HTTP traffic to sslstrip.

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port <listenPort>
```

Run sslstrip.

```
sslstrip.py -l <listenPort>
```

Run arpspoof to convince a network they should send their traffic to you.

```
arpspoof -i <interface> -t <targetIP> <gatewayIP>
```

sslyze

DESCRIPTION Fast and full-featured SSL scanner. **SSLyze** is a Python tool that can analyze the SSL configuration of a server by connecting to it. It is designed to be fast and comprehensive, and should help organizations and testers identify misconfigurations affecting their SSL servers.

More info: https://github.com/iSECPartners/sslyze/wiki

Key features include:

- *SSL 2.0/3.0 and TLS 1.0/1.1/1.2 compatibility*
- *Performance testing: session resumption and TLS tickets support*
- *Security testing: weak cipher suites, insecure renegation, CRIME and THC-SSL DOS attacks*
- *Server certificate validation*
- *Support for StartTLS with SMTP and XMPP, and traffic tunneling through an HTTPS proxy*
- *Client certificate support for servers performing mutual authentication*
- *Scan results can be written to an XML file for further processing*

USAGE python sslyze.py [options] www.target1.com www.target2.com:443

EXAMPLE python sslyze.py --regular www.isecpartners.com:443 www.google.com

stunnel4

DESCRIPTION The **stunnel** program is designed to work as SSL encryption wrapper between remote clients and local (inetd-startable) or remote servers. The concept is that having non-SSL aware daemons running on your system you can easily set them up to communicate with clients over secure SSL channels. **stunnel** can be used to add SSL functionality to commonly used Inetd daemons like POP-2, POP-3, and IMAP servers, to standalone daemons like NNTP, SMTP and HTTP, and in tunneling PPP over network sockets without changes to the source code.

USAGE stunnel [<filename>] | -fdn | -help | -version | -sockets

OPTIONS: http://man.he.net/man8/stunnel4

EXAMPLE

In order to provide SSL encapsulation to your local imapd service, use

```
[imapd]
accept = 993
exec = /usr/sbin/imapd
execargs = imapd
```

If you want to provide tunneling to your pppd daemon on port 2020, use

```
[vpn]
accept = 2020
exec = /usr/sbin/pppd
execargs = pppd local
pty = yes
```

tlssled

DESCRIPTION **TLSSLed** is a Linux shell script whose purpose is to evaluate the security of a target SSL/TLS (HTTPS) web server implementation. It is based on **sslscan**, a thorough SSL/TLS scanner that is based on the openssl library, and on the "openssl s_client" command line tool. The current tests include checking if the target supports the SSLv2 protocol, the NULL cipher, weak ciphers based on their key length (40 or 56 bits), the availability of strong ciphers (like AES), if the digital certificate is MD5 signed, and the current SSL/TLS renegotiation capabilities.

USAGE **TLSSLed** <url> <port>

EXAMPLE ./TLSSLed.sh www.owasp.org 443

[13] INFORMATION GATHERING - TELEPHONY ANALYSIS

- ace

ace

DESCRIPTION **ACE** (Automated Corporate Enumerator) is a simple yet powerful VoIP Corporate Directory enumeration tool that mimics the behavior of an IP Phone in order to download the name and extension entries that a given phone can display on its screen interface. **ACE** can be used in one of two ways. First, it can auto-discover the TFTP Server IP Address via DHCP, or (second) the user can specify the TFTP Server IP address as a command line parameter of the tool. In either case, you must supply the MAC Address of the IP Phone with the -m option in order for the tool to correctly download the configuration file via TFTP.

USAGE ace [-i interface] [-m mac address] [-t tftp server ip address | -c cdp mode | -v voice vlan id | -r vlan interface | -d verbose mode]

OPTIONS http://ucsniff.sourceforge.net/ace.html

EXAMPLE ace -i eth0 -v 96 -m 00:1E:F7:28:9C:8E (Mode to specify the Voice VLAN ID)
EXAMPLE ace -i eth0 -c 0 -m 00:1E:F7:28:9C:8E (Mode to auto-discover voice vlan ID in the listening mode for CDP
EXAMPLE ace -i eth0 -c 1 -m 00:1E:F7:28:9C:8E (Mode to auto-discover voice vlan ID in the spoofing mode for CDP)

TIP To view your MAC address root@bt:~# macchanger -s eth0

[14] INFORMATION GATHERING - TRAFFIC ANALYSIS

- cdpsnarf
- intrace
- irpas-ass
- irpass-cdp
- p0f
- tcpflow
- wireshark

cdpsnarf

DESCRIPTION **CDPSnarf** if a network sniffer exclusively written to extract information from CDP packets. It provides all the information a “show cdp neighbors detail” command would return on a Cisco router and even more.
Features: Time intervals between CDP advertisements, Source MAC address, CDP Version, TTL, Checksum, Device ID, Software version, Platform, Addresses, Port ID, Capabilities, Duplex, Save packets in PCAP dump file format, Read packets from PCAP dump files, Debugging information (using the "-d" flag), Tested with IPv4 and IPv6

USAGE cdpsnarf -i <device>

OPTIONS cdpsnarf -h

EXAMPLE ./cdpsnarf eth2

intrace

DESCRIPTION **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. It could be useful for network reconnaissance and firewall bypassing. The difference between **traceroute** and **InTrace** is that **InTrace** will make use of an existing TCP connection, and piggyback its packets on this connection, effectively bypassing any firewall rules that block them, and quite often giving you more internal information than you expected.

USAGE intrace [options] <url>

EXAMPLE ./intrace --h www.freescale.com (Locally initiated TCP connection)

EXAMPLE ./intrace -i eth0 -h 217.17.34.18 (Remotely initiated TCP connection)

EXAMPLE ./intrace -h paypal.com -p 80 (instead of port 80, you can use any other port such as 21 for FTP, or 22 for SSH)

irpas-ass

DESCRIPTION **Internet Router Protocol Attack Suite** - a suite of tools designed to abuse inherent design insecurity in routers and routing protocols. **Autonomous System Scanner – ASS** is a protocol-aware scanner used to query routers for AS information and a valuable reconnaissance technique for attackers looking for insecure boundaries" between networks. Because routing protocols use autonomous systems to distinguish between various routing "domains" and various ways to communicate, you need something which works like a TCP port scanner but knows more then one protocol.

USAGE ./ass [-v[v[v]]] -i <interface> [-p] [-c] [-A] [-M] [-P IER12]
-a <autonomous system start> -b <autonomous system stop>
[-S <spoofed source IP>] [-D <destination ip>]
[-T <packets per delay>]

OPTIONS http://www.phenoelit.org/irpas/docu.html

EXAMPLE ./ass -i eth0 (Passive Mode)

EXAMPLE ./ass -i eth0 –A (Active Mode)

irpass-cdp

DESCRIPTION **Internet Router Protocol Attack Suite** - a suite of tools designed to abuse inherent design insecurity in routers and routing protocols. This program is for sending CDP (Cisco router Discovery Protocol) messages to the wire.

The CDP tool can be used in two different modes:
1. The flood mode is used to send garbage CDP messages to the wire, which has different effects to the routers depending on their IOS version.
2. The second mode for CDP is spoofing. You can enable this mode with the command line option -m 1. It has no actuall use for attacking router and is mostly targeted fro social engineering or just to confuse the local administrator. It is used to send out 100% valid CDP infromation packets which look like generated by other Cisco routers. Here, you can specify any part of a CDP message yourself.

USAGE ./cdp [depends on the mode; see documenation]

OPTIONS http://www.phenoelit.org/irpas/docu.html

EXAMPLE ./cdp -i eth0 -n 10000 -l 1480 –r (flood mode)

EXAMPLE ./cdp -v -i eth0 -m 1 -D 'Hacker' -P 'Ethernet0' -C RI \ -L 'Intel' -S "`uname -a`" -F '255.255.255.255‘ (spoofing)

TIP if you want to flood the routers completly, start two processes of cdp with different sizes. One of them running on full size (1480) to fill up the major part of the memory and another to fill up the rest with a length of 10 octets.

p0f

DESCRIPTION **p0f** uses a fingerprinting technique based on analyzing the structure of a TCP/IP packet to determine the operating system and other configuration properties of a remote host. The process is completely passive and does not generate any suspicious network traffic. The other host has to either:

connect to your network - either spontaneously or in an induced manner, for example when trying to establish a ftp data stream, returning a bounced mail, performing auth lookup, using IRC DCC, external html mail image reference and so on, or be contacted by some entity on your network using some standard means (such as a web browsing); it can either accept or refuse the connection.

The method can see thru packet firewalls and does not have the restrictions of an active fingerprinting. The main uses of passive OS fingerprinting are attacker profiling (IDS and honeypots), visitor profiling (content optimization), customer/user profiling (policy enforcement), pen-testing, etc.

USAGE p0f [-f file] [-i device] [-s file] [-o file]
[-w file] [-Q sock [-0]] [-u user] [-FXVNDUKASCMROqtpvdlrx]
[-c size] [-T nn] [-e nn] ['filter rule']

OPTIONS http://www.aldeid.com/wiki/P0f

EXAMPLE p0f -i eth1 –vt (The following command will start p0f)

EXAMPLE p0f -i eth1 –vto output.txt (The output of the ingerprint information can also be directed to a file using the –o option)

tcpflow

DESCRIPTION **tcpflow** is a program that captures data transmitted as part of TCP connections (flows), and stores the data in a way that is convenient for protocol analysis and debugging. Each TCP flow is stored in its own file. Thus, the typical TCP flow will be stored in two files, one for each direction. **tcpflow** can also process stored 'tcpdump' packet flows.

tcpflow stores all captured data in files that have names of the form:
[timestampT]sourceip.sourceport-destip.destport[--VLAN][cNNNN]

USAGE tcpflow [-chpsv] [-b max_bytes] [-d debug_level] [-f max_fds] [-i iface] [-r file] [expression]

OPTIONS http://linux.die.net/man/1/tcpflow

EXAMPLE tcpflow -c -n en1 src or dst host api.example.com
EXAMPLE tcpflow host sundown (To record all packets arriving at or departing from *sundown*)
EXAMPLE tcpflow host helios and \(hot or ace \) (To record traffic between helios and either hot or ace)
EXAMPLE tcpflow host ace and not helios (To record traffic between ace and any host except helios)
EXAMPLE tcpflow net ucb-ether (To record all traffic between local hosts and hosts at Berkeley)
EXAMPLE tcpflow 'gateway snup and (port ftp or ftp-data)' (To record all ftp traffic through internet gateway snup: (note that the expression is quoted to prevent the shell from (mis-)interpreting the parentheses))

wireshark

DESCRIPTION **wireshark** - Interactively dump and analyze network traffic. **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. **Wireshark**'s native capture file format is **libpcap** format, which is also the format used by **tcpdump** and various other tools.

USAGE **wireshark** [**-a** <capture autostop condition>] ... [**-b** <capture ring buffer option>] ... [**-B** <capture buffer size (Win32 only)>] [**-c** <capture packet count>] [**-C** <configuration profile>] [**-D**] [**--display=**<X display to use>] [**-f** <capture filter>] [**-g** <packet number>] [**-h**] [**-H**] [**-i** <capture interface>|-] [**-k**] [**-K** <keytab>] [**-l**] [**-L**] [**-m** <font>] [**-n**] [**-N** <name resolving flags>] [**-o** <preference/recent setting>] ... [**-p**] [**-P** <path setting>] [**-Q**] [**-r** <infile>] [**-R** <read (display) filter>] [**-S**] [**-s** <capture snaplen>] [**-t**ad|a|r|d|dd|e] [**-v**] [**-w** <outfile>] [**-y** <capture link type>] [**-X** <eXtension option>] [**-z** <statistics>] [<infile>]

OPTIONS http://linux.die.net/man/1/wireshark

EXAMPLE n/a; GUI tool

[15] INFORMATION GATHERING - VOIP ANALYSIS

- ace
- enumiax

ace

DESCRIPTION **ACE** (Automated Corporate Enumerator) is a simple yet powerful VoIP Corporate Directory enumeration tool that mimics the behavior of an IP Phone in order to download the name and extension entries that a given phone can display on its screen interface. **ACE** can be used in one of two ways. First, it can auto-discover the TFTP Server IP Address via DHCP, or (second) the user can specify the TFTP Server IP address as a command line parameter of the tool. In either case, you must supply the MAC Address of the IP Phone with the -m option in order for the tool to correctly download the configuration file via TFTP.

USAGE ace [-i interface] [-m mac address] [-t tftp server ip address | -c cdp mode | -v voice vlan id | -r vlan interface | -d verbose mode]

OPTIONS http://ucsniff.sourceforge.net/ace.html

EXAMPLE ace -i eth0 -v 96 -m 00:1E:F7:28:9C:8E (Mode to specify the Voice VLAN ID)
EXAMPLE ace -i eth0 -c 0 -m 00:1E:F7:28:9C:8E (Mode to auto-discover voice vlan ID in the listening mode for CDP
EXAMPLE ace -i eth0 -c 1 -m 00:1E:F7:28:9C:8E (Mode to auto-discover voice vlan ID in the spoofing mode for CDP)

TIP To view your MAC address root@bt:~# macchanger -s eth0

enumiax

DESCRIPTION **enumIAX** is an Inter Asterisk Exchange version 2 (IAX2) protocol username brute-force enumerator. **enumIAX** may operate in two distinct modes; Sequential Username Guessing or Dictionary Attack.

USAGE enumiax <target-ip> [options]

OPTIONS

-d Dictionary attack using file
-i Interval for auto-save (# of operations, default 1000)
-m # Minimum username length (in characters)
-M # Maximum username length (in characters)
-r # Rate-limit calls (in microseconds)
-s Read session state from state file
-v Increase verbosity (repeat for additional verbosity)
-V Print version information and exit
-h Print help/usage information and exit

EXAMPLE enumiax 172.16.1.100 –m 4 –M 8 –v (enumIAX under sequential mode attempting usernames that have four and eight characters)

enumiax 172.16.1.100 –d dict –v (next, use enumIAX under dictionary mode by using the following syntax)

EXAMPLE ./enumiax -v -m3 -M3 192.168.1.104

./enumiax -d dict -v 192.168.1.104

[16] INFORMATION GATHERING - VPN ANALYSIS

- ike-scan

ike-scan

DESCRIPTION **Ike-scan** is a command-line tool that uses the IKE protocol to discover, fingerprint and test IPSec VPN servers.

ike-scan allows you to:

- *Send IKE packets to any number of destination hosts, using a configurable output bandwidth or packet rate. This is useful for VPN detection, when you may need to scan large address spaces.*
- *Construct the outgoing IKE packet in a flexible way. This includes IKE packets which do not comply with the RFC requirements.*
- *Decode and display any returned packets.*
- *Crack aggressive mode pre-shared keys. You can use **ike-scan** to obtain the PSK hash data, and then use **psk-crack** to obtain the key.*

USAGE ike-scan [options] [hosts...]

OPTIONS http://www.nta-monitor.com/wiki/index.php/Ike-scan_User_Guide or ike-scan --help

EXAMPLE ike-scan -M 10.0.0.0/24

EXAMPLE ike-scan --auth=3 10.0.0.0/24 (use the standard transform set with the authentication method set to RSA Signature instead of the default Pre-Shared key for each transform. This method is surprisingly effective at discovering VPN servers that won't respond to the standard transform set.)

EXAMPLE ike-scan -M --trans=5,2,1,2 --showbackoff 10.0.0.1 (backoff fingerprinting)

[17] VULNERABILITY ANALYSIS

- cisco-auditing-tool
- cisco-global-exploiter
- cisco-ocs
- cisco-torch
- yersinia

cisco-auditing-tool

DESCRIPTION **Cisco Auditing Tool** - Perl script which scans cisco routers for common vulnerabilities. Checks for default passwords, easily guessable community names, and the IOS history bug. Includes support for plugins and scanning multiple hosts.

USAGE ./CAT [options]

OPTIONS

-h hostname (for scanning single hosts)
-f hostfile (for scanning multiple hosts)
-p port # (default port is 23)
-w wordlist (wordlist for community name guessing)
-a passlist (wordlist for password guessing)
-i [ioshist] (Check for IOS History bug)
-l logfile (file to log to, default screen)
-q quiet mode (no screen output)

EXAMPLE ./CAT -h 192.168.1.100 -w wordlist -a passwords -i

EXAMPLE ./CAT -h 192.168.1.22 -a lists/passwords -w lists/community (Audit Cisco Telnet Password & SNMP Community String)

cisco-global-exploiter

DESCRIPTION **Cisco Global Exploiter (CGE),** is an advanced, simple and fast security testing tool/ exploit engine, that is able to exploit 14 vulnerabilities in disparate Cisco switches and routers. CGE is command-line driven Perl script which has a simple and easy to use front-end.

USAGE cge.pl <target> <vulnerability number>

OPTIONS (14 vulnerabilities)

[1] - Cisco 677/678 Telnet Buffer Overflow Vulnerability
[2] - Cisco IOS Router Denial of Service Vulnerability
[3] - Cisco IOS HTTP Auth Vulnerability
[4] - Cisco IOS HTTP Configuration Arbitrary Administrative Access Vulnerability
[5] - Cisco Catalyst SSH Protocol Mismatch Denial of Service Vulnerability
[6] - Cisco 675 Web Administration Denial of Service Vulnerability
[7] - Cisco Catalyst 3500 XL Remote Arbitrary Command Vulnerability
[8] - Cisco IOS Software HTTP Request Denial of Service Vulnerability
[9] - Cisco 514 UDP Flood Denial of Service Vulnerability
[10] - CiscoSecure ACS for Windows NT Server Denial of Service Vulnerability
[11] - Cisco Catalyst Memory Leak Vulnerability
[12] - Cisco CatOS CiscoView HTTP Server Buffer Overflow Vulnerability
[13] - 0 Encoding IDS Bypass Vulnerability (UTF)
[14] - Cisco IOS HTTP Denial of Service Vulnerability

EXAMPLE cge.pl 192.168.1.254 3 (exploit the Cisco IOS HTTP Auth Vulnerability and hopefully using the nice link provided we should have basic access to the switch we are attacking, (not enable))

cisco-ocs

DESCRIPTION **cisco-ocs** also known as **cisco-ocs Mass Scanner**. This tool provides a single function which is to scan large ranges of IP's looking for Cisco devices or really any device listening on TCP port 23, attempts to login using telnet with a password of cisco, then passes the enable command to the Cisco router if its able to login via telnet, uses cisco again for the enable password, and finally reports a success if its able to get to the enable prompt using these exact steps. Unfortunately, this is the only function of the tool as you cannot specify a wordlist of passwords to attempt or for that matter you cannot set anything accept for the range of IP addresses to scan.

USAGE ./ocs <range start IP> <range end IP>

EXAMPLE ./ocs 192.168.1.21 192.168.1.23

cisco-torch

DESCRIPTION **Cisco Torch** was designed as a mass scanning, fingerprinting, and exploitation tool. **Cisco Torch** is unlike other tools in that it utilises multiple threads, (forking techniques), to launch scanning processes. It also uses several methods to simultaneously carry out application layer fingerprinting. **Cisco Torch** can be used for launching dictionary based password attacks against the services and discovering hosts running the following services: Telnet, SSH, Web, NTP, SNMP.

USAGE ./cisco-torch.pl <options> <IP,hostname,network>

USAGE ./cisco-torch.pl <options> -F <hostlist>

OPTIONS check http://www.vulnerabilityassessment.co.uk/torch.htm

EXAMPLE ./cisco-torch.pl -A 10.10.0.0/16

EXAMPLE ./cisco-torch.pl -s -b -F sshtocheck.txt

EXAMPLE ./cisco-torch.pl -w -z 10.10.0.0/16

EXAMPLE ./cisco-torch.pl -j -b -g -F tftptocheck.txt

yersinia

DESCRIPTION **Yersinia** is a network tool designed to take advantage of some weaknesses in different network protocols. It pretends to be a solid framework for analyzing and testing the deployed networks and systems. *Attacks for the following network protocols are implemented: Spanning Tree Protocol (STP), Cisco Discovery Protocol (CDP), Dynamic Trunking Protocol (DTP), Dynamic Host Configuration Protocol (DHCP), Hot Standby Router Protocol (HSRP), IEEE 802.1Q, IEEE 802.1X, Inter-Switch Link Protocol (ISL), VLAN Trunking Protocol (VTP)*

USAGE yersinia [-hVID] [-l logfile] protocol [protocol_options]

OPTIONS

-V Program version.

-h This help screen.

-I Interactive mode (ncurses).

-D Daemon mode.

-l logfile Select logfile.

-c conffile Select config file.

protocol Can be one of the following: cdp, dhcp, dot1q, dtp, hsrp, stp, vtp

EXAMPLE yersinia –D (run in Daemon mode)

[18] DATABASE ASSESSMENT

- bbqsql
- dbpwaudit
- hexorbase
- mdb-export
- mdb-parsecsv
- mdb-sql
- mdb-tables
- oscanner
- sidguesser
- sqldict
- sqlmap
- sqlninja
- sqlsus
- tnscmd10g

bbqsql

DESCRIPTION Blind SQL injection can be a pain to exploit. When the available tools work they work well, but when they don't you have to write something custom. This is time-consuming and tedious. **BBQSQL** can help you address those issues.

BBQSQL is a blind SQL injection framework written in Python. It is extremely useful when attacking tricky SQL injection vulnerabilities. **BBQSQL** is also a semi-automatic tool, allowing quite a bit of customization for those hard to trigger SQL injection findings. The tool is built to be database agnostic and is extremely versatile. It also has an intuitive UI to make setting up attacks much easier. Python gevent is also implemented, making **BBQSQL** extremely fast.

USAGE n/a, option selection/configuration

OPTIONS https://github.com/Neohapsis/bbqsql

EXAMPLE n/a

dbpwaudit

DESCRIPTION **DBPwAudit** is a Java tool that allows you to perform online audits of password quality for several database engines. The application design allows for easy adding of additional database drivers by simply copying new JDBC drivers to the jdbc directory. Configuration is performed in two files, the aliases.conf file is used to map drivers to aliases and the rules.conf tells the application how to handle error messages from the scan.

The tool has been tested and known to work with:
- Microsoft SQL Server 2000/2005
- Oracle 8/9/10/11
- IBM DB2 Universal Database
- MySQL

USAGE dbpwaudit -s <server> -d <db> -D <driver> -U <users> -P <passwords> [options]

OPTIONS http://www.edwiget.name/2012/07/auditing-mysql-passwords-with-dbpwaudit/

EXAMPLE ./dbpwaudit.sh -s localhost -d mysql -D MySQL -U ~/mysql-users.txt -P ~/mysql-password.txt (Assuming I have a db server on localhost and a list of mysql usernames saved in my home directory as mysql-users.txt and a list of passwords to try also in my home directory as mysql-password.txt, this command would audit the mysql server)

TIP additional steps are required for this program to work: http://www.edwiget.name/2012/07/auditing-mysql-passwords-with-dbpwaudit/

hexorbase

DESCRIPTION **HexorBase** is a database application designed for administering and auditing multiple database servers simultaneously from a centralized location, it is capable of performing SQL queries and bruteforce attacks against common database servers (MySQL, SQLite, Microsoft SQL Server, Oracle, PostgreSQL). **HexorBase** allows packet routing through proxies or even **Metasploit** pivoting antics to communicate with remotely inaccessible servers which are hidden within local subnets.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

mdb-export

DESCRIPTION **The MDB Tools project** is a effort to document the MDB file format used in Microsoft's Access database package, and to provide a set of tools and applications to make that data available on other platforms. Specifically, MDB Tools includes programs to export schema and data to other databases such as **MySQL, Oracle, Sybase, PostgreSQL,** and others.

mdb-export is a utility program distributed with **MDB Tools**. It produces a CSV (comma separated value) output for the given table. Such output is suitable for importation into databases or spreadsheets.

USAGE mdb-export [-H] [-d <delimiter>] [-R <row delim>] [[-Q] || [-q <quote> [-X <escape>]]] [-I] [-D <format>] [-S] <database> <table>

OPTIONS http://linux.die.net/man/1/mdb-export

EXAMPLE mdb-export -d ';' something.mdb Main > Main.csv

EXAMPLE mdb-export -X \ -D "%Y-%m-%d %H:%M:%S" orsk.mdb Main > Main.csv

mdb-parsecsv

DESCRIPTION The **MDB Tools project** is a effort to document the MDB file format used in Microsoft's Access database package, and to provide a set of tools and applications to make that data available on other platforms.
Specifically, **MDB Tools** includes programs to export schema and data to other databases such as **MySQL, Oracle, Sybase, PostgreSQL,** and others.

mdb-parsecsv is a utility program distributed with **MDB Tools**. **mdb-parsecsv** takes a CSV file representing a database table, and converts it into a C array.
*If the first argument does not exist as a file, **mdb-parsecsv** will look for the same filename with '.txt' appended.*
The file extension is stripped, and the output written to the base name plus a '.c' extension.

USAGE n/a
EXAMPLE n/a

mdb-sql

DESCRIPTION The **MDB Tools project** is a effort to document the MDB file format used in Microsoft's Access database package, and to provide a set of tools and applications to make that data available on other platforms. Specifically, **MDB Tools** includes programs to export schema and data to other databases such as **MySQL, Oracle, Sybase, PostgreSQL**, and others.

mdb-sql - a command line SQL tool that allows one to type sql queries and get results. **mdb-sql** allows querying of an MDB database using a limited SQL subset language.

USAGE mdb-sql [**-HFp**] [**-d** <delimiter>] [**-i** <file>] [**-o** <file>] [<database>]

OPTIONS http://linux.die.net/man/1/mdb-sql

EXAMPLE n/a

mdb-tables

DESCRIPTION **The MDB Tools project** is a effort to document the MDB file format used in Microsoft's Access database package, and to provide a set of tools and applications to make that data available on other platforms. Specifically, **MDB Tools** includes programs to export schema and data to other databases such as **MySQL, Oracle, Sybase, PostgreSQL**, and others.

mdb-tables is a utility program distributed with **MDB Tools**. It produces a list of tables contained within an MDB database in a format suitable for use in shell scripts.

USAGE mdb-tables [**-S**] [**-1** | **-d**<delimiter>] <database>

OPTIONS http://linux.die.net/man/1/mdb-tables

EXAMPLE mdb-tables database.mdb

oscanner

DESCRIPTION **Oscanner** is an Oracle assessment framework developed in Java. It has a plugin-based architecture and comes with a couple of plugins that currently do: Sid Enumeration, Passwords tests (common & dictionary), Enumerate Oracle version, Enumerate account roles, Enumerate account privileges, Enumerate account hashes, Enumerate audit information, Enumerate password policies, Enumerate database links

USAGE OracleScanner -s <ip> -r <repfile> [options]

USAGE oscanner -s <ip> -r <repfile> [options]

OPTIONS

-s <servername>
-f <serverlist>
-P <portnr>
-v be verbose

EXAMPLE oscanner.sh -s 192.168.0.1

sidguesser

DESCRIPTION **sidguesser** guesses sids/instances against an Oracle database according to a predefined dictionary file. The speed is slow (80-100 guesses per second) but it does the job.

USAGE sidguesser -i <ip> -d <dictionary> [options]

OPTIONS

-p <portnr> Use specific port (default 1521)
-r <report> Report to file
-m <mode> findfirst OR findall (default)

EXAMPLE sidguesser -i 192.168.0.223 -d words.txt

sqldict

DESCRIPTION **SQLdict** is a basic single ip brute-force MS SQL Server password utility that can carry out a dictionary attack against a named SQL account.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

sqlmap

DESCRIPTION **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.

USAGE sqlmap.py [options]

OPTIONS https://github.com/sqlmapproject/sqlmap/wiki/Usage

EXAMPLE sqlmap.py -u "http://www.target.com/vuln.php?id=1" -f --banner --dbs –users (Run sqlmap against a single target URL)
EXAMPLE sqlmap.py -g "inurl:\".php?id=1\"" (test and inject on GET parameters based on results of your Google dork)
EXAMPLE sqlmap.py -u "http://www.target.com/vuln.php" --data="id=1" -f --banner --dbs --users (hunt for POST requests)
EXAMPLE sqlmap.py -u "http://192.168.136.131/sqlmap/mysql/basic/get_int.php?id=1" \ --auth-type Basic --auth-cred "testuser:testpass" (HTTP authentication)
EXAMPLE sqlmap.py -l burp.log --scope="(www)?\.target\.(com|net|org)“ (Filtering targets from provided proxy log using regular expression)

sqlninja

DESCRIPTION **Sqlninja** is a tool targeted to exploit SQL Injection vulnerabilities on a web application that uses Microsoft SQL Server as its back-end. Its main goal is to provide a remote access on the vulnerable DB server, even in a very hostile environment. It should be used by penetration testers to help and automate the process of taking over a DB Server when a SQL Injection vulnerability has been discovered.

USAGE configuration and execution happens via sqlninja.conf; see documentation for more details

OPTIONS http://sqlninja.sourceforge.net/sqlninja-howto.html

EXAMPLE n/a

sqlsus

DESCRIPTION **sqlsus** is an open source MySQL injection and takeover tool. Via a command line interface, you can retrieve the database(s) structure, inject your own SQL queries (even complex ones), download files from the web server, crawl the website for writable directories, upload and control a backdoor, clone the database(s), and much more... Whenever relevant, sqlsus will mimic a MySQL console output.
More info: http://sqlsus.sourceforge.net/

USAGE Generate a configuration file with **sqlsus --genconf my.cfg**, read the comments and adapt it to reflect your target. Launch **sqlsus**, with your configuration as a parameter sqlsus my.cfg, you will get a shell.

OPTIONS http://sqlsus.sourceforge.net/documentation.html

EXAMPLE n/a, check documentation

tnscmd10g

DESCRIPTION **tnscmd** can be used to speak, on a very simple level, with Oracle's TNS listener. The TNS listener (aka **tnslsnr**) is the network interface between a database client and the database server. **tnslsnr** listens on port 1521/tcp.

The **tnslnsr** keeps a spartan log of activity -- spartan in that it doesn't log a whole lot of useful information. For instance, it does not log the IP address of TNS sessions. If you initiate a TCP session to the **tnslsnr** port, you won't make much headway; it won't provide a banner and will probably disconnect if you type something. Don't worry; this is what **tnscmd** is for. **tnscmd** simply talks to the **tnslsnr** process. **tnslsnr** will respond to certain commands such as ping (an application-level no-op), version (dumps version information about Oracle), status (dumps status about the listener and database instances), and services (dumps info about the running services.) Commands are apparently case-insensitive.

More info: http://www.jammed.com/~jwa/hacks/security/tnscmd/tnscmd-doc.html

USAGE tnscmd [command] -h hostname where 'command' is something like ping, version, status, etc. (default is ping)

EXAMPLE tnscmd -h oraclebox.example.com -p 1521 –indent (ping this host to see if it is actually running tnslsnr)

EXAMPLE tnscmd [badcommand] -h oraclebox.example.com ("Bad" TNS packets can crash the listener, regardless of whether or not the DBA has set a password. badcommand can be any one of: trc_file trc_level use_plugandplay trc_directory snmp_visible log_file log_status log_directory)

[19] FUZZING TOOLS

- bed
- fuzz_ip6
- ohrwurm
- powerfuzzer
- sfuzz
- siparmyknife
- spike-generic_chunked
- spike-generic_listen_tcp
- spike-generic_send_tcp
- spike-generic_listen_upd

bed

DESCRIPTION **BED (aka Bruteforce Exploit Detector)** is a plain-text protocol fuzzer that checks software for common vulnerabilities like buffer overflows, format string bugs, integer overflows, etc. The tool currently supports following protocols: finger, ftp, http, imap, irc, lpd, pjl, pop, smtp

USAGE ./bed.pl -s <plugin> [options]

OPTIONS

-s <plugin> Plugin to use (mandatory).
Valid plugins are: FTP/SMTP/POP/HTTP/IRC/IMAP/PJL/LPD/FINGER/SOCKS4/SOCKS5
Use "./bed.pl -s <plugin>" to obtain the parameters you need for the plugin.

-t <target> Host to check (default: localhost)

-p <port> Port to connect to (default: standard port)

-o <timeout> seconds to wait after each test (default: 2 seconds)

EXAMPLE **./bed.pl -s HTTP -t 192.168.100.16 -p 80**

fuzz_ip6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fuzz_ip6 - fuzzes an icmp6 packet.

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbor solitications which are sent to a non-existing mac, and are therefore very easy to detect). If you dont want this, change the code.

USAGE fuzz_ip6 [-x] [-t number | -T number] [-p number] [-IFSDHRJ] [-X|-1|-2|-3|-4|-5|-6|-7|-8|-9|-0 port] interface unicast-or-multicast-address [address-in-data-pkt]

OPTIONS https://github.com/mmoya/thc-ipv6/blob/master/fuzz_ip6.c

EXAMPLE n/a

ohrwurm

DESCRIPTION **ohrwurm** is a small and simple RTP fuzzer, it has been tested it on a small number of SIP phones, none of them withstood the fuzzing.

Features:

- reads SIP messages to get information of the RTP port numbers
- reading SIP can be omitted by providing the RTP port numbers, so that any RTP traffic can be fuzzed
- RTCP traffic can be suppressed to avoid that codecs learn about the “noisy line”
- special care is taken to break RTP handling itself
- the RTP payload is fuzzed with a constant BER
- the BER is configurable
- requires arpspoof from dsniff to do the MITM attack
- requires both phones to be in a switched LAN (GW operation only works partially)

USAGE n/a

EXAMPLE n/a

powerfuzzer

DESCRIPTION **Powerfuzzer** is a highly automated and fully customizable web fuzzer (HTTP protocol based application fuzzer) based on many other Open Source fuzzers available and information gathered from numerous security resources and websites. It was designed to be user friendly, modern, effective and working.
More info: http://www.powerfuzzer.com/#news

Currently, it is capable of identifying these problems:
- Cross Site Scripting (XSS)
- Injections (SQL, LDAP, code, commands, and XPATH)
- CRLF
- HTTP 500 statuses (usually indicative of a possible misconfiguration/security flaw incl. buffer overflow)

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

sfuzz

DESCRIPTION **simple fuzzer** is exactly what it sounds like - a simple fuzzer. don't mistake simple with a lack of fuzz capability. this fuzzer has two network modes of operation, an output mode for developing command line fuzzing scripts, as well as taking fuzzing strings from literals and building strings from sequences.

simple fuzz is built to fill a need - the need for a quickly configurable black box testing utility that doesn't require intimate knowledge of the inner workings of C or require specialized software rigs. the aim is to just provide a simple interface, clear inputs/outputs, and reusability.

USAGE sfuzz [output modes] [logging] [config file] [network host] [variables] [misc]
USAGE sfuzz –h (for help) or read documentation by following the link below

OPTIONS https://github.com/orgcandman/Simple-Fuzzer

EXAMPLE sfuzz -T -f /tmp/myfirst.cfg -S foo.com -p 80

siparmyknife

DESCRIPTION **SIP Army Knife** is a fuzzer that searches for cross site scripting, SQL injection, log injection, format strings, buffer overflows, and more. **sipsak** − a utility for various tests on sip servers and user agents.

USAGE **sipsak [-dFGhiILnNMRSTUVvwz] [-a** *PASSWORD* **] [-b** *NUMBER* **] [-c** *SIPURI* **] [-C** *SIPURI* **] [-D** *NUMBER* **] [-e** *NUMBER* **] [-E** *STRING* **] [-f** *FILE* **] [-g** *STRING* **] [-H***HOSTNAME* **] [-l** *PORT* **] [-m** *NUMBER* **] [-o** *NUMBER* **] [-p** *HOSTNAME* **] [-P** *NUMBER* **] [-q** *REGEXP* **] [-r** *PORT* **] [-t** *NUMBER* **] [-u** *STRING* **] [-W** *NUMBER* **] [-x***NUMBER* **] -s** *SIPURI*

OPTIONS http://sipsak.org/man-page.html

EXAMPLE sipsak -vv -s sip:nobody@foo.bar (Send an OPTIONS request to nobody@foo.bar and display received replies)

EXAMPLE sipsak -T -s sip nobody@foo.bar (Trace the SIP path to nobody@foo.bar)

EXAMPLE sipsak -U -C sip:me@home -x 3600 -a password -s sip:myself@company (Insert a forwarding contact for myself at work to me at home for one hour and authenticated with password if required)

EXAMPLE sipsak -I -C empty -a password -s sip:myself@work (Query the currently registered bindings for myself at work and authenticate with password if required)

EXAMPLE sipsak -M -v -s sip:colleaue@work -B "Lunch time!“ (Send the instant message "Lunch time!" to the colleague and show result)

spike-generic_chunked

DESCRIPTION When you need to analyze a new network protocol for buffer overflows or similar weaknesses, the **SPIKE** is the tool of choice for professionals. While it requires a strong knowledge of C to use, it produces results second to none in the field. It gives you the possibility to incorporate his APIs inside the C code or simply using some pre-built tools which processes scripts created using the **SPIKE's** primitives.

More information: http://resources.infosecinstitute.com/intro-to-fuzzing/
More information: http://resources.infosecinstitute.com/fuzzer-automation-with-spike/

USAGE n/a
EXAMPLE n/a

spike-generic_listen_tcp

DESCRIPTION When you need to analyze a new network protocol for buffer overflows or similar weaknesses, the **SPIKE** is the tool of choice for professionals. While it requires a strong knowledge of C to use, it produces results second to none in the field. It gives you the possibility to incorporate his APIs inside the C code or simply using some pre-built tools which processes scripts created using the **SPIKE's** primitives.

More information: http://resources.infosecinstitute.com/intro-to-fuzzing/
More information: http://resources.infosecinstitute.com/fuzzer-automation-with-spike/

USAGE n/a
EXAMPLE n/a

spike-generic_send_tcp

DESCRIPTION When you need to analyze a new network protocol for buffer overflows or similar weaknesses, the **SPIKE** is the tool of choice for professionals. While it requires a strong knowledge of C to use, it produces results second to none in the field. It gives you the possibility to incorporate his APIs inside the C code or simply using some pre-built tools which processes scripts created using the **SPIKE's** primitives.

More information: http://resources.infosecinstitute.com/intro-to-fuzzing/
More information: http://resources.infosecinstitute.com/fuzzer-automation-with-spike/

USAGE n/a
EXAMPLE n/a

spike-generic_listen_upd

DESCRIPTION When you need to analyze a new network protocol for buffer overflows or similar weaknesses, the **SPIKE** is the tool of choice for professionals. While it requires a strong knowledge of C to use, it produces results second to none in the field. It gives you the possibility to incorporate his APIs inside the C code or simply using some pre-built tools which processes scripts created using the **SPIKE's** primitives.

More information: http://resources.infosecinstitute.com/intro-to-fuzzing/
More information: http://resources.infosecinstitute.com/fuzzer-automation-with-spike/

USAGE n/a
EXAMPLE n/a

[20] MISC SCANNERS

- lynis
- nikto
- nmap
- unix-privesc-check

lynis

DESCRIPTION **Lynis** is an auditing tool for Unix (specialists). It scans the system and available software, to detect security issues. Beside security related information it will also scan for general system information, installed packages and configuration mistakes.
This software aims in assisting automated auditing, software patch management, vulnerability and malware scanning of Unix based systems. It can be run without prior installation, so inclusion on read only storage is no problem (USB stick, cd/dvd).
Lynis assists auditors in performing Basel II, GLBA, HIPAA, PCI DSS and SOX (Sarbanes-Oxley) compliance audits.

USAGE ./lynis [options] [cronjob]

OPTIONS http://www.rootkit.nl/files/lynis-documentation.html

EXAMPLE ./lynis -c --auditor "automated" --cronjob

nikto

DESCRIPTION **Nikto** is web server scanner which performs comprehensive tests against web servers for multiple items, including over 6500 potentially dangerous files/CGIs, checks for outdated versions of over 1250 servers, and version specific problems on over 270 servers. It also checks for server configuration items such as the presence of multiple index files, HTTP server options, and will attempt to identify installed web servers and software. Scan items and plugins are frequently updated and can be automatically updated.

USAGE nikto.pl [host] [target] [options]

OPTIONS http://cirt.net/nikto2-docs/options.html

EXAMPLE nikto.pl -h 192.168.0.1 -T 58 (scan tuning)
EXAMPLE nmap -p80 192.168.0.0/24 -oG - | nikto.pl -h - (Multiple Host Testing)
EXAMPLE nikto.pl -h 192.168.0.1 -p 80,88,443 (Multiple Port Testing)
EXAMPLE nikto.pl -h https://192.168.0.1:443/ (basic testing and scanning using a full URL syntax)
EXAMPLE nikto.pl -h 192.168.0.1 -p 443 (checking specific port)

nmap

DESCRIPTION **nmap** is certainly THE scanner to know. Thanks to its numerous parameters, it is a swiss army knife to all situations where network identification is needed. It enables among other things to list network hosts and scan their ports.

USAGE ./nmap [Scan Type(s)] [Options] {target specification}

EXAMPLE ./nmap -sP 192.168.100.0/24 (Lists hosts on a network)

EXAMPLE ./nmap -sS -sV 192.168.100.18 (Scans a host. This example uses a TCP/SYN scan and tries to identify installed services)

unix-privesc-check

DESCRIPTION **Unix-privesc-checker** is a script that runs on Unix systems. It tries to find misconfigurations that could allow local unprivileged users to escalate privileges to other users or to access local apps (e.g., databases).

It's intended to be run by security auditors and penetration testers against systems they have been engaged to assess, and also by system administrators who want to check for "obvious" misconfigurations. It can even be run as a cron job so you can check regularly for misconfigurations that might be introduced.

More info: https://code.google.com/p/unix-privesc-check/wiki/Checks and http://pentestmonkey.net/tools/unix-privesc-check

USAGE unix-privesc-check { standard | detailed }

EXAMPLE unix-privesc-check standard

[21] OPEN SOURCE ASSESSMENT

- casefile
- maltego

casefile

DESCRIPTION **CaseFile** gives you the ability to quickly add, link and analyze data having the same graphing flexibility and performance as **Maltego** without the use of transforms. Combining **Maltego's** fantastic graph and link analysis this tool allows for analysts to examine links between manually added data to mind map your information.

- **CaseFile** is a visual intelligence application that can be used to determine the relationships and real world links between hundreds of different types of information.
- It gives you the ability to quickly view second, third and n-th order relationships and find links otherwise undiscoverable with other types of intelligence tools.
- **CaseFile** comes bundled with many different types of entities that are commonly used in investigations allowing you to act quickly and efficiently. **CaseFile** also has the ability to add custom entity types allowing you to extend the product to your own data sets.

USAGE n/a; GUI tool

EXAMPLE n/a, GUI tool

maltego

DESCRIPTION **Maltego** is a unique platform developed to deliver a clear threat picture to the environment that an organization owns and operates. **Maltego** can locate, aggregate and visualize this information. **Maltego** is a program that can be used to determine the relationships and real world links between people, groups of people (social networks), companies, organizations, web sites, phrases, affiliations, documents and files, internet infrastructure (domains, DNS names, netblocks, IP addresses).

USAGE n/a, GUI tool

EXAMPLE n/a, GUI tool

[22] OPEN-VAS

- penvas-gsd
- openvas-setup

openvas-gsd

DESCRIPTION The **Open Vulnerability Assessment System (OpenVAS**) is a framework of several services and tools. The core of this SSL-secured service-oriented architecture is the **OpenVAS Scanner**. The scanner very efficiently executes the actual Network Vulnerability Tests (NVTs) which are served with daily updates via the OpenVAS NVT Feed or via a commercial feed service.

The **Greenbone Security Desktop (GSD)** is a Qt-based desktop client for OMP.

More info: http://www.backtrack-linux.org/wiki/index.php/OpenVas

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

openvas-setup

DESCRIPTION The **Open Vulnerability Assessment System (OpenVAS**) is a framework of several services and tools. The core of this SSL-secured service-oriented architecture is the **OpenVAS Scanner**. The scanner very efficiently executes the actual Network Vulnerability Tests (NVTs) which are served with daily updates via the OpenVAS NVT Feed or via a commercial feed service.

More info: http://www.backtrack-linux.org/wiki/index.php/OpenVas

USAGE n/a; GUI tool

EXAMPLE n/a GUI tool

[23] WEB APPLICATIONS: CMS IDENTIFACTION

- blindelephant
- plecost
- wpscan

blindelephant

DESCRIPTION The **BlindElephant** Web Application Fingerprinter attempts to discover the version of a (known) web application by comparing static files at known locations against precomputed hashes for versions of those files in all of the available releases. The technique is fast, low-bandwidth, non-invasive, generic, and highly automatable. **BlindElephant** can be used directly as a tool on the command line, or as a library to provide fingerprinting functionality to another program.

USAGE BlindElephant.py [options] url appName

OPTIONS

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

-p PLUGINNAME, --pluginName=PLUGINNAME Fingerprint version of plugin (should apply to web app given in appname)

-s, --skip Skip fingerprinting webpp, just fingerprint plugin

-n NUMPROBES, --numProbes=NUMPROBES Number of files to fetch (more may increase accuracy). Default: 15

-w, --winnow If more than one version are returned, use winnowing to attempt to narrow it down (up to numProbes additional requests)

-l, --list List supported webapps and plugins

EXAMPLE BlindElephant.py http://laws.qualys.com movabletype

plecost

DESCRIPTION Wordpress finger printer tool, **plecost** searches and retrieves information about the plugin versions installed in Wordpress systems. It can analyze a single URL or perform an analysis based on the results indexed by Google. Additionally displays CVE code associated with each plugin, if exists.

Plecost retrieves the information contained on Web sites supported by Wordpress, and also allows a search on the results indexed by Google.

USAGE ./plecost.py [options] [URL | [-l num] -G]

OPTIONS https://code.google.com/p/plecost/

EXAMPLE plecost -R plugins.txt -n 5 (Reload first 5 plugins list)
EXAMPLE plecost -n 5 -G -i plugins.txt (Search vulnerable sites for first 5 plugins)
EXAMPLE plecost -i plugin_list.txt -s 12 -M 30 -t 20 -o results.txt www.example.com (Search plugins with 20 threads, sleep time between 12 and 30 seconds for www.example.com)

wpscan

DESCRIPTION **WPScan** is a black box WordPress vulnerability scanner.

USAGE wpscan.rb –url [target ip] [options]

OPTIONS http://wpscan.org/

EXAMPLE ruby wpscan.rb --url www.example.com (Do 'non-intrusive' checks...)

EXAMPLE ruby wpscan.rb --url www.example.com --wordlist darkc0de.lst --threads 50 (Do wordlist password brute force on enumerated users using 50 threads...)

EXAMPLE ruby wpscan.rb --url www.example.com --wordlist darkc0de.lst --username admin (Do wordlist password brute force on the 'admin' username only...)

EXAMPLE ruby wpscan.rb --url www.example.com --enumerate p (Enumerate installed plugins...)

EXAMPLE ruby wpscan.rb --url www.example.com –enumerate (Run all enumeration tools...)

[24] DATABASE EXPLOITATION

- bbqsql
- sqlninja
- sqlsus

bbqsql

DESCRIPTION Blind SQL injection can be a pain to exploit. When the available tools work they work well, but when they don't you have to write something custom. This is time-consuming and tedious. **BBQSQL** can help you address those issues.

BBQSQL is a blind SQL injection framework written in Python. It is extremely useful when attacking tricky SQL injection vulnerabilities. **BBQSQL** is also a semi-automatic tool, allowing quite a bit of customization for those hard to trigger SQL injection findings. The tool is built to be database agnostic and is extremely versatile. It also has an intuitive UI to make setting up attacks much easier. Python gevent is also implemented, making **BBQSQL** extremely fast.

USAGE n/a, option selection/configuration

OPTIONS https://github.com/Neohapsis/bbqsql

EXAMPLE n/a

sqlninja

DESCRIPTION **Sqlninja** is a tool targeted to exploit SQL Injection vulnerabilities on a web application that uses Microsoft SQL Server as its back-end. Its main goal is to provide a remote access on the vulnerable DB server, even in a very hostile environment. It should be used by penetration testers to help and automate the process of taking over a DB Server when a SQL Injection vulnerability has been discovered.

USAGE configuration and execution happens via sqlninja.conf; see documentation for more details

OPTIONS http://sqlninja.sourceforge.net/sqlninja-howto.html

EXAMPLE n/a

sqlsus

DESCRIPTION **sqlsus** is an open source MySQL injection and takeover tool. Via a command line interface, you can retrieve the database(s) structure, inject your own SQL queries (even complex ones), download files from the web server, crawl the website for writable directories, upload and control a backdoor, clone the database(s), and much more... Whenever relevant, sqlsus will mimic a MySQL console output.
More info: http://sqlsus.sourceforge.net/

USAGE Generate a configuration file with **sqlsus --genconf my.cfg**, read the comments and adapt it to reflect your target. Launch **sqlsus**, with your configuration as a parameter sqlsus my.cfg, you will get a shell.

OPTIONS http://sqlsus.sourceforge.net/documentation.html

EXAMPLE n/a, check documentation

[25] IDS/IPS IDENTIFACATION

- ua-tester

ua-tester

DESCRIPTION **UATester** is a python script that requests a page from a webserver with different user-agents. With this script one can quickly discover if different pages exist for different user-agents (e.g. mobile page, IE optimized page, etc). This tool is designed to automatically check a given URL using a list of standard and non-standard User Agent strings provided by the user (1 per line). The results of these checks are then reported to the user for further manual analysis where required.

USAGE ./uatester.py –u <url> [options]

OPTIONS http://packetstormsecurity.com/files/94305/UA-Tester-User-Agent-Tester-1.03.html

EXAMPLE ./UATester.py -u www.example.com -f ./useragentlist.txt -v
EXAMPLE ./UATester.py -u https://www.wordpress.com
EXAMPLE ./UATester.py -u http://www.defaultserver.com -v --debug
EXAMPLE ./UATester.py -u facebook.com -v -d MDBX
EXAMPLE ./UATester.py -u https://www.google.com -s "MySpecialUserAgent"\n'"

[26] WEB APPLICATION FUZZERS

- burpsuite
- powerfuzzer
- webscarab
- webslayer
- websploit
- wfuzz
- xsser
- zaproxy

burpsuite

DESCRIPTION **Burp Suite** is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp Suite *contains the following key components:*

- *An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.*
- *An application-aware **Spider**, for crawling content and functionality.*
- *An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.*
- *An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.*
- *A **Repeater** tool, for manipulating and resending individual requests.*
- *A **Sequencer** tool, for testing the randomness of session tokens.*
- *The ability to **save your work** and resume working later.*
- ***Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.*

More info: http://portswigger.net/burp/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

powerfuzzer

DESCRIPTION **Powerfuzzer** is a highly automated and fully customizable web fuzzer (HTTP protocol based application fuzzer) based on many other Open Source fuzzers available and information gathered from numerous security resources and websites. It was designed to be user friendly, modern, effective and working.
More info: http://www.powerfuzzer.com/#news

Currently, it is capable of identifying these problems:
- Cross Site Scripting (XSS)
- Injections (SQL, LDAP, code, commands, and XPATH)
- CRLF
- HTTP 500 statuses (usually indicative of a possible misconfiguration/security flaw incl. buffer overflow)

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webscarab

DESCRIPTION **WebScarab** is a framework for analysing applications that communicate using the HTTP and HTTPS protocols. It is written in Java, and is thus portable to many platforms. **WebScarab** has several modes of operation, implemented by a number of plugins. In its most common usage, **WebScarab** operates as an intercepting proxy, allowing the operator to review and modify requests created by the browser before they are sent to the server, and to review and modify responses returned from the server before they are received by the browser. **WebScarab** is able to intercept both HTTP and HTTPS communication. The operator can also review the conversations (requests and responses) that have passed through **WebScarab**.

More info: https://www.owasp.org/index.php/WebScarab_Getting_Started

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webslayer

DESCRIPTION **WebSlayer** is a tool designed for brute forcing Web Applications, it can be used to discover not linked resources (directories, servlets, scripts, etc), brute force GET and POST parameters, brute force forms parameters (User/Password), fuzzing, etc.

The tool has a powerful payload generator and a easy and flexible results analyzer.

More info: https://www.owasp.org/index.php/Category:OWASP_Webslayer_Project

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

websploit

DESCRIPTION **WebSploit** is used to scan and analysis remote system in order to find various type of vulnerabilities. This tool is very powerful and supports multiple vulnerabilities. **WebSploit** is used for: Social Engineering Works, Scan, Crawler & Web Analysis, Automatic Exploiter and Support Network Attacks.

Features:

- Autopwn - Used From Metasploit For Scan and Exploit Target Service
- wmap - Scan,Crawler Target Used From Metasploit wmap plugin
- format infector - inject reverse & bind payload into file format
- phpmyadmin Scanner
- LFI Bypasser
- Apache Users Scanner
- Dir Bruter
- admin finder
- MLITM Attack - Man Left In The Middle, XSS Phishing Attacks
- MITM - Man In The Middle Attack
- Java Applet Attack
- MFOD Attack Vector
- USB Infection Attack
- ARP Dos Attack
- Web Killer Attack
- Fake Update Attack
- Fake Access point Attack

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

wfuzz

DESCRIPTION **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.
More info: https://code.google.com/p/wfuzz/wiki/Howto

USAGE wfuzz.py [options] [file] [config] [target ip]

EXAMPLE wfuzz.py -c -z file, wordlists/commons.txt --hc 404 -R 2 http://www.mysite.com/FUZZ (Example of path discovery, using a recursive level of 2)
EXAMPLE wfuzz.py -z file,wordlists/http_methods.txt -X http://testphp.vulnweb.com/ (HTTP method scanning example)
EXAMPLE wfuzz.py -z list,TRACE -X http://testphp.vulnweb.com/ (Scanning for TRACE method using a list payload)
EXAMPLE wfuzz.py -c -z file,wordlists/methods.txt --hc 404 -v --follow http://www.mysite.com/FUZZ (Bruteforce following HTTP redirects)
EXAMPLE wfuzz.py -c -z file,wordlists/commons.txt --hc 404 -I http://www.mysite.com/FUZZ (Bruteforce using HEAD HTTP method)
EXAMPLE wfuzz.py -z list,http://mysite.com -z list,dir-dir2-dir3 FUZZ/FUZ2Z (Bruteforce using URL as payload and a list of directories.)

xsser

DESCRIPTION Cross Site "Scripter" (aka **XSSer**) is an automatic -framework- to detect, exploit and report XSS vulnerabilities in web-based applications. It contains several options to try to bypass certain filters, and various special techniques of code injection.

USAGE xsser [OPTIONS] [-u |-i |-d] [-g |-p |-c] [Request(s)] [Vector(s)] [Bypasser(s)] [Technique(s)] [Final Injection(s)]

OPTIONS http://xsser.sourceforge.net/

EXAMPLE python xsser.py -u http://host.com (Simple injection from URL)

EXAMPLE python xsser.py -i "file.txt" --proxy "http://127.0.0.1:8118" --referer "666.666.666.666“ (Simple injection from File, with tor proxy and spoofing HTTP Referer headers)

EXAMPLE python xsser.py -u "http://host.com" -p index.php?target=search&subtarget=top&searchstring=" –s (Simple injection from URL, using POST, with statistics results)

EXAMPLE python xsser.py -u "host.com" –hash (Send a pre-checking hash to see if target will generate -false positive- results)

zaproxy

DESCRIPTION The **OWASP Zed Attack Proxy (ZAP)** is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and as such is ideal for developers and functional testers who are new to penetration testing as well as being a useful addition to an experienced pen testers' toolbox.

More info: https://code.google.com/p/zaproxy/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[27] WEB APPLICATION PROXIES

- burpsuite
- paros
- proxystrike
- vega
- webscarab
- zaproxy

burpsuite

DESCRIPTION **Burp Suite** is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp Suite *contains the following key components:*

- *An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.*
- *An application-aware **Spider**, for crawling content and functionality.*
- *An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.*
- *An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.*
- *A **Repeater** tool, for manipulating and resending individual requests.*
- *A **Sequencer** tool, for testing the randomness of session tokens.*
- *The ability to **save your work** and resume working later.*
- ***Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.*

More info: http://portswigger.net/burp/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

paros

DESCRIPTION **Paros** is a valuable testing tool for your security and vulnerability testing. **Paros** can be used to spider/crawl your entire site, and then execute canned vulnerability scanner tests. But **Paros** goes beyond that, it comes with a built in utility that can proxy traffic. This **Paros Proxy** utility can be used to tamper or manipulate any http or https traffic on the fly. This makes some of the more interesting security types of testing. It will help you isolate potential area's of security concern and then manual attempt to perform the type of testing you desire.

More info: http://www.testingsecurity.com/paros_proxy and http://www.parosproxy.org/

To get it working run ./startserver.sh and configure the browser you are using so the proxy uses 127.0.01: 8080 for http and https.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

proxystrike

DESCRIPTION **ProxyStrike** is an active Web Application Proxy. It's a tool designed to find vulnerabilities while browsing an application. It was created because the problems we faced in the pentests of web applications that depends heavily on Javascript, not many web scanners did it good in this stage, so we came with this proxy.

Right now it has available SQL injection and XSS plugins. Both plugins are designed to catch as many vulnerabilities as we can, it's that why the SQL Injection plugin is a Python port of the great DarkRaver "Sqlibf".

The process is very simple, **ProxyStrike** runs like a proxy listening in port 8008 by default, so you have to browse the desired web site setting your browser to use **ProxyStrike** as a proxy, and **ProxyStrike** will analyze all the parameters in background mode. For the user is a passive proxy because you won't see any different in the behaviour of the application, but in the background is very active.

More info: https://code.google.com/p/proxystrike

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

vega

DESCRIPTION **Vega** is an open source platform to test the security of web applications. **Vega** can help you find and validate SQL Injections, Cross-Site Scripting (XSS), inadvertently disclosed sensitive information, and other vulnerabilities.

Features:

- *Automated Crawler and Vulnerability Scanner*
- *Consistent UI*
- *Website Crawler*
- *Intercepting Proxy*
- *SSL MITM*
- *Content Analysis*
- *Extensibility through a Powerful Javascript Module API*
- *Customizable alerts*
- *Database and Shared Data Model*

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webscarab

DESCRIPTION **WebScarab** is a framework for analysing applications that communicate using the HTTP and HTTPS protocols. It is written in Java, and is thus portable to many platforms. **WebScarab** has several modes of operation, implemented by a number of plugins. In its most common usage, **WebScarab** operates as an intercepting proxy, allowing the operator to review and modify requests created by the browser before they are sent to the server, and to review and modify responses returned from the server before they are received by the browser. **WebScarab** is able to intercept both HTTP and HTTPS communication. The operator can also review the conversations (requests and responses) that have passed through **WebScarab**.

More info: https://www.owasp.org/index.php/WebScarab_Getting_Started

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

zaproxy

DESCRIPTION The **OWASP Zed Attack Proxy (ZAP)** is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and as such is ideal for developers and functional testers who are new to penetration testing as well as being a useful addition to an experienced pen testers' toolbox.

More info: https://code.google.com/p/zaproxy/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[28] WEB CRAWLERS

- apache-users
- burpsuite
- cutycapt
- dirb
- dirbuster
- vega
- webscarab
- webslayer
- zaproxy

apache-users

DESCRIPTION **apache-users** allows searching server username directory (if use from apache webserver) / allows scanning directory of apache users. This module is part of the WebSploit Toolkit.

USAGE n/a

EXAMPLE n/a

burpsuite

DESCRIPTION **Burp Suite** is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp Suite *contains the following key components:*

- *An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.*
- *An application-aware **Spider**, for crawling content and functionality.*
- *An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.*
- *An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.*
- *A **Repeater** tool, for manipulating and resending individual requests.*
- *A **Sequencer** tool, for testing the randomness of session tokens.*
- *The ability to **save your work** and resume working later.*
- ***Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.*

More info: http://portswigger.net/burp/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

cutycapt

DESCRIPTION **CutyCapt** is a small cross-platform command-line utility to capture WebKit's rendering of a web page into a variety of vector and bitmap formats, including SVG, PDF, PS, PNG, JPEG, TIFF, GIF, and BMP.

USAGE CutyCapt --url=<target ip> --out=<output file>.<extension>

OPTIONS http://cutycapt.sourceforge.net/

EXAMPLE CutyCapt --url=http://www.example.org/ --out=localfile.png

TIP

Using CutyCapt without X server

You cannot use CutyCapt without an X server, but you can use e.g. Xvfb as light-weight server if you are not running an interactive graphical desktop environment.

For example, you could use:

```
% xvfb-run --server-args="-screen 0, 1024x768x24" ./CutyCapt --url=... --out=...
```

dirb

DESCRIPTION **DIRB** is a Web Content Scanner. It looks for existing (and/or hidden) Web Objects. It basically works by launching a dictionary based attack against a web server and analysing the response.
DIRB comes with a set of preconfigured attack wordlists for easy usage but you can use your custom wordlists. Also **DIRB** sometimes can be used as a classic CGI scanner, but remember – it is a content scanner not a vulnerability scanner.

More info: http://dirb.sourceforge.net/faq.txt

USAGE dirb <url> <your dict file>

EXAMPLE ./dirb http://www.site.com/directory1/ wordlist.txt

EXAMPLE ./dirb http://www.site.com/ wordlist.txt,dirnamefile.txt

dirbuster

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

More info: https://www.owasp.org/index.php/Category:OWASP_DirBuster_Project

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

vega

DESCRIPTION **Vega** is an open source platform to test the security of web applications. **Vega** can help you find and validate SQL Injections, Cross-Site Scripting (XSS), inadvertently disclosed sensitive information, and other vulnerabilities.

Features:

- *Automated Crawler and Vulnerability Scanner*
- *Consistent UI*
- *Website Crawler*
- *Intercepting Proxy*
- *SSL MITM*
- *Content Analysis*
- *Extensibility through a Powerful Javascript Module API*
- *Customizable alerts*
- *Database and Shared Data Model*

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webscarab

DESCRIPTION **WebScarab** is a framework for analysing applications that communicate using the HTTP and HTTPS protocols. It is written in Java, and is thus portable to many platforms. **WebScarab** has several modes of operation, implemented by a number of plugins. In its most common usage, **WebScarab** operates as an intercepting proxy, allowing the operator to review and modify requests created by the browser before they are sent to the server, and to review and modify responses returned from the server before they are received by the browser. **WebScarab** is able to intercept both HTTP and HTTPS communication. The operator can also review the conversations (requests and responses) that have passed through **WebScarab**.

More info: https://www.owasp.org/index.php/WebScarab_Getting_Started

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webslayer

DESCRIPTION **WebSlayer** is a tool designed for brute forcing Web Applications, it can be used to discover not linked resources (directories, servlets, scripts, etc), brute force GET and POST parameters, brute force forms parameters (User/Password), fuzzing, etc.

The tool has a powerful payload generator and a easy and flexible results analyzer.

More info: https://www.owasp.org/index.php/Category:OWASP_Webslayer_Project

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

zaproxy

DESCRIPTION The **OWASP Zed Attack Proxy (ZAP)** is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and as such is ideal for developers and functional testers who are new to penetration testing as well as being a useful addition to an experienced pen testers' toolbox.

More info: https://code.google.com/p/zaproxy/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[29] WEB VULNERABILITY SCANNERS

- burpsuite
- cadaver
- davtest
- deblaze
- fimap
- grabber
- joomscan
- nikto
- padbuster
- proxystrike
- skipfish
- sqlmap
- vega
- w3af
- wapiti
- webscarab
- webshag-cli
- webshaggui
- websploit
- wpscan
- xsser
- zaproxy

burpsuite

DESCRIPTION **Burp Suite** is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp Suite *contains the following key components:*

- *An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.*
- *An application-aware **Spider**, for crawling content and functionality.*
- *An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.*
- *An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.*
- *A **Repeater** tool, for manipulating and resending individual requests.*
- *A **Sequencer** tool, for testing the randomness of session tokens.*
- *The ability to **save your work** and resume working later.*
- ***Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.*

More info: http://portswigger.net/burp/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

cadaver

DESCRIPTION **cadaver** is a command-line WebDAV client for Unix. It supports file upload, download, on-screen display, namespace operations (move/copy), collection creation and deletion, and locking operations.

USAGE ./cadaver xxx.xxx.xxx.xxx

USAGE cadaver [-et][-V][-h] http://hostname[:port]/path

OPTIONS http://linux.die.net/man/1/cadaver

EXAMPLE cadaver http://dav.example.com/ (Connects to the server myserver.example.com, opening the root collection)

EXAMPLE cadaver http://zope.example.com:8022/Users/fred/ (Connects to the server zope.example.com using port 8022, opening the collection "/Users/fred/")

EXAMPLE cadaver https://secure.example.com/ (Connects to a server called secure.example.com using SSL)

TIP

~/.cadaverrcIndividual user settings that can override cadaver defaults.

davtest

DESCRIPTION DAVTest tests WebDAV enabled servers by uploading test executable files, and then (optionally) uploading files which allow for command execution or other actions directly on the target.

Features

- Automatically send exploit files if code executes
- Automatic randomization of directory to help hide files using MKCOL
- Send text files and try MOVE to executable name
- Basic and Digest authorization
- Automatic clean-up of uploaded files
- Send an arbitrary file

USAGE davtest.pl –url <url> [options]

OPTIONS https://code.google.com/p/davtest/wiki/Usage

EXAMPLE davtest.pl -url http://localhost/davdir (Test file uploads at this location)

EXAMPLE davtest.pl -url http://localhost/davdir -sendbd auto (Test file uploads at this location and send backdoors for any types which execute successfully)

EXAMPLE davtest.pl -url http://localhost/davdir -auth user:pass -uploadfile backdoors/perl_cmd.pl -uploadloc perl.pl (Upload a file using authentication, send the perl_cmd.pl backdoor and call it perl.pl on the server)

deblaze

DESCRIPTION **deblaze** is a remote method enumeration tool for flex servers. This tool will allow you to perform method enumeration and interrogation against flash remoting end points. **Deblaze** provides the following functionality: Brute Force Service and Method Names, Method Interrogation, Flex Technology Fingerprinting, Parameter detection, Basic parameter fuzzing, Proxy AMF requests/responses, HTML reporting.

USAGE deblaze [option]

OPTIONS http://deblaze-tool.appspot.com/usage

EXAMPLE deBlaze.py -u http://192.168.165.132:8080/amfphp/gateway.php -s securityService -m sendEmail -p test@test.com
EXAMPLE python2.5 deblaze-0.3.py -P 8080:targetIP:targetPort
EXAMPLE python deBlaze.py -u http://192.168.165.132:8080/amfphp/gateway.php -s Discoveryservice -m getServices
EXAMPLE python deBlaze.py -u http://192.168.165.132:8400/samples/messagebroker/qosamfpolling -1 names.txt -m test

fimap

DESCRIPTION **fimap** is a little python tool which can find, prepare, audit, exploit and even google automatically for local and remote file inclusion bugs in webapps. **fimap** should be something like **sqlmap** just for LFI/RFI bugs instead of sql injection. The goal of **fimap** is to improve the quality and security of your website.

More info: https://code.google.com/p/fimap

USAGE ./fimap.py [options]

OPTIONS https://code.google.com/p/fimap/wiki/FimapHelpPage

EXAMPLE fimap -u http://localhost/vulnerable.php?inc=index.php
EXAMPLE ./fimap.py -u 'http://localhost/test.php?file=bang&id=23‘ (Scan a single URL for FI errors)
EXAMPLE ./fimap.py -m -l '/tmp/urllist.txt‘ (Scan a list of URLS for FI errors)
EXAMPLE ./fimap.py -g -q 'inurl:include.php‘ (Scan Google search results for FI errors)
EXAMPLE ./fimap.py -H -u 'http://localhost' -d 3 -w /tmp/urllist (Harvest all links of a webpage with recurse level of 3 and write the URLs to /tmp/urllist)

grabber

DESCRIPTION **Grabber** is a web application scanner. Basically it detects some kind of vulnerabilities in your website. **Grabber** is simple, not fast but portable and really adaptable. This software is designed to scan small websites such as personals, forums etc. absolutely not big application: it would take too long time and flood your network.

Features

- *Cross-Site Scripting*
- *SQL Injection (there is also a special Blind SQL Injection module)*
- *File Inclusion*
- *Backup files check*
- *Simple AJAX check (parse every JavaScript and get the URL and try to get the parameters)*
- *Hybrid analysis/Crystal ball testing for PHP application using PHP-SAT*
- *JavaScript source code analyzer: Evaluation of the quality/correctness of the JavaScript with JavaScript Lint*
- *Generation of a file [session_id, time(t)] for next stats analysis.*

USAGE grabber.py [options]

OPTIONS http://www.securitytube-tools.net/index.php@title=Grabber.html

EXAMPLE read: http://rgaucher.info/beta/grabber/

joomscan

DESCRIPTION **joomscan** detects file inclusion, SQL injection, command execution vulnerabilities of a target **Joomla**! web site.

Features

- *Exact version Probing (the scanner can tell whether a target is running version 1.5.12)*
- *Common Joomla! based web application firewall detection*
- *Searching known vulnerabilities of Joomla! and its components*
- *Reporting to Text & HTML output*
- *Immediate update capability via scanner or svn*

USAGE joomscan.pl -u <string> -x proxy:port

OPTIONS https://www.owasp.org/index.php/OWASP_Joomla_Vulnerability_Scanner_Usage

EXAMPLE joomscan.pl -pv -u victim.com -x localhost:8080

EXAMPLE joomscan.pl read DOCFILE (where DOCFILE is one of these: changelog,release_note,readme,credits,faq,owasp_project)

nikto

DESCRIPTION **Nikto** is web server scanner which performs comprehensive tests against web servers for multiple items, including over 6500 potentially dangerous files/CGIs, checks for outdated versions of over 1250 servers, and version specific problems on over 270 servers. It also checks for server configuration items such as the presence of multiple index files, HTTP server options, and will attempt to identify installed web servers and software. Scan items and plugins are frequently updated and can be automatically updated.

USAGE nikto.pl [host] [target] [options]

OPTIONS http://cirt.net/nikto2-docs/options.html

EXAMPLE nikto.pl -h 192.168.0.1 -T 58 (scan tuning)
EXAMPLE nmap -p80 192.168.0.0/24 -oG - | nikto.pl -h - (Multiple Host Testing)
EXAMPLE nikto.pl -h 192.168.0.1 -p 80,88,443 (Multiple Port Testing)
EXAMPLE nikto.pl -h https://192.168.0.1:443/ (basic testing and scanning using a full URL syntax)
EXAMPLE nikto.pl -h 192.168.0.1 -p 443 (checking specific port)

padbuster

DESCRIPTION **PadBuster** - Automated script for performing Padding Oracle attacks. **PadBuster** provides the capability to decrypt arbitrary ciphertext, encrypt arbitrary plaintext, and perform automated response analysis to determine whether a request is vulnerable to padding oracle attacks.

More information and how to use: http://blog.gdssecurity.com/labs/2010/9/14/automated-padding-oracle-attacks-with-padbuster.html and http://howtohack.poly.edu/wiki/Padding_Oracle_Attack

USAGE padBuster.pl <url> <encrypted sample> <block size> <encoding>

EXAMPLE padBuster.pl
http://sampleapp/home.jsp?UID=7B216A634951170FF851D6CC68FC9537858795A28ED4AAC6
7B216A634951170FF851D6CC68FC9537858795A28ED4AAC6 8 -encoding 2

EXAMPLE padBuster.pl
http://sampleapp/home.jsp?UID=7B216A634951170FF851D6CC68FC9537858795A28ED4AAC6
7B216A634951170FF851D6CC68FC9537858795A28ED4AAC6 8 -encoding 2 -plaintext "ENCRYPT TEST"

proxystrike

DESCRIPTION **ProxyStrike** is an active Web Application Proxy. It's a tool designed to find vulnerabilities while browsing an application. It was created because the problems we faced in the pentests of web applications that depends heavily on Javascript, not many web scanners did it good in this stage, so we came with this proxy.

Right now it has available SQL injection and XSS plugins. Both plugins are designed to catch as many vulnerabilities as we can, it's that why the SQL Injection plugin is a Python port of the great DarkRaver "Sqlibf".

The process is very simple, **ProxyStrike** runs like a proxy listening in port 8008 by default, so you have to browse the desired web site setting your browser to use **ProxyStrike** as a proxy, and **ProxyStrike** will analyze all the parameters in background mode. For the user is a passive proxy because you won't see any different in the behaviour of the application, but in the background is very active.

More info: https://code.google.com/p/proxystrike

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

skipfish

DESCRIPTION **Skipfish** is an active web application security reconnaissance tool. It prepares an interactive sitemap for the targeted site by carrying out a recursive crawl and dictionary-based probes. The resulting map is then annotated with the output from a number of active (but hopefully non-disruptive) security checks. The final report generated by the tool is meant to serve as a foundation for professional web application security assessments.

USAGE ./skipfish [options] -o output_dir @/path/to/url_list.txt

OPTIONS https://code.google.com/p/skipfish/wiki/SkipfishDoc

EXAMPLE ./skipfish -o output_dir -S existing_dictionary.wl -W new_dict.wl \
http://www.example.com/some/starting/path.txt

EXAMPLE ./skipfish -D test2.example.com -o output-dir http://test1.example.com/

EXAMPLE ./skipfish –D

./skipfish -MEU -S dictionaries/minimal.wl -W new_dict.wl \
-C "AuthCookie=value" -X /logout.aspx -o output_dir \
http://www.example.com/ .example.com -o output-dir http://test1.example.com/

sqlmap

DESCRIPTION **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.

USAGE sqlmap.py [options]

OPTIONS https://github.com/sqlmapproject/sqlmap/wiki/Usage

EXAMPLE sqlmap.py -u "http://www.target.com/vuln.php?id=1" -f --banner --dbs –users (Run sqlmap against a single target URL)
EXAMPLE sqlmap.py -g "inurl:\".php?id=1\"" (test and inject on GET parameters based on results of your Google dork)
EXAMPLE sqlmap.py -u "http://www.target.com/vuln.php" --data="id=1" -f --banner --dbs --users (hunt for POST requests)
EXAMPLE sqlmap.py -u "http://192.168.136.131/sqlmap/mysql/basic/get_int.php?id=1" \ --auth-type Basic --auth-cred "testuser:testpass" (HTTP authentication)
EXAMPLE sqlmap.py -l burp.log --scope="(www)?\.target\.(com|net|org)“ (Filtering targets from provided proxy log using regular expression)

vega

DESCRIPTION **Vega** is an open source platform to test the security of web applications. **Vega** can help you find and validate SQL Injections, Cross-Site Scripting (XSS), inadvertently disclosed sensitive information, and other vulnerabilities.

Features:

- *Automated Crawler and Vulnerability Scanner*
- *Consistent UI*
- *Website Crawler*
- *Intercepting Proxy*
- *SSL MITM*
- *Content Analysis*
- *Extensibility through a Powerful Javascript Module API*
- *Customizable alerts*
- *Database and Shared Data Model*

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

w3af

DESCRIPTION **w3af** is a **Web Application Attack and Audit Framework**. The project provides a vulnerability scanner and exploitation tool for Web applications. It provides information about security vulnerabilities and aids in penetration testing efforts. **w3af** identifies most web application vulnerabilities using more than 130 plug-ins. After identification, vulnerabilities like (blind) SQL injections, OS commanding, remote file inclusions (PHP), cross-site scripting (XSS), and unsafe file uploads, can be exploited in order to gain different types of access to the remote system.

More info: http://w3af.org/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

wapiti

DESCRIPTION **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 scans 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 can detect the following vulnerabilities :

- *File Handling Errors (Local and remote include/require, fopen, readfile...)*
- *Database Injection (PHP/JSP/ASP SQL Injections and XPath Injections)*
- *XSS (Cross Site Scripting) Injection*
- *LDAP Injection*
- *Command Execution detection (eval(), system(), passtru()...)*
- *CRLF Injection (HTTP Response Splitting, session fixation...)*

USAGE python wapiti.py http://server.com/base/url/ [options]

OPTIONS http://wapiti.sourceforge.net/

EXAMPLE python wapiti.py http://127.0.0.1/vuln/ -c cookies.txt -x http://127.0.0.1/vuln/index.php?page=logout

webscarab

DESCRIPTION **WebScarab** is a framework for analysing applications that communicate using the HTTP and HTTPS protocols. It is written in Java, and is thus portable to many platforms. **WebScarab** has several modes of operation, implemented by a number of plugins. In its most common usage, **WebScarab** operates as an intercepting proxy, allowing the operator to review and modify requests created by the browser before they are sent to the server, and to review and modify responses returned from the server before they are received by the browser. **WebScarab** is able to intercept both HTTP and HTTPS communication. The operator can also review the conversations (requests and responses) that have passed through **WebScarab**.

More info: https://www.owasp.org/index.php/WebScarab_Getting_Started

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webshag-cli

DESCRIPTION **Webshag** is a multi-threaded, multi-platform web server audit tool. Written in Python, it gathers commonly useful functionalities for web server auditing like website crawling, URL scanning or file fuzzing.
Webshag can be used to scan a web server in HTTP or HTTPS, through a proxy and using HTTP authentication (Basic and Digest). In addition to that it proposes innovative IDS evasion functionalities aimed at making correlation between request more complicated (e.g. use a different random per request HTTP proxy server).

The common functionalities of the webshag are

- *Port Scanning*
- *Web Crawling*
- *Url Scanning*
- *Retrieving the list of domain names*
- *File fuzzing*

More info: http://www.scrt.ch/en/attack/downloads/webshag

USAGE ./webshag-cli -m pscan < target-ip >
USAGE ./webshag_cli.py -m spider -p 80 / < target-ip >

EXAMPLE ./webshag_cli.py -m uscan -x -o html -f '/root/Desktop/hackingDNA.html'

webshag-gui

DESCRIPTION **Webshag** is a multi-threaded, multi-platform web server audit tool. Written in Python, it gathers commonly useful functionalities for web server auditing like website crawling, URL scanning or file fuzzing.

Webshag can be used to scan a web server in HTTP or HTTPS, through a proxy and using HTTP authentication (Basic and Digest). In addition to that it proposes innovative IDS evasion functionalities aimed at making correlation between request more complicated (e.g. use a different random per request HTTP proxy server).

The common functionalities of the webshag are

- *Port Scanning*
- *Web Crawling*
- *Url Scanning*
- *Retrieving the list of domain names*
- *File fuzzing*

More info: http://www.scrt.ch/en/attack/downloads/webshag

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

websploit

DESCRIPTION **WebSploit** is used to scan and analysis remote system in order to find various type of vulnerabilities. This tool is very powerful and supports multiple vulnerabilities. **WebSploit** is used for: Social Engineering Works, Scan, Crawler & Web Analysis, Automatic Exploiter and Support Network Attacks.

Features:

- Autopwn - Used From Metasploit For Scan and Exploit Target Service
- wmap - Scan,Crawler Target Used From Metasploit wmap plugin
- format infector - inject reverse & bind payload into file format
- phpmyadmin Scanner
- LFI Bypasser
- Apache Users Scanner
- Dir Bruter
- admin finder
- MLITM Attack - Man Left In The Middle, XSS Phishing Attacks
- MITM - Man In The Middle Attack
- Java Applet Attack
- MFOD Attack Vector
- USB Infection Attack
- ARP Dos Attack
- Web Killer Attack
- Fake Update Attack
- Fake Access point Attack

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

wpscan

DESCRIPTION **WPScan** is a black box WordPress vulnerability scanner.

USAGE wpscan.rb –url [target ip] [options]

OPTIONS http://wpscan.org/

EXAMPLE ruby wpscan.rb --url www.example.com (Do 'non-intrusive' checks...)

EXAMPLE ruby wpscan.rb --url www.example.com --wordlist darkc0de.lst --threads 50 (Do wordlist password brute force on enumerated users using 50 threads...)

EXAMPLE ruby wpscan.rb --url www.example.com --wordlist darkc0de.lst --username admin (Do wordlist password brute force on the 'admin' username only...)

EXAMPLE ruby wpscan.rb --url www.example.com --enumerate p (Enumerate installed plugins...)

EXAMPLE ruby wpscan.rb --url www.example.com –enumerate (Run all enumeration tools...)

xsser

DESCRIPTION Cross Site "Scripter" (aka **XSSer**) is an automatic -framework- to detect, exploit and report XSS vulnerabilities in web-based applications. It contains several options to try to bypass certain filters, and various special techniques of code injection.

USAGE xsser [OPTIONS] [-u |-i |-d] [-g |-p |-c] [Request(s)] [Vector(s)] [Bypasser(s)] [Technique(s)] [Final Injection(s)]

OPTIONS http://xsser.sourceforge.net/

EXAMPLE python xsser.py -u http://host.com (Simple injection from URL)

EXAMPLE python xsser.py -i "file.txt" --proxy "http://127.0.0.1:8118" --referer "666.666.666.666“ (Simple injection from File, with tor proxy and spoofing HTTP Referer headers)

EXAMPLE python xsser.py -u "http://host.com" -p index.php?target=search&subtarget=top&searchstring=" –s (Simple injection from URL, using POST, with statistics results)

EXAMPLE python xsser.py -u "host.com" –hash (Send a pre-checking hash to see if target will generate -false positive- results)

zaproxy

DESCRIPTION The **OWASP Zed Attack Proxy (ZAP)** is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and as such is ideal for developers and functional testers who are new to penetration testing as well as being a useful addition to an experienced pen testers' toolbox.

More info: https://code.google.com/p/zaproxy/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[30] PASSWORD ATTACKS: GPU TOOLS

- oclhashcat-lite
- oclhashcat-plus
- pyrit

oclhashcat-lite

DESCRIPTION **oclhashcat-lite** – world's fastest NTLM, MD5, SHA1, SHA256 and descrypt cracker. **oclHashcat-lite** is a GPGPU cracker that is optimized for cracking performance. Therefore, it is limited to only doing single-hash cracking using Markov attack, Brute-Force attack and Mask attack.

USAGE oclHashcat-lite [options]... hash [mask]

OPTIONS http://hashcat.net/wiki/doku.php?id=oclhashcat_lite

EXAMPLE ./oclHashcat-lite64.bin 9b957cc6ab97cbf88c4f6f0f146adafe

EXAMPLE ./oclHashcat-lite64 -m 1900 -n 80 -1 00010203040506070809 --outfile=out.txt 21B1E417AF2DE6496772BCC2FE33D2593A9BB7A0:003515230478373400 ?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1

EXAMPLE ./oclHashcat-lite64.bin -m 1900 -n 800 -1 00010203040506070809 --outfile=351514044968571.txt --session=35151404496857_1 514D1FCDE9231B61DAD191F7BC7675B87D8628B5:003515140449685700 ?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1

EXAMPLE ./oclHashcat-lite64.bin -m 1900 -n 800 -1 00010203040506070809 --outfile=355933045509554.txt --session=35593304550955_1 B928680D8D7B1242BEBC8B7AC24FF2B90198E213:003559330455095500 ?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1

oclhashcat-plus

DESCRIPTION **oclhashcat-plus** - world's fastest md5crypt, phpass, mscash2 and WPA / WPA2 cracker and world's first and only GPGPU based rule engine. **oclHashcat-plus** is a GPGPU-based multi-hash cracker using a brute-force attack (implemented as mask attack),combinator attack, dictionary attack, hybrid attack, mask attack, and rule-based attack.

USAGE oclHashcat-plus [options]... hash|hashfile|hccapfile [dictionary|mask|directory]...

OPTIONS http://hashcat.net/wiki/doku.php?id=oclhashcat_plus

EXAMPLE oclHashcat-plus64.exe -m 2500 capture.hccap rockyou.txt (Dictionary attack)
EXAMPLE oclhashcat-plus64.exe -m 2500 -a3 capture.hccap ?d?d?d?d?d?d?d?d (Brute-Force Attack)
EXAMPLE oclHashcat-plus64.exe -m 2500 -r rules/best64.rule capture.hccap rockyou.txt (Rule-based attack)

pyrit

DESCRIPTION **Pyrit** allows to create massive databases, pre-computing part of the IEEE 802.11 WPA/WPA2-PSK authentication phase in a space-time-tradeoff. Exploiting the computational power of Many-Core- 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.

USAGE pyrit [options] command

OPTIONS https://code.google.com/p/pyrit/wiki/ReferenceManual

EXAMPLE pyrit -e NETGEAR verify
EXAMPLE pyrit -r /temp/kismet_dump -o small_dump.pcap stripLive
EXAMPLE pyrit -r "large_dumps_*.pcap" -e MyNetwork -o tiny_compressed_dump_MyNetwork.dump.gz strip
EXAMPLE pyrit -r test.pcap -b 00:de:ad:be:ef:00 -i words.txt attack_passthrough
EXAMPLE pyrit -i dirty_words.txt.gz -e NETGEAR -o - passthrough | cowpatty -d - -r wpatestcapture.cap -s NETGEAR
EXAMPLE pyrit -u sqlite:////var/local/pyrit.db relay
EXAMPLE pyrit -u http://192.168.0.100:17934 batch

[31] OFFLINE ATTACKS

- cachedump
- chntpw
- cmospwd
- crunch
- dictstat
- hashcat
- hash-identifier
- john the ripper
- johnny
- lsadump
- maskgen
- oclhashcat-lite
- oclhashcat-plus
- ophcrack
- ophcrack-cli
- policygen
- pwdump
- pyrit
- rainbowcrack
- rcracki_mt
- rsmangler
- samdump2
- sipcrack
- sucrack
- truecrack

cachedump

DESCRIPTION **Cachedump** is great for retrieving the cached Active Directory credentials from XP machines. **CacheDump** will create a CacheDump NT Service to get SYSTEM right and make his stuff on the registry. Then, it will retrieve the LSA Cipher Key to decrypt (rc4/hmac_md5 GloubiBoulga) cache entry values. A **John The Ripper** module has been developed to attack the hashed values that are retrieved (timing equivalent to MD4(MD4(password|U(username))).

USAGE you need patched John The Ripper, then ./john -format:mscash file.txt

OPTIONS: https://github.com/mcandre/fgdump/tree/master/cachedump

EXAMPLE

Cachedump: c:\cachedump.exe user:2d9f0b052932ad18b87f315641921cda:lab:lab.internal
Copy the result in mscash.txt
c:\cachedump.exe –v
Service not found. Installing CacheDump Service (C:\cachedump.exe -s)
CacheDump service successfully installed.
Service started. user:2d9f0b052932ad18b87f315641921cda:lab:lab.internals
Service currently active. Stopping service... Service successfully removed.
John Plugin:
$ ./john -format:mscash ./mscash.txt
Loaded 1 password hash (M$ Cache Hash [mscash])
password (user)

chntpw

DESCRIPTION **chntpw** is a Linux utility to (re)set the password of any user that has a valid (local) account on your WinNT or Win2000 system, by modifying the crypted password in the registry's SAM file. You do not need to know the old password to set a new one. It works offline (i.e., you have to shutdown your computer and boot off a linux floppy disk). The bootdisk includes stuff to access NTFS partitions and scripts to glue the whole thing together. This utility works with SYSKEY and includes the option to turn it off. A bootdisk image is provided.

USAGE chntpw [options] <systemfile> [securityfile] [otherreghive] [...]

OPTIONS: chntpw -h

EXAMPLE chntpw -i sam (starts the program in the interactive mode and specifies the name of the Windows sam file)

cmospwd

DESCRIPTION **CmosPwd** is a cmos/bios password recovery tool.
More info: http://www.cgsecurity.org/cmospwd.txt

USAGE cmospwd [/d]
USAGE cmospwd [/d] /[rlw] cmos_backup_file restore/load/write
USAGE cmospwd /k kill cmos cmospwd /m[01]* execute selected module

EXAMPLE cmospwd /d (to dump cmos in ascii and scan code)
EXAMPLE cmospwd /m0010011 (to execute module 3,6 and 7)

crunch

DESCRIPTION **crunch** is a tool for creating bruteforce wordlists which can be used to audit password strength.

The size of these wordlists is not to be underestimated, however **crunch** can make use of patterns to reduce wordlist sizes, can compress output files in various formats and (since v2.6) now includes a message advising the size of the wordlist that will be created, giving you a 3 second window to stop the creation should the size be too large for your intended use. More info: http://adaywithtape.blogspot.co.uk/2011/05/creating-wordlists-with-crunch-v30.html

USAGE crunch [min length] [max length] [character set] [options]

USAGE ./crunch [min length] [max length] [character set] [options] -o filename.txt (writing to file)

EXAMPLE crunch 8 8 abc + + \!\@\# -t TEST^%,@ -o test.txt

EXAMPLE ./crunch 1 1 -p bird cat dog (creating permutations of lists of words)

EXAMPLE ./crunch 6 6 0123456789ABCDEF

EXAMPLE ./crunch 6 6 ABC\!\@\#\$

EXAMPLE ./crunch 6 6 0123456789 -b 1mb -o START (creating wordlists in blocks of a certain size)

EXAMPLE ./crunch 8 8 abcDEF123 -b 100mb -o START (create a wordlist split in files of no more than 100mb)

EXAMPLE ./crunch 6 6 0123456789 -c 200000 -o START (create files containing no more than 200000 (200 thousand) lines (passphrases))

EXAMPLE ./crunch "123abcDEF " -t TEST@@@@ (include a space in the charset, then enclose the charset in quotes)

dictstat

DESCRIPTION The **dictstat** Python script is a great little tool for password cracking results analysis or for regular wordlist analysis. More info: http://www.question-defense.com/2012/12/16/dictstat-backtrack-5-privilege-escalation-password-attacks-offline-attacks-dictstat

USAGE dictstat.py [options] passwords.txt

OPTIONS

--version show program's version number and exit
-h, --help show this help message and exit
-l 8, --length=8 Password length filter.
-c loweralpha, --charset=loweralpha Password charset filter.
-m stringdigit, --mask=stringdigit Password mask filter
-o masks.csv, --maskoutput=masks.csv Save masks to a file

EXAMPLE ./dictstat.py /root/wordlists/500-worst-passwords.txt
EXAMPLE ./dictstat.py -l 4 /root/wordlists/500-worst-passwords.txt (Password Length Filter)
EXAMPLE ./dictstat.py -c numeric /root/wordlists/500-worst-passwords.txt (Password Charset Filter)
EXAMPLE ./dictstat.py -m stringdigit -o stringdigit.txt /root/wordlists/500-worst-passwords.txt (Password Mask Filter)

hashcat

DESCRIPTION **Hashcat** is the world's fastest CPU-based password recovery tool. While it's not as fast as its GPU counterparts **oclHashcat-plus** and **oclHashcat-lite**, large lists can be easily split in half with a good dictionary and a bit of knowledge of the command switches.

USAGE hashcat [options] hashfile [mask|wordfiles|directories]

OPTIONS http://hashcat.net/wiki/doku.php?id=hashcat

EXAMPLE hashcat-cli64.exe -a 3 –bf-cs-buf abcdefghijklmnopqrstuvwxyz –bf-pw-max 16 -m 0 –o yourfoundpasswords.txt -n 4 –remove yourhashlist.txt (brutecforce)
EXAMPLE hashcat-cli64.exe -a 0 -m 0 -o yourfoundpasswords.txt -n 4 –remove yourhashlist.txt C:\yourwordlist.txt
EXAMPLE hashcat-cli64.exe -a 0 -r rules\best64.rule -m 0 -o yourfoundpasswords.txt -n 4 –remove yourhashlist.txt C:\yourwordlist.txt (rules)

hash-identifier

DESCRIPTION **hash-identifier** is a software to identify the different types of hashes used to encrypt data and especially passwords. More info: https://code.google.com/p/hash-identifier/

USAGE type your hash and get most and least possible hashes

EXAMPLE

Start program:

python ./Hash_ID_v1.1.py

Submit your hash:

e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855

john the ripper

DESCRIPTION **John the Ripper (JTR)** is a free password cracking software tool. It is one of the most popular password testing and breaking programs as it combines a number of password crackers into one package, autodetects password hash types, and includes a customizable cracker. It can be run against various encrypted password formats including several crypt password hash types most commonly found on various Unix flavors (based on DES, MD5, or Blowfish), Kerberos AFS, and Windows NT/2000/XP/2003 LM hash. Additional modules have extended its ability to include MD4-based password hashes and passwords stored in LDAP, MySQL, and others.

USAGE john [OPTIONS] [PASSWORD-FILES]

OPTIONS http://www.osix.net/modules/article/?id=455

EXAMPLE

Save User:gyuJo098KkLy9 into crackme.txt

Then run john with any desirable options

```
john -single crackme.txt (Single crack mode)
john -wordfile:password.lst crackme.txt (dictionary attack)
john -incremental:alpha crackme.txt (only letters; incremental method)
john -incremental:digits crackme.txt (only numbers; incremental method)
john -incremental:lanman crackme.txt (letters, numbers, and some special characters; incremental method)
john -incremental:all crackme.txt (all characters; incremental method)
...
```

johnny

DESCRIPTION **Johnny** is a GUI for **John the Ripper**.
All basic things work well:

- export of cracked passwords through clipboard,
- export works with office suits (tested with LibreOffice Calc),
- user could start, pause and resume attack (though only one session is allowed globally),
- all attack related options work,
- all input file formats are supported (pure hashes, pwdump, passwd, mixed),
- “smart” default options,
- accurate output of cracked passwords,
- smooth work, i.e. no lags,
- config is stored in .conf file (~/.john/johnny.conf),
- nice error messages and other user friendly things,
- many minor fixes to polish ui.

USAGE n/a; GUI tool
EXAMPLE n/a; GUI tool

lsadump

DESCRIPTION **LSADump** is an application that is used to gather Windows password hashes from computers running Windows.

USAGE lsadump $machine.acc [machine]

USAGE ./lsadump.py <system hive> <security hive>

EXAMPLE lsadump -f /home/morgan/Memory\
Images/PhysicalMemory.bin -y 0xe1035b60 -s 0xe77b9b60

maskgen

DESCRIPTION While analyzing passwords using **DictGen** can be both revealing and exciting, it is simply not feasible for larger data sets. **MaskGen** will analyze the masks output file produced by **DictGen** and help you generate optimal password mask collection for input to the **Hashcat** password cracker.

USAGE maskgen [options] masksfile.csv

OPTIONS maksgen –h

TIP

```
[*] [5] [259174/14344391] [1.00] [0d|0h|0m|7s] ?s?u?l?d ?s?u?l?d ...
 \          \                  \            \              \
  \          \                  \            \              \ matching mask
   \          \                  \            \ time to crack
    \          \                  \ percent coverage from sample
     \          \ total number of matching passwords
      \ password length
```

?l - lowercase characters
?u - uppercase characters
?d – digits
?s - special characters

EXAMPLE maskgen.py rockyou.csv

EXAMPLE python maskgen.py --occurrence=10000 rockyou.csv

EXAMPLE maskgen.py --occurrence=100000 --maxtime=8640 rockyou.csv

EXAMPLE maskgen.py --checkmask="?l ?l ?l ?l ?l ?l ?l?d ?l?d" --showmasks rockyou.csv

oclhashcat-lite

DESCRIPTION **oclhashcat-lite** – world's fastest NTLM, MD5, SHA1, SHA256 and descrypt cracker. **oclHashcat-lite** is a GPGPU cracker that is optimized for cracking performance. Therefore, it is limited to only doing single-hash cracking using Markov attack, Brute-Force attack and Mask attack.

USAGE oclHashcat-lite [options]... hash [mask]

OPTIONS http://hashcat.net/wiki/doku.php?id=oclhashcat_lite

EXAMPLE ./oclHashcat-lite64.bin 9b957cc6ab97cbf88c4f6f0f146adafe

EXAMPLE ./oclHashcat-lite64 -m 1900 -n 80 -1 00010203040506070809 --outfile=out.txt 21B1E417AF2DE6496772BCC2FE33D2593A9BB7A0:003515230478373400 ?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1

EXAMPLE ./oclHashcat-lite64.bin -m 1900 -n 800 -1 00010203040506070809 --outfile=351514044968571.txt --session=35151404496857_1 514D1FCDE9231B61DAD191F7BC7675B87D8628B5:003515140449685700 ?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1

EXAMPLE ./oclHashcat-lite64.bin -m 1900 -n 800 -1 00010203040506070809 --outfile=355933045509554.txt --session=35593304550955_1 B928680D8D7B1242BEBC8B7AC24FF2B90198E213:003559330455095500 ?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1?1

oclhashcat-plus

DESCRIPTION **oclhashcat-plus** - world's fastest md5crypt, phpass, mscash2 and WPA / WPA2 cracker and world's first and only GPGPU based rule engine. **oclHashcat-plus** is a GPGPU-based multi-hash cracker using a brute-force attack (implemented as mask attack),combinator attack, dictionary attack, hybrid attack, mask attack, and rule-based attack.

USAGE oclHashcat-plus [options]... hash|hashfile|hccapfile [dictionary|mask|directory]...

OPTIONS http://hashcat.net/wiki/doku.php?id=oclhashcat_plus

EXAMPLE oclHashcat-plus64.exe -m 2500 capture.hccap rockyou.txt (Dictionary attack)
EXAMPLE oclhashcat-plus64.exe -m 2500 -a3 capture.hccap ?d?d?d?d?d?d?d?d (Brute-Force Attack)
EXAMPLE oclHashcat-plus64.exe -m 2500 -r rules/best64.rule capture.hccap rockyou.txt (Rule-based attack)

ophcrack

DESCRIPTION **Ophcrack** is a free open source (GPL licensed) program that cracks Windows 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. On most computers, **ophcrack** can crack most passwords within a few minutes.

Ophcrack is a Windows password cracker based on a time-memory trade-off using rainbow tables.
This is a new variant of Hellman's original trade-off, with better performance.
It recovers 99.9% of alphanumeric passwords in seconds. **Ophcrack** works for Windows NT/2000/XP/Vista.

Ophcrack can be used with command line using the options below, or can be run as a pure graphical software.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

ophcrack-cli

DESCRIPTION **Ophcrack** is a free open source (GPL licensed) program that cracks Windows 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. On most computers, **ophcrack** can crack most passwords within a few minutes.

If you have installed **ophrack-cli** package, graphical interface is not available.

USAGE ophcrack [options]

OPTIONS http://dev.man-online.org/man1/ophcrack-cli/

OPTIONS ophcrack -h

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)

policygen

DESCRIPTION **PolicyGenerator** generates a new reference policy module or updates an existing module based on requested access in the form of access vectors. It generates allow rules and optionally module require statements and reference policy interfaces. By default only allow rules are generated.

PolicyGenerator can also optionally add comments explaining why a particular access was allowed based on the audit messages that generated the access.

USAGE n/a

EXAMPLE n/a

pwdump

DESCRIPTION **pwdump** dumps Windows password hashes. **pwdump** is the name of various Windows programs that output the LM and NTLM password hashes of local user accounts from the Security Account Manager (SAM). In order to work, it must be run under an Administrator account, or be able to access an Administrator account on the computer where the hashes are to be dumped. **Pwdump** could be said to compromise security because it could allow a malicious administrator to access user's passwords.

More info: http://www.foofus.net/~fizzgig/pwdump/

USAGE n/a

EXAMPLE pwdump -u AnAdministrativeUser [-p password] 192.168.0.10

pyrit

DESCRIPTION **Pyrit** allows to create massive databases, pre-computing part of the IEEE 802.11 WPA/WPA2-PSK authentication phase in a space-time-tradeoff. Exploiting the computational power of Many-Core- 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.

USAGE pyrit [options] command

OPTIONS https://code.google.com/p/pyrit/wiki/ReferenceManual

EXAMPLE pyrit -e NETGEAR verify
EXAMPLE pyrit -r /temp/kismet_dump -o small_dump.pcap stripLive
EXAMPLE pyrit -r "large_dumps_*.pcap" -e MyNetwork -o tiny_compressed_dump_MyNetwork.dump.gz strip
EXAMPLE pyrit -r test.pcap -b 00:de:ad:be:ef:00 -i words.txt attack_passthrough
EXAMPLE pyrit -i dirty_words.txt.gz -e NETGEAR -o - passthrough | cowpatty -d - -r wpatestcapture.cap -s NETGEAR
EXAMPLE pyrit -u sqlite:////var/local/pyrit.db relay
EXAMPLE pyrit -u http://192.168.0.100:17934 batch

rainbowcrack

DESCRIPTION **RainbowCrack** is a computer program which generates rainbow tables to be used in password cracking. **RainbowCrack** differs from "conventional" brute force crackers in that it uses large pre-computed tables called rainbow tables to reduce the length of time needed to crack a password drastically. **RainbowCrack** was developed by Zhu Shuanglei, and implements an improved time-memory trade-off cryptanalysis attack which originated in Philippe Oechslin's **Ophcrack**.

USAGE rcrack rainbow_table_pathname -h hash
USAGE rcrack rainbow_table_pathname -l hash_list_file
USAGE rcrack rainbow_table_pathname -f pwdump_file

OPTIONS
rainbow_table_pathname pathname of the rainbow table(s), wildchar(*, ?) supported
-h hash use raw hash as input
-l hash_list_file use hash list file as input, each hash in a line
-f pwdump_file use pwdump file as input, this will handle LAN Manager hash only

EXAMPLE rcrack *.rt -h 5d41402abc4b2a76b9719d911017c592
EXAMPLE rcrack *.rt -l hash.txt
EXAMPLE rcrack *.rt -f hash.txt

rcracki_mt

DESCRIPTION **Rcracki_mt** can be used to perform a rainbow table attack on password hashes. It is intended for indexed & perfected rainbow tables, mainly generated by the distributed project www.freerainbowtables.com

USAGE rcracki_mt [options] [hash]

OPTIONS http://sourceforge.net/project/shownotes.php?release_id=682650

EXAMPLE rcracki_mt -h 5d41402abc4b2a76b9719d911017c592 -t 4 -o save.txt C:\md5
EXAMPLE rcracki_mt -r -s my_personal_hashes

rsmangler

DESCRIPTION **RSMangler** will take a wordlist and perform various manipulations on it similar to those done by **John the Ripper** with a few extras. The main new feature is permutations mode which takes each word in the list and combines it with the others to produce all possible permutations (not combinations, order matters).

USAGE rsmangler.rb [OPTIONS]

OPTIONS http://www.randomstorm.com/rsmangler-security-tool.php

EXAMPLE ./rsmangler.rb –file wordlist.txt > new_wordlist.txt

samdump2

DESCRIPTION **samdump2** dumps Windows 2k/NT/XP/Vista password hashes.
More info: http://www.hackingdna.com/2012/05/learn-samdump-on-backtrack-5.html

USAGE samdump2 samhive keyfile

EXAMPLE ./samdump2 SAM bootkey > hashes
EXAMPLE samdump2 /mnt/Windows/Windows/System32/config/SAM syskey.txt
EXAMPLE samdump2 /mnt/Windows/Windows/System32/config/SAM syskey.txt>hash.txt
EXAMPLE samdump2 /root/sda2/Windows/System32/config/SAM saved-syskey.txt > /root/pass1

EXAMPLE
Example of retrieving the SAM hashes from a Windows partition /dev/sda1:

```
# mkdir -p /mnt/sda1
# mount /dev/sda1 /mnt/sda1
# bkhive /mnt/sda1/Windows/System32/config/SYSTEM /tmp/saved-syskey.txt
# samdump2 /mnt/sda1/Windows/System32/config/SAM /tmp/saved-syskey.txt > /tmp/hashes.txt
```

sipcrack

DESCRIPTION **sipcrack** - a suite of tools to sniff and crack the digest authentications within the SIP protocol.

Session Initiation Protocol (SIP) is a protocol developed by the IETF MMUSIC Working Group and is a proposed standard for initiating, modifying, and terminating an interactive user session that involves multimedia elements such as video, voice, instant messaging, online games, and virtual reality.

SIPcrack *is a SIP login sniffer/cracker that contains 2 programs:* ***sipdump*** *to capture the digest authentication and* ***sipcrack*** *to bruteforce the hash using a wordlist or standard input.* ***sipdump*** *dumps SIP digest authentications. If a login is found, the sniffed login is written to the dump file. See 'sipdump -h' for options.* ***sipcrack*** *bruteforces the user's password with the dump file generated by* ***sipdump****. If a password is found, the sniffed and cracked login will be updated in the dump file. See 'sipcrack -h' for options.*

USAGE sipcrack [options] <dump_file>

OPTIONS

-s Use stdin for passwords

-w wordlist file containing all passwords to try.

-p num print cracking process every n passwords (for -w) (ATTENTION: slows down heavily)

EXAMPLE sipdump -i eth0 logins.dump

sipcrack -w mywordlist.txt logins.dump

sucrack

DESCRIPTION **sucrack** is a multithreaded Linux/UNIX tool for brute-force cracking local user accounts via su. This tool comes in handy as final instance on a system where you have not to many privileges but you are in the wheel group. Many su implementations require a pseudo terminal to be attached in order to take the password from the user. This is why you couldn't just use a simple shell script to do this work. This tool, written in c, is highly efficient and can attempt multiple logins at the same time.

Please be advised that using this tool will take a lot of the CPU performance and fill up the logs quite quickly.

USAGE sucrack [options] <wordlist.txt>

OPTIONS http://www.leidecker.info/projects/sucrack.shtml

EXAMPLE sucrack -s 10 -a wordlist.txt

EXAMPLE sucrack -c -a wordlist.txt

EXAMPLE sucrack -u myuser wordlist.txt

EXAMPLE sucrack -b 50 -w 10 wordlist.txt

EXAMPLE sucrack -r -l AFL wordlist.txt

rule	description	original	rewritten
A	all characters to upper case	myPassword	MYPASSWORD
F	first character to upper case	myPassword	MyPassword
L	last character to upper case	myPassword	myPasswordD
a	all characters to lower case	AnotherPASS	anotherpass
f	first character to lower case	AnotherPASS	anotherPASS
l	last character to lower case	AnotherPASS	AnotherPASs
D	prepend a digit (0..9)	password	1password
d	append a digit (0..9)	password	password1
e	1337ify the word	password	p455w0rd
x	enable all of the above rules		

truecrack

DESCRIPTION **TrueCrack** is a brute-force password cracker for **TrueCrypt** (Copyrigth) volume files. It works on Linux and it is optimized for Nvidia Cuda technology.

It supports:

- *PBKDF2 (defined in PKCS5 v2.0) is based on based on key derivation functions: Ripemd160, Sha512 and Whirlpool.*
- *XTS block cipher mode for hard disk encryption based on AES.*

***TrueCrack** is able to perform a brute-force attack based on:*

- *Dictionary: read the passwords from a file of words.*
- *Alphabet: generate all passwords of given length from given alphabet.*

***TrueCrack** works on gpu and cpu. In gpu, **TrueCrack** requires a lots of resources. We suggest to run **TrueCrack** in a remote session without Xserver and framebuffer.*

USAGE Dictionary attack: truecrack -t <truecrypt_file> -k <ripemd160|sha512|whirlpool> -w <wordlist_file> [-b <parallel_blocks>]

USAGE Alphabet attack: truecrack -t <truecrypt_file> -k <ripemd160|sha512|whirlpool> -c <charset> [-s <minlength>] -m <maxlength> [-b <parallel_block>]

OPTIONS https://code.google.com/p/truecrack/

EXAMPLE ./truecrack –t test_12345678 –w ../../pass.lst -v

[32] ONLINE ATTACKS

- accheck
- burpsuite
- cewl
- cisco-auditing-tool
- dbpwaudit
- findmyhash
- hydra
- hydra-gtk
- medusa
- ncrack
- onesixtyone
- patator
- phrasendrescher
- thc-pptp-bruter
- webscarab
- zaproxy

accheck

DESCRIPTION no info

USAGE no info

EXAMPLE no info

burpsuite

DESCRIPTION **Burp Suite** is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp Suite *contains the following key components:*

- *An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.*
- *An application-aware **Spider**, for crawling content and functionality.*
- *An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.*
- *An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.*
- *A **Repeater** tool, for manipulating and resending individual requests.*
- *A **Sequencer** tool, for testing the randomness of session tokens.*
- *The ability to **save your work** and resume working later.*
- ***Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.*

More info: http://portswigger.net/burp/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

cewl

DESCRIPTION **CeWL** is a ruby app which spiders a given url to a specified depth, optionally following external links, and returns a list of words which can then be used for password crackers such as **John the Ripper**.

By default, **CeWL** sticks to just the site you have specified and will go to a depth of 2 links, this behaviour can be changed by passing arguments. Be careful if setting a large depth and allowing it to go offsite, you could end up drifting on to a lot of other domains. All words of three characters and over are output to stdout. This length can be increased and the words can be written to a file rather than screen so the app can be automated.

USAGE cewl [OPTION] ... URL

OPTIONS http://www.digininja.org/projects/cewl.php

EXAMPLE ./cewl.rb -w passwords.txt http://www.digininja.org/projects/cewl.php (create a password file from http://www.digininja.org/projects/cewl.php and save the password file in passwords.txt)

cisco-auditing-tool

DESCRIPTION **Cisco Auditing Tool** - Perl script which scans cisco routers for common vulnerabilities. Checks for default passwords, easily guessable community names, and the IOS history bug. Includes support for plugins and scanning multiple hosts.

USAGE ./CAT [options]

OPTIONS

-h hostname (for scanning single hosts)
-f hostfile (for scanning multiple hosts)
-p port # (default port is 23)
-w wordlist (wordlist for community name guessing)
-a passlist (wordlist for password guessing)
-i [ioshist] (Check for IOS History bug)
-l logfile (file to log to, default screen)
-q quiet mode (no screen output)

EXAMPLE ./CAT -h 192.168.1.100 -w wordlist -a passwords -i

EXAMPLE ./CAT -h 192.168.1.22 -a lists/passwords -w lists/community (Audit Cisco Telnet Password & SNMP Community String)

dbpwaudit

DESCRIPTION **DBPwAudit** is a Java tool that allows you to perform online audits of password quality for several database engines. The application design allows for easy adding of additional database drivers by simply copying new JDBC drivers to the jdbc directory. Configuration is performed in two files, the aliases.conf file is used to map drivers to aliases and the rules.conf tells the application how to handle error messages from the scan.

The tool has been tested and known to work with:
- Microsoft SQL Server 2000/2005
- Oracle 8/9/10/11
- IBM DB2 Universal Database
- MySQL

USAGE dbpwaudit -s <server> -d <db> -D <driver> -U <users> -P <passwords> [options]

OPTIONS http://www.edwiget.name/2012/07/auditing-mysql-passwords-with-dbpwaudit/

EXAMPLE ./dbpwaudit.sh -s localhost -d mysql -D MySQL -U ~/mysql-users.txt -P ~/mysql-password.txt (Assuming I have a db server on localhost and a list of mysql usernames saved in my home directory as mysql-users.txt and a list of passwords to try also in my home directory as mysql-password.txt, this command would audit the mysql server)

TIP additional steps are required for this program to work: http://www.edwiget.name/2012/07/auditing-mysql-passwords-with-dbpwaudit/

findmyhash

DESCRIPTION **findmyhash.py** attempts to crack different types of hashes using free online services.

USAGE python findmyash.py <algorithm> [OPTIONS]
USAGE findmyash.py <algorithm> [OPTIONS]

OPTIONS

h <hash_value> If you only want to crack one hash, specify its value with this option.

-f <file> If you have several hashes, you can specify a file with one hash per line. NOTE: All of them have to be the same type.

-g If your hash cannot be cracked, search it in Google and show all the results. NOTE: This option ONLY works with -h (one hash input) option.

EXAMPLE python findmyhash.py MD5 -h 098f6bcd4621d373cade4e832627b4f6
EXAMPLE python findmyhash.py MD4 -h "db346d691d7acc4dc2625db19f9e3f52“
EXAMPLE python findmyhash.py SHA224 -h "90a3ed9e32b2aaf4c61c410eb925426119e1a9dc53d4286ade99a809“
EXAMPLE python findmyhash.py LM -h "01fc5a6be7bc6929aad3b435b51404ee“
EXAMPLE python findmyhash.py CISCO7 -h "12090404011C03162E"

hydra

DESCRIPTION **THC-Hydra** is a very fast (multi-threaded) network logon cracker which supports many different services: afp, cisco, cisco-enable, cvs, firebird, ftp, http-get, http-head, http-proxy, https-get, https-head, https-form-get, https-form-post, icq, imap, imap-ntlm, ldap2, ldap3, mssql, mysql, ncp, nntp, oracle-listener, pcanywhere, pcnfs, pop3, pop3-ntlm, postgres, rexec, rlogin, rsh, sapr3, sip, smb, smbnt, smtp-auth, smtp-auth-ntlm, snmp, socks5, ssh2, svn, teamspeak, telnet, vmauthd, vnc.

USAGE hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e ns] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-f] [-s PORT] [-S] [-vV] server service [OPT]

OPTIONS http://www.aldeid.com/wiki/Thc-hydra#Usage

EXAMPLE hydra 127.0.0.1 mysql -l root -P /data/dictionnaires/test.txt -t 4

EXAMPLE hydra 192.168.1.26 ssh2 -s 22 -P pass.txt -L users.txt -e ns -t 10

EXAMPLE hydra 192.168.1.69 http-form-post "/w3af/bruteforce/form_login/dataReceptor.php:user=^USER^&pass=^PASS^:Bad login" -L users.txt -P pass.txt -t 10 -w 30 -o hydra-http-post-attack.txt

hydra-gtk

DESCRIPTION **THC-Hydra** is a very fast (multi-threaded) network logon cracker which supports many different services: afp, cisco, cisco-enable, cvs, firebird, ftp, http-get, http-head, http-proxy, https-get, https-head, https-form-get, https-form-post, icq, imap, imap-ntlm, ldap2, ldap3, mssql, mysql, ncp, nntp, oracle-listener, pcanywhere, pcnfs, pop3, pop3-ntlm, postgres, rexec, rlogin, rsh, sapr3, sip, smb, smbnt, smtp-auth, smtp-auth-ntlm, snmp, socks5, ssh2, svn, teamspeak, telnet, vmauthd, vnc.

Also a GUI tool.

USAGE hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e ns] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-f] [-s PORT] [-S] [-vV] server service [OPT]

OPTIONS http://www.aldeid.com/wiki/Thc-hydra#Usage

EXAMPLE hydra 127.0.0.1 mysql -l root -P /data/dictionnaires/test.txt -t 4

EXAMPLE hydra 192.168.1.26 ssh2 -s 22 -P pass.txt -L users.txt -e ns -t 10

EXAMPLE hydra 192.168.1.69 http-form-post "/w3af/bruteforce/form_login/dataReceptor.php:user=^USER^&pass=^PASS^:Bad login" -L users.txt -P pass.txt -t 10 -w 30 -o hydra-http-post-attack.txt

medusa

DESCRIPTION **Medusa** - Open Source Software 'Login Brute-Forcer' for Password Auditing. Speedy, massively parallel, modular, login brute-forcer" with modules available to support almost any service that allows remote authentication using a password, including: CVS, FTP, HTTP, IMAP, MS-SQL, MySQL, POP3, PostgreSQL, SMTP-AUTH, Telnet and VNC. **Medusa** has been designed to run faster than **Hydra** by using thread-based (rather than Hydra's process-based) parallel testing to attempt to log in to multiple hosts or users concurrently.

More info: http://www.foofus.net/jmk/medusa/medusa.html#how

USAGE [-h host|-H file] [-u username|-U file] [-p password|-P file] [-C file] -M module [OPTIONS]

OPTIONS http://www.irongeek.com/i.php?page=backtrack-r1-man-pages/medusa

EXAMPLE

To use Medusa, the following must be specified:

- The host "192.168.1.1" to connect to, using the -h switch
- The user name "admin" to connect with, using the -u switch
- The name of the textfile containing the list of passwords to try, using the -P switch
- The module to use for the service we are contacting (in this case http) using the -M switch

```
medusa -h 192.168.1.1 -u "admin" -P hugewordlist.txt -M http
```

ncrack

DESCRIPTION **ncrack** — Network authentication cracking tool. 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.

USAGE ncrack [<*Options*>] { <*target specification*> }

OPTIONS http://nmap.org/ncrack/man.html

EXAMPLE ncrack 10.0.0.130:21 192.168.1.2:22

EXAMPLE ncrack scanme.nmap.org 192.168.0.0/8 10.0.0,1,3-7.- -p22 (Ncrack accepts multiple host specifications on the command line, and they don't need to be the same type)

EXAMPLE ncrack scanme.nmap.org:22 ftp://10.0.0.10 ssh://192.168.1.*:5910# (Per-host service specification)

EXAMPLE ncrack scanme.nmap.org 10.0.0.120-122 192.168.2.0/24 -p 22,ftp:3210,telnet (Global service specificatio)

onesixtyone

DESCRIPTION **onesixtyone** takes advantage of the fact that SNMP is a connectionless protocol and sends all SNMP requests as fast as it can. Then the scanner waits for responses to come back and logs them, in a fashion similar to **Nmap** ping sweeps. By default **onesixtyone** waits for 10 milliseconds between sending packets, which is adequate for 100MBs switched networks. The user can adjust this value via the -w command line option. If set to 0, the scanner will send packets as fast as the kernel would accept them, which may lead to packet drop.

USAGE onesixtyone [options] <host> <community>

OPTIONS

-c <communityfile> file with community names to try

-i <inputfile> file with target hosts

-o <outputfile> output log

-d debug mode, use twice for more information

-w <n> wait n milliseconds (1/1000 of a second) between sending packets (default 10)

-q quiet mode, do not print log to stdout, use with –l

EXAMPLE onesixtyone 192.168.100.51

patator

DESCRIPTION **Patator** is a multi-purpose brute-forcer, with a modular design and a flexible usage.
More info: https://code.google.com/p/patator/

USAGE python patator.py <module> -h
USAGE <module> -h (if you created the shortcuts)

EXAMPLE patator.py ftp_login host=10.0.0.1 user=FILE0 password=qsdf 0=logins.txt -x ignore:mesg='Login incorrect.' (FTP : Enumerate valid logins on a too verbose server)

phrasendrescher

DESCRIPTION **phrasen|drescher** is a cracking tool used for the purpose of finding the pass phrase for RSA or DSA keys as they would be used by SSH for instance. It performs wordlist and rule based attacks against the key.
More info: http://leidecker.info/projects/phrasendrescher.shtml

USAGE Incremental mode: phrasendrescher -i 6:8 key-file
USAGE Incremental mode: phrasendrescher -i 8 key-file (generating 8 characters long words)
USAGE Incremental mode: phrasendrescher -i 8:12 key-file (specify range)
USAGE Dictionary mode: phrasendrescher -d wordlist key-file
USAGE Dictionary mode: phrasendrescher -d wordlist directory-containing-keys (read and try multiple keys if you specify a directory instead of a single key file)

EXAMPLE ./phrasendrescher -vd wordlist.txt my.key

thc-pptp-bruter

DESCRIPTION a brute force program that works against pptp vpn endpoints. The use of the tool is pretty straightforward: just pipe a dictionary file into the **thc-pptp-bruter** and specify both the username and the host you are attacking. Note that upon connecting to the device, you would see some brief information about the host to which you are connecting, such as "Hostname ˜c2611wooter, Vendor ˜Cisco Systems, Inc., Firmware: 4608." This is a useful method of remote application layer fingerprinting.

More info: http://flylib.com/books/en/3.418.1.83/1/

USAGE thc-pptp-brute [options] <remote ip>

OPTIONS

-v Verbose output / Debug output
-W Disable windows hack [default: enabled]
-u <user> User [default: administrator]
-w <file> Wordlist file [default: stdin]
-p < > PPTP port [default: 1723]
-n < > Number of parallel tries [default: 5]
-l < > Limit to n passwords / sec [default: 100]

EXAMPLE thc-pptp-bruter -u g0tmi1k -n 99 -l 999 10.0.0.3

webscarab

DESCRIPTION **WebScarab** is a framework for analysing applications that communicate using the HTTP and HTTPS protocols. It is written in Java, and is thus portable to many platforms. **WebScarab** has several modes of operation, implemented by a number of plugins. In its most common usage, **WebScarab** operates as an intercepting proxy, allowing the operator to review and modify requests created by the browser before they are sent to the server, and to review and modify responses returned from the server before they are received by the browser. **WebScarab** is able to intercept both HTTP and HTTPS communication. The operator can also review the conversations (requests and responses) that have passed through **WebScarab**.

More info: https://www.owasp.org/index.php/WebScarab_Getting_Started

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

zaproxy

DESCRIPTION The **OWASP Zed Attack Proxy (ZAP)** is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and as such is ideal for developers and functional testers who are new to penetration testing as well as being a useful addition to an experienced pen testers' toolbox.

More info: https://code.google.com/p/zaproxy/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[33] WIRELESS ATTACKS: BLUETOOTH TOOLS

- bluelog
- bluemaho
- blueranger
- btscanner
- fang
- spooftooph

bluelog

DESCRIPTION **Bluelog** is a Bluetooth scanner designed to tell you how many discoverable devices there are in an area as quickly as possible. It is intended to be used as a site survey tool, identifying the number of possible **Bluetooth** targets there are in the surrounding environment.

USAGE bluelog [options]

OPTIONS https://github.com/MS3FGX/Bluelog

EXAMPLE bluelog –vtn (basic scanning - this will turn on verbose output, timestamps, device names, and output to the default log file)

bluemaho

DESCRIPTION **BlueMaho** is GUI-shell (interface) for suite of tools for testing security of Bluetooth devices. It is freeware, opensource, written on python, uses wxPyhon. It can be used for testing BT-devices for known vulnerabilities and major thing to do - testing to find unknown vulnerabilities. Also it can form nice statistics.
More info: http://wiki.thc.org/BlueMaho

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

blueranger

DESCRIPTION **BlueRanger** is a simple Bash script which uses *Link Quality* to locate Bluetooth device radios. It sends l2cap (Bluetooth) pings to create a connection between Bluetooth interfaces, since most devices allow pings without any authentication or authorization. The higher the link quality, the closer the device (in theory).

Use a Bluetooth Class 1 adapter for long range location detection. Switch to a Class 3 adapter for more precise short range locating. The precision and accuracy depend on the build quality of the Bluetooth adapter, interference, and response from the remote device. Fluctuations may occur even when neither device is in motion.

USAGE blueranger.sh [options] <device address>

EXAMPLE blueranger.sh hci0 6C:D6:8A:B1:30:BC (Provide the local interface and Device Address of the device you are trying to locate.)

btscanner

DESCRIPTION **btscanner** is a tool designed specifically to extract as much information as possible from a Bluetooth device without the requirement to pair. A detailed information screen extracts HCI and SDP information, and maintains an open connection to monitor the RSSI and link quality. **btscanner** is based on the BlueZ Bluetooth stack, which is included with recent Linux kernels, and the BlueZ toolset. **btscanner** also contains a complete listing of the IEEE OUI numbers and class lookup tables. Using the information gathered from these sources it is possible to make educated guesses as to the host device type.

USAGE btscanner [options]

OPTIONS

--help Display help

--cfg=<file> Use <file> as the config file

--no-reset Do not reset the Bluetooth adapter before scanning

EXAMPLE btscanner --help

fang

DESCRIPTION n/a

USAGE n/a

EXAMPLE n/a

spooftooph

DESCRIPTION **Spooftooph** is designed to automate spoofing or cloning Bluetooth device Name, Class, and Address. Cloning this information effectively allows Bluetooth device to hide in plain site. Bluetooth scanning software will only list one of the devices if more than one device in range shares the same device information when the devices are in *Discoverable Mode* (specifically the same Address).

Option 1: Continuously scan an area for Bluetooth devices. Make a selection on which device in the list to clone. This option also allows for logging of the scanned devices.
Option 2: Randomly generate and assign valid Bluetooth interface information. The *class* and *address* are randomly generated and the *name* is derived from a list of the top 100 most common names in US and the type of device. For example if the randomly generated *class* is a phone, SpoofTooph might generate the name "Bob's Phone".
Option 3: Specify the *name, class,* and *address* a user wishes for the Bluetooth interface to have.
Option 4: Read in the log of previous scans and select a device to clone. Users can also manually add Bluetooth profiles to these log files.
Option 5: *Incognito mode.* Scan for and clone new devices at user assigned intervals.

USAGE spooftooph [options] <device address>

EXAMPLE spooftooph -i hci0 -n new_name -a 00:11:22:33:44:55 -c 0x1c010c (Specify NAME, CLASS and ADDR)

EXAMPLE spooftooph -i hci0 –R (Randomly generate NAME, CLASS and ADDR)

EXAMPLE spooftooph -i hci0 -s -w file.csv (Scan for devices in range and select device to clone. Optionally dump the device information in a specified log file)

EXAMPLE spooftooph -i hci0 -r file.csv (Load in device info from log file and specify device info to clone)

EXAMPLE spooftooph -i hci0 -t 1 (Clone a random devices info in range every X seconds)

[34] OTHER WIRELESS TOOLS

- zbassocflood
- zbconvert
- zbdsniff
- zbdump
- zbfind
- zbgoodfind
- zbid
- zbreplay
- zbstumbler

zbassocflood

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbassocflood - Repeatedly associate to the target PANID in an effort to cause the device to crash from too many connected stations.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbconvert

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbconvert - Convert a packet capture from Libpcap to Daintree SNA format, or vice-versa.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbdsniff

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbdsniff - Captures ZigBee traffic, looking for NWK frames and over-the-air key provisioning. When a key is found, **zbdsniff** prints the key to stdout. The sample packet capture sample/zigbee-network-key-ota.dcf can be used to demonstrate this functionality.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbdump

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbdump - A tcpdump-like took to capture IEEE 802.15.4 frames to a libpcap or Daintree SNA packet capture file. Does not display real-time stats like tcpdump when not writing to a file.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbfind

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbfind - A GTK GUI application for tracking the location of an IEEE 802.15.4 transmitter by measuring RSSI. **Zbfind** can be passive in discovery (only listen for packets) or it can be active by sending Beacon Request frames and recording the responses from ZigBee routers and coordinators. If you get a bunch of errors after starting this tool, make sure your DISPLAY variable is set properly. If you know how to catch these errors to display a reasonable error message, please drop me a note.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbgoodfind

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbgoodfind - Implements a key search function using an encrypted packet capture and memory dump from a legitimate ZigBee or IEEE 802.15.4 device. This tool accompanies Travis Goodspeed's GoodFET hardware attack tool, or other binary data that could contain encryption key information such as bus sniffing with legacy chips (such as the CC2420). **Zbgoodfind's** search file must be in binary format (obj hexfile's are not supported). To convert from the hexfile format to a binary file, use the objcopy tool: objcopy -I ihex -O binary mem.hex mem.bin.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbid

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbid - Identifies available interfaces that can be used by KillerBee and associated tools.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbreplay

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbreplay - Implements a replay attack, reading from a specified Daintree DCF or libpcap packet capture file, retransmitting the frames. ACK frames are not retransmitted.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

zbstumbler

DESCRIPTION **KillerBee** includes several tools designed to attack ZigBee and IEEE 802.15.4 networks, built using the **KillerBee** framework. Each tool has its own usage instructions documented by running the tool with the "-h" argument, and summarized below.

zbstumbler - Active ZigBee and IEEE 802.15.4 network discovery tool. **Zbstumbler** sends beacon request frames out while channel hopping, recording and displaying summarized information about discovered devices. Can also log results to a CSV file.

The **KillerBee** API is documented in epydoc format, with HTML documentation in the doc/ directory of this distribution. If you have epydoc installed, you can also generate a convenient PDF for printing, if desired, as shown: $ cd **killerbee** $ mkdir pdf $ epydoc --pdf -o pdf killerbee/ The pdf/ directory will have a file called "**api.pdf**" which includes the framework documentation.

USAGE n/a

EXAMPLE n/a

[35] RFID/NFC TOOLS: NFC TOOLS

- mfcuk
- mfoc
- mifare-classic-format
- nfc-list
- nfc-mfclassic

mfcuk

DESCRIPTION **MFCUK - MiFare Classic Universal toolKit.**

Toolkit containing samples and various tools based on and around libnfc and crapto1, with emphasis on Mifare Classic NXP/Philips RFID cards.

Special emphasis of the toolkit is on the following:

- mifare classic weakness demonstration/exploitation
- demonstrate use of libnfc (and ACR122 readers)
- demonstrate use of Crapto1 implementation to confirm internal workings and to verify theoretical/practical weaknesses/attacks

USAGE ./mfcuk [OPTIONS]

EXAMPLE ./mfcuk_keyrecovery_darkside -C -R 0:A -v 2 (Recovering a key)

mfoc

DESCRIPTION **MFOC** is an open source implementation of "offline nested" attack by Nethemba. **MFOC** uses LIBNFC and CRAPTO1 library to recover the keys, provided at least one valid Key-A/Key-B of any sector is known, or if the card uses a default key. If a card uses at least one block encrypted with a default key, all the other keys can be extracted in minutes. If the card does not use default keys, one key for a sector can be retrieved using the **MFCUK** library, after which this library can be used.

This program allows to recover authentication keys from MIFARE Classic card.

Please note MFOC is able to recover keys from target only if it have a known key: default one (hardcoded in MFOC) or custom one (user provided using command line).

USAGE mfoc [-h] [-k key]... [-P probnum] [-T tolerance] [-O output]

EXAMPLE mfoc –h

EXAMPLE mfoc -O mycard.mfd

EXAMPLE mfoc -k ffffeeeedddd -O mycard.mfd

EXAMPLE mfoc -P 50 -T 30 -O mycard.mfd

mifare-classic-format

DESCRIPTION **MifareClassicTool -** An Android NFC-App for reading/writing/analysing/etc Mifare Classic RFID-Tags.

USAGE mifare-classic-format [option]

OPTIONS
-h Help
-f Fast format (only erase MAD)
-y Do not ask for confirmation (dangerous)

EXAMPLE mifare-classic-format -h

nfc-list

DESCRIPTION **libnfc** is a library which allows userspace application access to NFC devices.

nfc-list is part of **libnfc** .

nfc-list is a utility for listing any available tags like ISO14443-A, FeliCa, Jewel or ISO14443-B (according to the device capabilities). It may detect several tags at once thanks to a mechanism called anti-collision but all types of tags don't support anti-collision and there is some physical limitation of the number of tags the reader can discover.

This tool displays all available information at selection time.

More info: http://nfc-tools.org/index.php?title=Main_Page

USAGE nfc-list

EXAMPLE nfc-list

nfc-mfclassic

DESCRIPTION **libnfc** is a library which allows userspace application access to NFC devices.
nfc-mfclassic is part of **libnfc** .

With this tool a complete MIFARE card can be dumped to / restored from a MIFARE dump file (*.mfd).

More info: http://nfc-tools.org/index.php?title=Libnfc:nfc-mfclassic and http://nfc-tools.org/index.php?title=Main_Page

USAGE nfc-mfclassic [Options] <dump file>
OPTIONS https://code.google.com/p/libnfc/source/browse/utils/nfc-mfclassic.1?r=03a6f5e29c9a177788c20470b71128938d90120c

EXAMPLE nfc-mfclassic r X ~/Desktop/dump.mfd

[36] RFIDIOT A CG

- brute force hitag2
- bruteforce mifare
- calculate jcop mifare keys
- continuous select tag
- copy iso15693b tag
- epassport read write clone
- format mifare 1k value blocks
- identify hf tag type
- identify if tag type
- jcop info
- jcop mifare read write
- jcop set atr historical bytes
- read acg reader eeprom
- read if tag
- read mifare
- read tag
- read write clone unique (em4x02)
- reset q5 tag
- select tag
- set fdx-b id
- test acg lahf

brute force hitag2

DESCRIPTION no info

USAGE no info

EXAMPLE no info

bruteforce mifare

DESCRIPTION no info

USAGE no info

EXAMPLE no info

calculate jcop mifare keys

DESCRIPTION text

USAGE text

EXAMPLE text

continuous select tag

DESCRIPTION text

USAGE text

EXAMPLE text

copy iso15693b tag

DESCRIPTION text

USAGE text

EXAMPLE text

epassport read write clone

DESCRIPTION text

USAGE text

EXAMPLE text

format mifare 1k value blocks

DESCRIPTION text

USAGE text

EXAMPLE text

format mifare 1k value blocks

DESCRIPTION text

USAGE text

EXAMPLE text

identify hf tag type

DESCRIPTION text

USAGE text

EXAMPLE text

identify if tag type

DESCRIPTION text

USAGE text

EXAMPLE text

jcop info

DESCRIPTION text

USAGE text

EXAMPLE text

jcop mifare read write

DESCRIPTION text

USAGE text

EXAMPLE text

jcop set atr historical bytes

DESCRIPTION text

USAGE text

EXAMPLE text

read acg reader eeprom

DESCRIPTION text

USAGE text

EXAMPLE text

read if tag

DESCRIPTION text

USAGE text

EXAMPLE text

read mifare

DESCRIPTION text

USAGE text

EXAMPLE text

read tag

DESCRIPTION text

USAGE text

EXAMPLE text

read write clone unique (em4x02)

DESCRIPTION text

USAGE text

EXAMPLE text

reset q5 tag

DESCRIPTION text

USAGE text

EXAMPLE text

select tag

DESCRIPTION text

USAGE text

EXAMPLE text

set fdx-b id

DESCRIPTION text

USAGE text

EXAMPLE text

test acg lahf

DESCRIPTION text

USAGE text

EXAMPLE text

[37] RFIDIOT FROSCH

- read write clone unique (em4x02)
- reset hitag2 tag
- set fdx-b id
- test frosch reader

read write clone unique (em4x02)

DESCRIPTION text

USAGE text

EXAMPLE text

reset hitag2 tag

DESCRIPTION text

USAGE text

EXAMPLE text

set fdx-b id

DESCRIPTION text

USAGE text

EXAMPLE text

test frosch reader

DESCRIPTION text

USAGE text

EXAMPLE text

[38] RFIDIOT PCSC

- bruteforce mifare
- calculate jcop mifare keys
- chip & pin info
- continuous select tag
- epassport read write clone
- identify hf tag type
- jcop info
- jcop mifare read write
- jcop set atr historical bytes
- read mifare
- read tag
- select tag

bruteforce mifare

DESCRIPTION text

USAGE text

EXAMPLE text

calculate jcop mifare keys

DESCRIPTION text

USAGE text

EXAMPLE text

chip & pin info

DESCRIPTION text

USAGE text

EXAMPLE text

continuous select tag

DESCRIPTION text

USAGE text

EXAMPLE text

epassport read write clone

DESCRIPTION text

USAGE text

EXAMPLE text

identify hf tag type

DESCRIPTION text

USAGE text

EXAMPLE text

jcop info

DESCRIPTION text

USAGE text

EXAMPLE text

jcop mifare read write

DESCRIPTION text

USAGE text

EXAMPLE text

jcop set atr historical bytes

DESCRIPTION text

USAGE text

EXAMPLE text

read mifare

DESCRIPTION text

USAGE text

EXAMPLE text

read tag

DESCRIPTION text

USAGE text

EXAMPLE text

select tag

DESCRIPTION text

USAGE text

EXAMPLE text

[39] WIRELESS TOOLS

- aircrack-ng
- aireplay-ng
- airmon-ng
- airodump-ng
- asleap
- cowpatty
- eapmd5pass
- fern-wifi-cracker
- genkeys
- genpmk
- giskismet
- kismet
- mdk3
- wifiarp
- wifidns
- wifi-honey
- wifiping
- wifitap
- wifite

aircrack-ng

DESCRIPTION **Aircrack-ng** is an 802.11 WEP and WPA-PSK keys cracking program that can recover keys once enough data packets have been captured. **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.

More info: www.**aircrack-ng**.org/

USAGE **aircrack-ng** [options] <.cap / .ivs file(s)>

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/aircrack-ng.1.html

EXAMPLE aircrack-ng -a 2 -w dictionary.txt handshake-01.cap

aireplay-ng

DESCRIPTION **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. **aireplay-ng** supports single-NIC injection/monitor. This feature needs driver patching.
More info: www.**aircrack-ng**.org/

USAGE **aireplay-ng** [options] <replay interface>

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man8/aireplay-ng.8.html

EXAMPLE aireplay-ng -0 5 -a 00:1D:7E:56:FD:F6 -c 00:1A:73:D7:CA:88 mon0

airmon-ng

DESCRIPTION **airmon-ng** is 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 list/kill programs that can interfere with the wireless card and set the right sources in /etc/kismet/kismet.conf too.

More info: www.**aircrack-ng**.org/

USAGE **airmon-ng** <start|stop> <interface> [channel] **airmon-ng** <check> [kill]

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man8/airmon-ng.8.html

EXAMPLE airmon-ng start mon0

airodump-ng

DESCRIPTION **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.

More info: www.**aircrack-ng**.org/

USAGE **airodump-ng** [options] <interface name>

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man8/airodump-ng.8.html

EXAMPLE airodump-ng -c 10 --bssid 00:1D:7E:56:FD:F6 --showack -w handshake mon0

EXAMPLE **airodump-ng** --band bg ath0

asleap

DESCRIPTION **asleap** - recovers weak LEAP password. This tool is released as a proof-of-concept to demonstrate weaknesses in the LEAP and PPTP protocols.

LEAP is the Lightweight Extensible Authentication Protocol, intellectual property of Cisco Systems, Inc. LEAP is a security mechanism available only on Cisco access points to perform authentication of end-users and access points. LEAP is written as a standard EAP-type, but is not compliant with the 802.1X specification since the access point modifies packets in transit, instead of simply passing them to a authentication server (e.g. RADIUS).

PPTP is a Microsoft invention for deploying virual private networks (VPN). PPTP uses a tunneling method to transfer PPP frames over an insecure network such as a wireless LAN. RFC 2637 documents the operation and functionality of the PPTP protocol.

USAGE asleap [options]

OPTIONS http://www.willhackforsushi.com/code/asleap/2.2/README

EXAMPLE ./asleap –r leap.dump –f dict.dat –n dict.idx

TIP using **asleap** with **genkeys**: http://wirelessdefence.org/Contents/AsleapMain.htm

cowpatty

DESCRIPTION **coWPAtty** - brute-force dictionary attack against WPA-PSK. **coWPAtty** is designed to audit the pre-shared key (PSK) selection for WPA networks based on the TKIP protocol.

```
Note that it is also possible to mount a precomputed attack against the PSK.
The PBKDF2 algorithm used to generate the PMK takes two non-fixed inputs: the
passphrase and the network SSID.  For a given SSID, we can precompute all the
PMK's from a dictionary file with the "genpmk" tool:

$ ./genpmk
genpmk 1.0 - WPA-PSK precomputation attack. <jwright@hasborg.com>
genpmk: Must specify a dictionary file with -f
Usage: genpmk [options]

-f                  Dictionary file
-d                  Output hash file
-s                  Network SSID
-h                  Print this help information and exit
-v                  Print verbose information (more -v for more verbosity)
-V                  Print program version and exit

After precomputing the hash file, run cowpatty with the -d argument.
$ ./genpmk -f dict -d hashfile -s somethingclever
```

USAGE cowpatty [options]

OPTIONS http://www.willhackforsushi.com/code/cowpatty/4.3/README

EXAMPLE ./cowpatty –r wpa2psk-linksys.dump –d linksys.hash –s links

EXAMPLE ./cowpatty -r eap-test.dump -f dict -s somethingclever (if you are auditing WPA-PSK or WPA2-PSK networks, you can use this tool to identify weak passphrases that were used to generate the PMK. Supply a libpcap capture file that includes the 4-way handshake, a dictionary file of passphrases to guess with, and the SSID for the network)

EXAMPLE john -wordfile:dictfile -rules -session:johnrestore.dat -stdout:63 | \ cowpatty -r eap-test.dump -f - -s somethingclever (accept dictionary words from STDIN, allowing us to utilize a tool such as John the Ripper to create lots of word permutations from a dictionary file)

EXAMPLE ./cowpatty -r eap-test.dump -d hashfile -s somethingclever

eapmd5pass

DESCRIPTION EAP-MD5 is a legacy authentication mechanism that does not provide sufficient protection for user authentication credentials. Users who authenticate using EAP-MD5 subject themselves to an offline dictionary attack vulnerability.

This tool reads from a live network interface in monitor-mode, or from a stored libpcap capture file, and extracts the portions of the EAP-MD5 authentication exchange. Once the challenge and response portions have been collected from this exchange, **eapmd5pass** will mount an offline dictionary attack against the user's password.

This utility implements a dictionary attack against the EAP-MD5 protocol. With an EAP-MD5 authentication capture, you can audit the password for a given user, or specify the EAP-MD5 authentication parameters on the command-line to audit any EAP-MD5 exchange.

USAGE eapmd5pass [-I <int> | -r <pcapfile>] [-w worfile] [options]

OPTIONS http://www.willhackforsushi.com/code/eapmd5pass/1.4/README

EXAMPLE ./eapmd5pass -w dict -r eapmd5-sample.dump

fern-wifi-cracker

DESCRIPTION **Fern Wifi Cracker** is a Wireless security auditing and attack software program that is able to crack and recover WEP/WPA/WPS keys and also run other network based attacks on wireless or Ethernet based networks. More info: https://code.google.com/p/fern-wifi-cracker/

Features

- *Fern Wifi Cracker* currently supports the following features:*WEP Cracking with Fragmentation,Chop-Chop, Caffe-Latte, Hirte, ARP Request Replay or WPS attack*
- *WPA/WPA2 Cracking with Dictionary or WPS based attacks*
- *Automatic saving of key in database on successful crack*
- *Automatic Access Point Attack System*
- *Session Hijacking (Passive and Ethernet Modes)*
- *Access Point MAC Address Geo Location Tracking*
- *Internal MITM Engine*
- *Bruteforce Attacks (HTTP,HTTPS,TELNET,FTP)*
- *Update Support*

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

genkeys

DESCRIPTION **genkeys** - generates the database and index files to use with **asleap** lookups.

USAGE genkeys [options] (must supply -r -f and -n)

OPTIONS

-r	Input dictionary file, one word per line
-f	Output pass+hash filename
-n	Output index filename
-h	Last 2 hash bytes to filter with (optional)

EXAMPLE ./genkeys -r dict -f dict.dat -n dict.idx

genpmk

DESCRIPTION **genpmk** is used to precompute the hash files in a similar way to Rainbow tables is used to pre-hash passwords in Windows LANMan attacks. There is a slight difference, however, in WPA in that the SSID of the network is used as well as the WPA-PSK to "salt" the hash. This means that we need a different set of hashes for each and every unique SSID i.e. a set for "linksys" a set for "tsunami" etc..

USAGE genpmk [options]

OPTIONS

-f	Dictionary file
-d	Output hash file
-s	Network SSID
-h	Print this help information and exit
-v	Print verbose information (more -v for more verbosity)
-V	Print program version and exit

EXAMPLE ./genpmk -f dict -d hashfile -s cuckoo (generate some hash files for a network using the SSID cuckoo)

giskismet

DESCRIPTION **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. **GISKismet** currently uses SQLite for the database and GoogleEarth / KML files for graphing.

USAGE giskismet [Options]

OPTIONS http://www.irongeek.com/i.php?page=backtrack-r1-man-pages/giskismetp

EXAMPLE perl giskismet -x examples/Kismet-Feb-05-2009-1.netxml (Insert all data from a Kismet-newcore netxml file into the GISKismet database)

EXAMPLE perl giskismet -x examples/Kismet-Feb-05-2009-1.netxml --channel 2 (Insert only the APs on channel 2)

EXAMPLE perl giskismet -q "select * from wireless" -o ex1.kml (Generate a graph based on the GISKismet database) The ex1.kml file can be found at:

wget -O ex1.kml \ "http://my-trac.assembla.com/giskismet/browser/trunk/examples/ex1.kml?format=raw"

EXAMPLE perl giskismet -x examples/Kismet-Feb-05-2009-1.netxml \ -q "select * from wireless where ESSID='linksys' and Encryption='None'" -o ex2.kml (Insert all the information from a Kismet-newcore netxml file and generate a graph of the APs named linksys without encryption) The ex2.kml file can be found at:

wget -O ex2.kml \ "http://my-trac.assembla.com/giskismet/browser/trunk/examples/ex2.kml?format=raw"

kismet

DESCRIPTION **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. **Kismet** identifies networks by passively collecting packets and detecting standard named networks, detecting (and given time, decloaking) hidden networks, and inferring the presence of nonbeaconing networks via data traffic.
kismet supports logging to the wtapfile packet format (readable by tcpdump and ethereal) and saves detected network informat as plaintext, CSV, and XML. **kismet** is capable of using any GPS supported by **gpsd** and logs and plots network data.
kismet *is divided into three basic programs,* ***kismet_server kismet_client*** *and* ***gpsma***

USAGE *kismet* [server-options] [-] [client-options]
USAGE *kismet_server* [**-nqs**] [**-t** *title*] [**-f** *config-file*] [**-c** *capture-source*] [**-C** *enable-capture-sources*] [**-l** *log-types*] [**-d** *dump-type*] [**-m** *max-packets-per-file*] [**-g** *gpshost:port*] [**-p** *listen-port*] [**-a***allowed-hosts*] [**-N** *server-name*]
USAGE *kismet_client* [**-qr**] [**-f** *config-file*] [**-s** *serverhost:port*] [**-g** *gui-type*] [**-c** *display-columns*]
OPTIONS http://www.irongeek.com/i.php?page=backtrack-r1-man-pages/kismet

EXAMPLE n/a

mdk3

DESCRIPTION **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.*

Features:

- *Bruteforce MAC Filters*
- *Bruteforce hidden SSIDs (some small SSID wordlists included)*
- *Probe networks to check if they can hear you*
- *intelligent Authentication-DoS to freeze APs (with success checks)*
- *FakeAP - Beacon Flooding with channel hopping (can crash NetStumbler and some buggy drivers)*
- *Disconnect everything (aka AMOK-MODE) with Deauthentication and Disassociation packets*
- *WPA TKIP Denial-of-Service*
- *WDS Confusion - Shuts down large scale multi-AP installations*

USAGE mdk3 <interface> <test_mode> [test_options]

OPTIONS http://hack-it.org/index.php?title=Mdk3

EXAMPLE mdk3 –fullhelp (for all test options)

wifiarp

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

wifidns

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

wifi-honey

DESCRIPTION **wifi-honey** works out what encryption a client is looking for in a given network by setting up four fake access points, each with a different type of encryption - None, WEP, WPA and WPA2 - and then observing which of the four the client connects to.

In the case of WPA/WPA2, by running **airodump-ng** along side this you also end up capturing the first two packets of the four way handshake and so can attempt to crack the key with either **aircrack-ng** or **coWPAtty**.

What this script does - is to automate the setup process, it creates five monitor mode interfaces, four are used as APs and the fifth is used for **airodump-ng**. To make things easier, rather than having five windows all this is done in a screen session which allows you to switch between screens to see what is going on. All sessions are labelled so you know which is which.

USAGE ./wifi_honey.sh <essid> <channel> <interface>

USAGE ./wifi_honey.sh fake_wpa_net (start the script with the ESSID of the network you want to impersonate)

USAGE ./wifi_honey.sh fake_wpa_net 1 wlan1 (You can also specify the channel to use and the interface you want to base the whole lot on)

EXAMPLE ./wifi_honey.sh THECRIB 11 wlan2

wifiping

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

wifitap

DESCRIPTION **Wifitap** is a proof of concept for communication over WiFi networks using traffic injection.

Wifitap allows direct communication with an associated station to a given access point directly, meaning: not being associated ourselves; not being handled by access point. More info: http://sid.rstack.org/static/articles/w/i/f/Wifitap_EN_9613.html

USAGE wifitap -b <BSSID> [-o <iface>] [-i <iface>] [-s <SMAC>] [-w <WEP key> [-k <key id>]] [-d [-v]] [-h]

OPTIONS

```
-b <BSSID>    specify BSSID for injection
-o <iface>    specify interface for injection (default: ath0)
-i <iface>    specify interface for listening (default: ath0)
-s <SMAC>     specify source MAC address
-w <key>      WEP mode and key
-k <key id>   WEP key id (default: 0)
-d            activate debug
-v            verbose debugging
-h            this so helpful output
```

EXAMPLE wifitap.py -b 00:13:10:30:22:5C -i eth1 -p -o eth1

wifite

DESCRIPTION **Wifite** is a python script which automates the WEP and WPA cracking process with **aircrack-ng** tools.

TIP Wifite can and will delete certain existing .CAP and .XOR files inside of the directory it is run; specifically any *.XOR files and replay-*.cap files. Please move wifite.py into its own directory to avoid the deleting of these kinds of files.

TIP Before you run **wifite**, please learn and use the command-line tools available with **aircrack-ng**. Here is an easy guide to WEP cracking and here is an easy guide to WPA cracking. Only after you have tested and successfully cracked WEP and WPA without the use of an automated tool should you use **Wifite**.

USAGE *python wifite.py [SETTINGS] [FILTERS]*

OPTIONS python wifite.py –help and http://wifite.googlecode.com/svn-history/r5/trunk/wifite.py

EXAMPLE ./wifite.py -all –wepto (to crack all WEP access points)

EXAMPLE ./wifite.py -p 50 –wpsto (crack all WPS access points with signal strength greater than (or equal to) 50dB)

EXAMPLE ./wifite.py -all --dict /pentest/passwords/wordlists/darkc0de.lst (attack all access points, use 'darkc0de.lst' for cracking WPA handshakes)

EXAMPLE ./wifite.py -all -wpa --dict none (to attack all WPA access points, but do not try to crack -- any captured handshakes are saved automatically)

EXAMPLE ./wifite.py --pow 50 -wept 300 -pps 600 (to crack all WEP access points greater than 50dB in strength, giving 5 minutes for each WEP attack method, and send packets at 600 packets/sec)

EXAMPLE ./wifite.py -e "2WIRE752" -wept 0 (to crack all WEP access points greater than 50dB in strength, giving 5 minutes for each WEP attack method, and send packets at 600 packets/sec)

[40] EXPLOITATION TOOLS: CISCO ATTACKS

- cisco-auditing-tool
- cisco-global-exploiter
- cisco-ocs
- cisco-torch
- yersinia

cisco-auditing-tool

DESCRIPTION **Cisco Auditing Tool** - Perl script which scans cisco routers for common vulnerabilities. Checks for default passwords, easily guessable community names, and the IOS history bug. Includes support for plugins and scanning multiple hosts.

USAGE ./CAT [options]

OPTIONS

-h hostname (for scanning single hosts)
-f hostfile (for scanning multiple hosts)
-p port # (default port is 23)
-w wordlist (wordlist for community name guessing)
-a passlist (wordlist for password guessing)
-i [ioshist] (Check for IOS History bug)
-l logfile (file to log to, default screen)
-q quiet mode (no screen output)

EXAMPLE ./CAT -h 192.168.1.100 -w wordlist -a passwords -i

EXAMPLE ./CAT -h 192.168.1.22 -a lists/passwords -w lists/community (Audit Cisco Telnet Password & SNMP Community String)

cisco-global-exploiter

DESCRIPTION **Cisco Global Exploiter (CGE),** is an advanced, simple and fast security testing tool/ exploit engine, that is able to exploit 14 vulnerabilities in disparate Cisco switches and routers. CGE is command-line driven Perl script which has a simple and easy to use front-end.

USAGE cge.pl <target> <vulnerability number>

OPTIONS (14 vulnerabilities)

[1] - Cisco 677/678 Telnet Buffer Overflow Vulnerability
[2] - Cisco IOS Router Denial of Service Vulnerability
[3] - Cisco IOS HTTP Auth Vulnerability
[4] - Cisco IOS HTTP Configuration Arbitrary Administrative Access Vulnerability
[5] - Cisco Catalyst SSH Protocol Mismatch Denial of Service Vulnerability
[6] - Cisco 675 Web Administration Denial of Service Vulnerability
[7] - Cisco Catalyst 3500 XL Remote Arbitrary Command Vulnerability
[8] - Cisco IOS Software HTTP Request Denial of Service Vulnerability
[9] - Cisco 514 UDP Flood Denial of Service Vulnerability
[10] - CiscoSecure ACS for Windows NT Server Denial of Service Vulnerability
[11] - Cisco Catalyst Memory Leak Vulnerability
[12] - Cisco CatOS CiscoView HTTP Server Buffer Overflow Vulnerability
[13] - 0 Encoding IDS Bypass Vulnerability (UTF)
[14] - Cisco IOS HTTP Denial of Service Vulnerability

EXAMPLE cge.pl 192.168.1.254 3 (exploit the Cisco IOS HTTP Auth Vulnerability and hopefully using the nice link provided we should have basic access to the switch we are attacking, (not enable))

cisco-ocs

DESCRIPTION **cisco-ocs** also known as **cisco-ocs Mass Scanner**. This tool provides a single function which is to scan large ranges of IP's looking for Cisco devices or really any device listening on TCP port 23, attempts to login using telnet with a password of cisco, then passes the enable command to the Cisco router if its able to login via telnet, uses cisco again for the enable password, and finally reports a success if its able to get to the enable prompt using these exact steps. Unfortunately, this is the only function of the tool as you cannot specify a wordlist of passwords to attempt or for that matter you cannot set anything accept for the range of IP addresses to scan.

USAGE ./ocs <range start IP> <range end IP>

EXAMPLE ./ocs 192.168.1.21 192.168.1.23

cisco-torch

DESCRIPTION **Cisco Torch** was designed as a mass scanning, fingerprinting, and exploitation tool. **Cisco Torch** is unlike other tools in that it utilises multiple threads, (forking techniques), to launch scanning processes. It also uses several methods to simultaneously carry out application layer fingerprinting. **Cisco Torch** can be used for launching dictionary based password attacks against the services and discovering hosts running the following services: Telnet, SSH, Web, NTP, SNMP.

USAGE ./cisco-torch.pl <options> <IP,hostname,network>

USAGE ./cisco-torch.pl <options> -F <hostlist>

OPTIONS check http://www.vulnerabilityassessment.co.uk/torch.htm

EXAMPLE ./cisco-torch.pl -A 10.10.0.0/16

EXAMPLE ./cisco-torch.pl -s -b -F sshtocheck.txt

EXAMPLE ./cisco-torch.pl -w -z 10.10.0.0/16

EXAMPLE ./cisco-torch.pl -j -b -g -F tftptocheck.txt

yersinia

DESCRIPTION **Yersinia** is a network tool designed to take advantage of some weaknesses in different network protocols. It pretends to be a solid framework for analyzing and testing the deployed networks and systems. *Attacks for the following network protocols are implemented: Spanning Tree Protocol (STP), Cisco Discovery Protocol (CDP), Dynamic Trunking Protocol (DTP), Dynamic Host Configuration Protocol (DHCP), Hot Standby Router Protocol (HSRP), IEEE 802.1Q, IEEE 802.1X, Inter-Switch Link Protocol (ISL), VLAN Trunking Protocol (VTP)*

USAGE yersinia [-hVID] [-l logfile] protocol [protocol_options]

OPTIONS

-V Program version.

-h This help screen.

-I Interactive mode (ncurses).

-D Daemon mode.

-l logfile Select logfile.

-c conffile Select config file.

protocol Can be one of the following: cdp, dhcp, dot1q, dtp, hsrp, stp, vtp

EXAMPLE yersinia –D (run in Daemon mode)

[41] EXPLOIT DATABASE

- searchsploit

searchsploit

DESCRIPTION **searchsploit** - a shell script to search a local repository of exploitdb

USAGE searchsploit [term1] [term2] [term3]

EXAMPLE searchsploit oracle windows local (Use lower case in the search terms; second and third terms are optional. searchsploit will search each line of the csv file left to right so order your search terms accordingly. (i.e., 'oracle local' will yield better results than 'local oracle'))

[42] METASPLOIT

- metasploit community / pro
- metasploit diagnostic logs
- metasploit diagnostic shell
- metasploit framework
- update metasploit

metasploit community / pro

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

Metasploit Community Edition, a free, web-based user interface for **Metasploit. Metasploit Community** is based on the commercial functionality of the paid-for editions with a reduced set of features, including network discovery, module browsing, and manual exploitation. **Metasploit Community** is included in the main installer.

Metasploit Pro, an open-core commercial **Metasploit** edition for penetration testers. **Metasploit Pro** includes all features of Metasploit Express and adds web application scanning and exploitation, social engineering campaigns, and VPN pivoting.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

metasploit diagnostic logs

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

metasploit diagnostic shell

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

metasploit framework

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

Metasploit Framework, a tool for developing and executing exploit code against a remote target machine. The most basic and free version. It contains a command line interface, third-party import, manual exploitation and manual brute forcing

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

update metasploit

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

[43] NETWORK EXPLOITATION

- exploit6
- ikat
- jboss-autopwn-linux
- jboss-autopwn-win
- termineter

exploit6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

exploit6 - performs exploits of various CVE known IPv6 vulnerabilities on the destination. Note that for exploitable overflows only “AAA...” strings are used. If a system is vulnerable, it will crash, so be careful.

USAGE exploit6 interface destination [test-case-number]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbor solitications which are sent to a non-existing mac, and are therefore very easy to detect). If you dont want this, change the code.

ikat

DESCRIPTION **iKAT (The Interactive Kiosk Attack Tool)** has become the world's de-facto standard security tool when conducting penetration testing of controlled browser environments, such as : Citrix Sessions, WebTVs, In-Flight Entertainment Systems and Internet Kiosk platforms. **iKAT** is a 100% free SaaS website that you can visit from any browser environment. iKAT will attempt to exploit the browser and spawn a local shell for you.

USAGE n/a

EXAMPLE n/a

jboss-autopwn-linux

DESCRIPTION **JBoss Autopwn** - A **JBoss** script for obtaining remote shell access. This **JBoss** script deploys a JSP shell on the target **JBoss** AS server. Once deployed, the script uses its upload and command execution capability to provide an interactive session.

Features

- *Multiplatform support - tested on Windows, Linux and Mac targets*
- *Support for bind and reverse bind shells*
- *Meterpreter shells and VNC support for Windows targets*

USAGE n/a

EXAMPLE n/a

jboss-autopwn-win

DESCRIPTION **JBoss Autopwn** - A **JBoss** script for obtaining remote shell access. This **JBoss** script deploys a JSP shell on the target **JBoss** AS server. Once deployed, the script uses its upload and command execution capability to provide an interactive session.

Features

- *Multiplatform support - tested on Windows, Linux and Mac targets*
- *Support for bind and reverse bind shells*
- *Meterpreter shells and VNC support for Windows targets*

USAGE n/a

EXAMPLE n/a

termineter

DESCRIPTION **Termineter** is a framework written in python to provide a platform for the security testing of smart meters. It implements the C12.18 and C12.19 protocols for communication. Currently supported are Meters using C12.19 with 7-bit character sets. **Termineter** communicates with Smart Meters via a connection using an ANSI type-2 optical probe with a serial interface.

USAGE n/a

EXAMPLE

Below is a summary of the basic steps to get started with Termineter after the environment has been configured.

- Connect the optical probe to the smart meter and start termineter
- Configure the connection options. On Windows, this would be something like COM1 and on Linux something like /dev/ttyS0. Check Configuring the Connection for more details.
- Use the connect command, this will also check that the meter is responding.

```
[someone@localhost ~]$ lsusb
 Bus 006 Device 002: ID 0403:f458 Future Technology Devices International, Ltd
 Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
 Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
 Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
 Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
[someone@localhost ~]$ sudo modprobe ftdi-sio vendor=0x0403 product=0xf458
[someone@localhost ~]$ ls /dev/tty* | grep USB
/dev/ttyUSB0
[someone@localhost ~]$
```

[44] SOCIAL ENGINEERING TOOLKIT

- se-toolkit

se-toolkit

DESCRIPTION **The Social-Engineer Toolkit (SET)** 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 in a fraction of the time.
More info: http://www.binarytides.com/hack-windows-social-engineering-toolkit-java-applet/

USAGE n/a

EXAMPLE Select from the menu:

1) Social-Engineering Attacks
2) Fast-Track Penetration Testing
3) Third Party Modules
4) Update the Metasploit Framework
5) Update the Social-Engineer Toolkit
6) Update SET configuration
7) Help, Credits, and About
99) Exit the Social-Engineer Toolkit

[45] SNIFFING/SPOOFING: NETWORK SNIFFERS

- darkstat
- dnschef
- dnsspoof
- dsniff
- ettercap-graphical
- hexinject
- mailsnarf
- netsniff-ng
- passive_discovery6
- sslsniff
- tcpflow
- urlsnarf
- webmitm
- webspy
- wireshark

darkstat

DESCRIPTION **darkstat** is a packet sniffer that runs as a background process, gathers all sorts of statistics about network usage, and serves them over HTTP.

USAGE **darkstat** [**-i***interface*] [**-r** *file*] [**--snaplen** *bytes*] [**--pppoe**] [**--syslog**] [**--verbose**] [**--no-daemon**] [**--no-promisc**] [**--no-dns**] [**--no-macs**] [**--no-lastseen**] [**-p** *port*] [**-b** *bindaddr*] [**-f** *filter*] [**-l** *network/netmask*] [**--local-only**] [**--chroot***dir*] [**--user** *username*] [**--daylog** *filename*] [**--import** *filename*] [**--export***filename*] [**--pidfile***filename*] [**--hosts-max** *count*] [**--hosts-keep** *count*] [**--ports-max** *count*] [**--ports-keep** *count*] [**--highest-port** *port*] [**--wait** *secs*] [**--hexdump**]

OPTIONS http://linux.die.net/man/8/darkstat

EXAMPLE darkstat -i fxp0 (gather statistics on the fxp0 interface)

EXAMPLE darkstat -i fxp0 -b 192.168.0.1 (account for traffic on the Internet-facing interface, but only serve web pages to our private local network where we have the IP address 192.168.0.1)

EXAMPLE darkstat -i fxp0 -p 80 (serve web pages on the standard HTTP port)

EXAMPLE darkstat -i fxp0 -f "port 22“ (account for SSH traffic)

EXAMPLE darkstat -i fxp0 -f "not (src net 192.168.0 and dst net 192.168.0)" (don't account for traffic between internal IPs)

dnschef

DESCRIPTION **DNSChef** is a highly configurable DNS proxy for Penetration Testers and Malware Analysts. A DNS proxy (aka "Fake DNS") is a tool used for application network traffic analysis among other uses. For example, a DNS proxy can be used to fake requests for "badguy.com" to point to a local machine for termination or interception instead of a real host somewhere on the Internet. More info: http://thesprawl.org/projects/dnschef/

Without any parameters, DNSChef will run in full proxy mode. This means that all requests will simply be forwarded to an upstream DNS server (8.8.8.8 by default) and returned back to the quering host.

USAGE dnschef.py [options]

OPTIONS https://github.com/bigsnarfdude/pythonNetworkProgrammingN00B/blob/master/dnschef.py

EXAMPLE ./dnschef.py -6
EXAMPLE ./dnschef.py --fakeip 127.0.0.1 –q
EXAMPLE ./dnschef.py --fakeip 127.0.0.1 --fakedomains thesprawl.org -q
EXAMPLE ./dnschef.py --fakeip 127.0.0.1 --truedomains thesprawl.org,*.webfaction.com -q

dnsspoof

DESCRIPTION **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.

USAGE **dnsspoof** [**-i** *interface*] [**-f** *hostsfile*] [*expression*]

OPTIONS

-i interface Specify the interface to use.

-f hostsfile Specify the pathname of a file in hosts(5) format. Only one hostname allowed per line (no aliases), although hostnames may contain wildcards (such as *.doubleclick.net).

expression Specify a tcpdump(8) filter expression to select traffic to sniff.

If no hostsfile is specified, replies will be forged for all address queries on the LAN with an answer of the local machine's IP address.

EXAMPLE

```
# echo 1 > /proc/sys/net/ipv4/ip_forward (enable port forwarding)
# arpspoof -t 192.168.1.245 192.168.1.5 &;
# arpspoof -t 192.168.1.5 192.168.1.245 &;
# dnsspoof -f  spoofhosts.txt host  192.168.1.245  and  udp  port 53
```

dsniff

DESCRIPTION **dSniff** - is a set of password sniffing and network traffic analysis tools to parse different application protocols and extract relevant information. **dsniff**, **filesnarf**, **mailsnarf**, **msgsnarf**, **urlsnarf**, and **webspy** passively monitor a network for interesting data (passwords, e-mail, files, etc.). **arpspoof**, **dnsspoof**, and **macof** facilitate the interception of network traffic normally unavailable to an attacker (e.g., due to layer-2 switching). **sshmitm** and **webmitm** implement active man-in-the-middle attacks against redirected SSH and HTTPS sessions by exploiting weak bindings in ad-hoc PKI.

dsniff is a password sniffer which handles FTP, Telnet, SMTP, HTTP, POP, poppass, NNTP, IMAP, SNMP, LDAP, Rlogin, RIP, OSPF, PPTP MS-CHAP, NFS, VRRP, YP/NIS, SOCKS, X11, CVS, IRC, AIM, ICQ, Napster, PostgreSQL, Meeting Maker, Citrix ICA, Symantec pcAnywhere, NAI Sniffer, Microsoft SMB, Oracle SQL*Net, Sybase and Microsoft SQL protocols.

dsniff automatically detects and minimally parses each application protocol, only saving the interesting bits, and uses Berkeley DB as its output file format, only logging unique authentication attempts. Full TCP/IP reassembly is provided by **libnids**.

USAGE **dsniff** [**-c**] [**-d**] [**-m**] [**-n**] [**-i** *interface* | **-p** *pcapfile*] [**-s** *snaplen*] [**-f** *services*] [**-t** *trigger[,...]*] [**-r**|**-w** *savefile*] [*expression*]

OPTIONS http://linux.die.net/man/8/dsniff

EXAMPLE dsniff -ni eth0 (The following example demonstrates how to use dsniff to an ftp sessions)

ettercap-graphical

DESCRIPTION **Ettercap** is a comprehensive suite for man in the middle attacks. It features sniffing of live connections, content filtering on the fly and many other interesting tricks. It supports active and passive dissection of many protocols and includes many features for network and host analysis.
More info: http://ettercap.github.io/ettercap/

USAGE **ettercap** [*OPTIONS*] [*TARGET1*] [*TARGET2*] *TARGET* is in the form *MAC/IPs/IPv6/PORTs*
where IPs and PORTs can be ranges (e.g. /192.168.0.1-30,40,50/20,22,25)
OPTIONS http://linux.die.net/man/8/ettercap

EXAMPLE **ettercap –Tp** (Use the console interface and do not put the interface in promisc mode. You will see only your traffic.)
EXAMPLE **ettercap –Tzq** (Use the console interface, do not ARP scan the net and be quiet. The packet content will not be displayed, but user and passwords, as well as other messages, will be displayed.)
EXAMPLE **ettercap -T -j /tmp/victims -M arp /10.0.0.1-7/ /10.0.0.10-20/** (Will load the hosts list from /tmp/victims and perform an ARP poisoning attack against the two target. The list will be joined with the target and the resulting list is used for ARP poisoning.)
EXAMPLE **ettercap -Tzq /10.0.0.1/21,22,23** (Sniff telnet, ftp and ssh connections to 10.0.0.1.)
EXAMPLE **ettercap -T -M arp:remote /192.168.1.1/ /192.168.1.2-10** (Perform the ARP poisoning against the gateway and the host in the lan between 2 and 10. The 'remote' option is needed to be able to sniff the remote traffic the hosts make through the gateway.)

hexinject

DESCRIPTION **HexInject** is a very versatile **packet injector and sniffer**, that provide a command-line framework for raw network access.
It's designed to work together with others command-line utilities, and for this reason it facilitates the creation of powerful shell scripts capable of reading, intercepting and modifying network traffic in a transparent manner. In a single line, why should you consider hexinject? Because it's able to inject anything into the network, and, for the TCP/IP protocols, it automatically calculates the checksum and the packet size fields.

USAGE hexinject <mode> <options>

OPTIONS http://www.securitytube-tools.net/index.php@title=Hexinject.html

EXAMPLE hexinject -s -i eth0 -c 1 -f 'arp' | replace '06 04 00 01' '06 04 00 02' | hexinject -p -i eth0
EXAMPLE hexinject -s -i eth0 -c 1 -f 'src host 192.168.1.9' | hexinject -p -i eth1
EXAMPLE hexinject -s -i mon0
EXAMPLE hexinject -s -r -i mon1 | strings

mailsnarf

DESCRIPTION **mailsnarf** outputs e-mail messages sniffed from SMTP and POP traffic in Berkeley mbox format, suitable for offline browsing with your favourite mail reader (**mail**, **pine**, etc.).

USAGE **mailsnarf** [**-i** *interface* | **-p** *pcapfile*] [[**-v**] *pattern [expression*]]

OPTIONS

-i interface Specify the interface to listen on.

-p pcapfile Process packets from the specified PCAP capture file instead of the network.

-v "Versus" mode. Invert the sense of matching, to select non-matching messages.

pattern Specify regular expression for message header/body matching.

expression Specify a tcpdump(8) filter expression to select traffic to sniff.

EXAMPLE
```
mailsnarf –v “-----BEGIN PGP MESSAGE-----” | \
perl –ne ‘print if /^From / .. /^$/;’ | \
tee insecure-mail-headers
```

netsniff-ng

DESCRIPTION **netsniff-ng** is a free, performant Linux networking toolkit. **netsniff-ng** is a high-performance network analyzer based on packet **mmap** mechanisms. It can record **pcap** files to disc, replay them and also do an offline and online analysis. Capturing, analysis or replay of raw 802.11 frames are supported as well. **pcap** files are also compatible with **tcpdump** or **Wireshark** traces. **netsniff-ng** processes those **pcap** traces either in scatter-gather I/O or by **mmap** I/O.

USAGE The newly introduced command line option of --in and --out allows a flexible combination for different purposes, i.e.

1) --in <netdev> --out <pcap> writes a network trace to disc
2) --in <pcap> --out <netdev> replays a network trace from disc
3) --in <pcap> performs an offline analysis of a trace file
4) --in <netdev> performs an online analysis
5) --in <netdev> --out <folder> periodically writes network trace files
6) --in <netdev1> --out <netdev2> redirects network traffic
7) --in <pcap> --out <txf> rewrites a pcap file into a txf file for trafgen

OPTIONS http://pub.netsniff-ng.org/docs/Netsniff-ng

EXAMPLE n/a

passive_discovery6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

exploit6 - Performs exploits of various CVE known IPv6 vulnerabilities on the destination. Note that for exploitable overflows only 'AAA...' strings are used. If a system is vulnerable, it will crash, so be careful!

USAGE exploit6 interface destination [test-case-number]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbor solitications which are sent to a non-existing mac, and are therefore very easy to detect). If you dont want this, change the code.

sslsniff

DESCRIPTION It is designed to MITM all SSL connections on a LAN, and dynamically generates certificates for the domains that are being accessed on the fly. The new certificates are constructed in a certificate chain that is signed by any certificate that you provide.
More info: https://github.com/moxie0/sslsniff

USAGE sslsniff -a -c <path/to/your/certificate> -f ios -h <httpPort> -s <sslPort> -w iphone.log
USAGE ./sslsniff -t -s <$listenPort> -w <$logFile> -m IPSCACLASEA1.crt \ -c <$certDir>

EXAMPLE

Assuming we want to intercept SSL traffic from 172.17.10.36, we need to trick that host into thinking that we're the router. Using arpspoof, we can convince the target that the router's MAC address is our MAC address.

arpspoof -i eth0 -t 172.17.10.36 172.17.8.1

or

arp-sk -r -S 172.17.8.1 -D 172.17.10.36

At this point, any SSL traffic should get proxied by sslsniff and logged to a file.

First, arpspoof convinces a host that our MAC address is the router's MAC address, and the target begins to send us all its network traffic. The kernel forwards everything along except for traffic destined to port 443, which it redirects to $listenPort (10000, for example).
At this point, sslsniff receives the client connection, makes a connection to the real SSL site, and looks at the information in the server's certificate. sslsniff then generates a new certificate with an identical Distinguished Name and signs it with the end-entity certificate in $certificateFile. sslsniff uses the generated certificate chain to do a SSL handshake with the client and proxy data between both hosts (while logging it, of course).

tcpflow

DESCRIPTION **tcpflow** is a program that captures data transmitted as part of TCP connections (flows), and stores the data in a way that is convenient for protocol analysis and debugging. Each TCP flow is stored in its own file. Thus, the typical TCP flow will be stored in two files, one for each direction. **tcpflow** can also process stored 'tcpdump' packet flows.

tcpflow stores all captured data in files that have names of the form:
[timestampT]sourceip.sourceport-destip.destport[--VLAN][cNNNN]

USAGE tcpflow [-chpsv] [-b max_bytes] [-d debug_level] [-f max_fds] [-i iface] [-r file] [expression]

OPTIONS http://linux.die.net/man/1/tcpflow

EXAMPLE tcpflow -c -n en1 src or dst host api.example.com
EXAMPLE tcpflow host sundown (To record all packets arriving at or departing from *sundown*)
EXAMPLE tcpflow host helios and \(hot or ace \) (To record traffic between helios and either hot or ace)
EXAMPLE tcpflow host ace and not helios (To record traffic between ace and any host except helios)
EXAMPLE tcpflow net ucb-ether (To record all traffic between local hosts and hosts at Berkeley)
EXAMPLE tcpflow 'gateway snup and (port ftp or ftp-data)' (To record all ftp traffic through internet gateway snup: (note that the expression is quoted to prevent the shell from (mis-)interpreting the parentheses))

urlsnarf

DESCRIPTION **urlsnarf** outputs all requested URLs sniffed from HTTP traffic in CLF (Common Log Format, used by almost all web servers), suitable for offline post-processing with your favorite web log analysis tool (analog, wwwstat, etc.).

USAGE **urlsnarf** [**-n**] [**-i** *interface*] [[**-v**] *pattern [expression*]]

OPTIONS

-n	Do not resolve IP addresses to hostnames.
-i interface	
-v	"Versus" mode. Invert the sense of matching, to select non-matching URLs. Specify the interface to listen on.
pattern	Specify regular expression for URL matching.
expression	Specify a tcpdump(8) filter expression to select traffic to sniff.

EXAMPLE urlsnarf -i lo

webmitm

DESCRIPTION **webmitm** transparently proxies and sniffs HTTP / HTTPS traffic redirected by **dnsspoof**, capturing most "secure" SSL-encrypted webmail logins and form submissions.

USAGE webmitm [-d] [host]

OPTIONS

-d	Enable debugging mode. May be specified multiple times to greater effect
host	Specify a host to proxy to. If none given, only requests containing an HTTP/1.1 Host: header or absolute URI will be relayed transparently

EXAMPLE webmitm -d

webspy

DESCRIPTION **webspy** sends URLs sniffed from a client to your local Netscape browser for display, updated in real-time (as the target surfs, your browser surfs along with them, automagically). Netscape must be running on your local X display ahead of time.

USAGE webspy [-i interface | -p pcapfile] host

OPTIONS

-i interface (Specify the interface to listen on)

-p pcapfile (Process packets from the specified PCAP capture file instead of the network)

Host (Specify the web client to spy on)

EXAMPLE webspy -i eth0 192.168.1.66

wireshark

DESCRIPTION **wireshark** - Interactively dump and analyze network traffic. **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. **Wireshark**'s native capture file format is **libpcap** format, which is also the format used by **tcpdump** and various other tools.

USAGE **wireshark** [**-a** <capture autostop condition>] ... [**-b** <capture ring buffer option>] ... [**-B** <capture buffer size (Win32 only)>] [**-c** <capture packet count>] [**-C** <configuration profile>] [**-D**] [**--display=**<X display to use>] [**-f** <capture filter>] [**-g** <packet number>] [**-h**] [**-H**] [**-i** <capture interface>|-] [**-k**] [**-K** <keytab>] [**-l**] [**-L**] [**-m** <font>] [**-n**] [**-N** <name resolving flags>] [**-o** <preference/recent setting>] ... [**-p**] [**-P** <path setting>] [**-Q**] [**-r** <infile>] [**-R** <read (display) filter>] [**-S**] [**-s** <capture snaplen>] [**-t**ad|a|r|d|dd|e] [**-v**] [**-w** <outfile>] [**-y** <capture link type>] [**-X** <eXtension option>] [**-z** <statistics>] [<infile>]

OPTIONS http://linux.die.net/man/1/wireshark

EXAMPLE n/a; GUI tool

[46] NETWORK SPOOFING

- dnschef
- ettercap-graphical
- evilgrade
- fake_advertise6
- fake_dns6d
- fake_dnsupdate6
- fake_mipv6
- fake_mld26
- fake_mld6
- fake_mldrouter6
- fake_router6
- fake_solicitate6
- fiked
- macchanger
- parasite6
- randicmp6
- rebind
- redir6
- sniffjoke
- sslstrip
- tcpreplay
- wifi-honey
- yersinia

dnschef

DESCRIPTION **DNSChef** is a highly configurable DNS proxy for Penetration Testers and Malware Analysts. A DNS proxy (aka "Fake DNS") is a tool used for application network traffic analysis among other uses. For example, a DNS proxy can be used to fake requests for "badguy.com" to point to a local machine for termination or interception instead of a real host somewhere on the Internet. More info: http://thesprawl.org/projects/dnschef/

Without any parameters, DNSChef will run in full proxy mode. This means that all requests will simply be forwarded to an upstream DNS server (8.8.8.8 by default) and returned back to the quering host.

USAGE dnschef.py [options]

OPTIONS https://github.com/bigsnarfdude/pythonNetworkProgrammingN00B/blob/master/dnschef.py

EXAMPLE ./dnschef.py -6
EXAMPLE ./dnschef.py --fakeip 127.0.0.1 –q
EXAMPLE ./dnschef.py --fakeip 127.0.0.1 --fakedomains thesprawl.org -q
EXAMPLE ./dnschef.py --fakeip 127.0.0.1 --truedomains thesprawl.org,*.webfaction.com -q

ettercap-graphical

DESCRIPTION **Ettercap** is a comprehensive suite for man in the middle attacks. It features sniffing of live connections, content filtering on the fly and many other interesting tricks. It supports active and passive dissection of many protocols and includes many features for network and host analysis.
More info: http://ettercap.github.io/ettercap/

USAGE **ettercap** [*OPTIONS*] [*TARGET1*] [*TARGET2*] *TARGET* is in the form *MAC/IPs/IPv6/PORTs*
where IPs and PORTs can be ranges (e.g. /192.168.0.1-30,40,50/20,22,25)
OPTIONS http://linux.die.net/man/8/ettercap

EXAMPLE **ettercap –Tp** (Use the console interface and do not put the interface in promisc mode. You will see only your traffic.)
EXAMPLE **ettercap –Tzq** (Use the console interface, do not ARP scan the net and be quiet. The packet content will not be displayed, but user and passwords, as well as other messages, will be displayed.)
EXAMPLE **ettercap -T -j /tmp/victims -M arp /10.0.0.1-7/ /10.0.0.10-20/** (Will load the hosts list from /tmp/victims and perform an ARP poisoning attack against the two target. The list will be joined with the target and the resulting list is used for ARP poisoning.)
EXAMPLE **ettercap -Tzq /10.0.0.1/21,22,23** (Sniff telnet, ftp and ssh connections to 10.0.0.1.)
EXAMPLE **ettercap -T -M arp:remote /192.168.1.1/ /192.168.1.2-10** (Perform the ARP poisoning against the gateway and the host in the lan between 2 and 10. The 'remote' option is needed to be able to sniff the remote traffic the hosts make through the gateway.)

evilgrade

DESCRIPTION **Evilgrade** is a modular framework that allows the user to take advantage of poor upgrade implementations by injecting fake updates. This framework comes into play when the attacker is able to make traffic redirection, and such thing can be done in several ways such as: DNS tampering, DNS Cache Poisoning, ARP spoofing Wi-Fi Access Point impersonation, DHCP hijacking with your favourite tools. This way you can easy take control of a fully patched machine during a penetration test in a clean and easy way. The main idea behind the is to show the amount of trivial errors in the update process of mainstream applications.

USAGE ./evilgrade show modules

OPTIONS

-show <object> Used to show different information.
-conf <object> Enter to the configure mode.
-set <option> “value” Configures different options.
-start Services starts.
-stop Services stops.
-status Services status.

EXAMPLE http://r00tsec.blogspot.co.uk/2011/07/hacking-with-evilgrade-on-backtrack5.html
EXAMPLE https://forum.intern0t.org/offensive-guides-information/761-how-use-evilgrade.html

fake_advertise6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_advertise6 - advertise ipv6 address on the network (with own mac if not defined) sending it to the all-nodes multicast address if no target specified.

USAGE fake_advertise6 <interface> <ip-address> [target-address [own-mac-address]]

EXAMPLE fake_advertise6 eth1 fe80::fd:ff:fe00:401 ff02::1 02:fd:00:00:04:01 (With the thc-ipv6 tool fake_advertise6 we will advertise Ethernet addresses for that host. To send a Neighbor Advertisement (NA) with valid parameters, we use the following command)

EXAMPLE fake_advertise6 eth1 fe80::fd:ff:fe00:401 ff02::1 02:fd:00:00:04:10 (Such an alert is raised when a node's Ethernet address changes. In our example, we advertise a new Etherner address 02:fd:00:00:04:10)

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_dns6d

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_dns6d - fake DNS server that serves the same ipv6 address to any lookup request. You can use this together with **parasite6** if clients have a fixed DNS server. *Note: very simple server.* Does not honour multiple queries in a packet, nor NS, MX, etc. lookups.

USAGE fake_dns6d <interface> <ip-address> [fake-ipv6-address [fake-mac]]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_dnsupdate6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_dnsupdate6 – fake DNS update.

USAGE fake_dnsupdate6 dns-server full-qualified-host-dns-name ipv6address

EXAMPLE fake_dnsupdate6 dns.test.com myhost.sub.test.com ::1

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_mipv6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_mipv6 - if the mobile IPv6 home-agent is mis-configured to accept MIPV6 updates without IPSEC, this will redirect all packets for home-address to care-of-address. **fake_mipv6** - steal a mobile IP to yours if IPSEC is not needed for authentication.

USAGE fake_mipv6 <interface> <home-address> <home-agent-address> <care-of-address>

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_mld26

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_mld26 – announce yourself in a multicast group of your choice on the net (for MLDv2).

USAGE fake_mld26 [-l] <interface> <add|delete|query> [multicast-address [target-address [ttl [own-ip [own-mac-address [destination-mac-address]]]]]]

TIP Use -l to loop and send (in 5s intervals) until Control-C is pressed.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_mld6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_mld6 - advertise yourself in a multicast group of your choice on the net. Query ask on the network who is listening to multicast address. Ad(d)vertise or delete yourself - or anyone you want - in a multicast group of your choice

USAGE fake_mld6 <interface> <multicast-address> [[target-address] [[ttl] [[own-ip] [own-mac-address]]]]

TIP Use -l to loop and send (in 5s intervals) until Control-C is pressed.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_mldrouter6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_mldrouter6 – fake MLD router messages. Announce, delete or solicitate MLD router - yourself or others.

USAGE fake_mldrouter6 [-l] <interface> <advertise|solicitate|terminate> [own-ip [own-mac-address]]

TIP Use -l to loop and send (in 5s intervals) until Control-C is pressed.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_router6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_router6 - announce yourself as a router and try to become the default router. If a non-existing mac-address is supplied, this results in a DOS.

USAGE fake_router6 **<interface> <router-ip-link-local** network-address/prefix-length> <mtu> [mac-address]

OPTIONS option -H adds hop-by-hop, -F fragmentation header and -D dst header.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fake_solicitate6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fake_solicitate6 - solicate ipv6 address on the network, sending it to the all-nodes multicast address.

USAGE fake_solicitate6 [-DHF] <interface> <ip-address-solicitated> [target-address [mac-address-solicitated [source-ip-address]]]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fiked

DESCRIPTION **fiked** - a fake IKE PSK+XAUTH daemon based on vpnc. **Fiked** impersonates a VPN gateway's IKE responder in order to capture XAUTH login credentials in insecure PSK+XAUTH setups, such as commonly found in Cisco based VPN sites.

Fiked supports IKEv1 in aggressive mode, using pre-shared keys and XAUTH. Supported algorithms are DES, 3DES, AES-128, AES-192, AES-256; MD5, SHA1; and DH groups 1, 2 and 5. IKE main mode is not supported.

To actually implement an attack against a VPN site, you have to intercept the IKE traffic to the VPN gateway and redirect it to **fiked**. Your options include **hostap** or ARP poisoning, and usually will require some (e.g., **pf**) redirection magic, depending on the situation.

USAGE **fiked** [**-rdqhV**] **-g** *gateway* **-k** *id*:*secret* [**-u** *user*] [**-l** *file*] [**-L** *file*]

OPTIONS http://www.irongeek.com/i.php?page=backtrack-r1-man-pages/fiked

EXAMPLE **fiked -g 10.0.0.1 -k group1:g3h31m -k group2:s3kr3t -l account.log** (To impersonate *gateway* 10.0.0.1 using *secret*s for group *id*s group1 and group2, writing results to *file* account.log)

EXAMPLE **fiked -g 10.0.0.1 -k group1:g3h31m -l account.log -d -L fiked.log** (The same with only one key, and running as a daemon logging to file fiked.log)

macchanger

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

USAGE **macchanger** [*options*] device

OPTIONS

-h, --help Show summary of options
-V, --version Show version of program
-e, --endding Don't change the vendor bytes
-a, --another Set random vendor MAC of the same kind
-A Set random vendor MAC of any kind
-r, --random Set fully random MAC
-l, --list[=keyword] Print known vendors (with keyword in the vendor's description string)
-m, --mac XX:XX:XX:XX:XX:XX Set the MAC XX:XX:XX:XX:XX:XX

EXAMPLE macchanger eth1
EXAMPLE macchanger -A eth1
EXAMPLE macchanger --endding eth1
EXAMPLE macchanger --mac=01:23:45:67:89:AB eth1

parasite6

DESCRIPTION **thc-ipv6** - Tools to play with IPv6 .

parasite6 – This is an "ARP spoofer" for IPv6, redirecting all local traffic to your own system (or nirvana if fake-mac does not exist) by answering falsely to Neighbor Solitication requests, specifying FAKE-MAC results in a local DOS

USAGE **parasite6** [-lRFHD] **<interface> [fake-mac]**

OPTIONS Option -l loops and resends the packets per target every 5 seconds

OPTIONS Option -R will also try to inject the destination of the solicitation

OPTIONS NS security bypass: -F fragment, -H hop-by-hop and -D large destination header

EXAMPLE n/a

randicmp6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

randicmp6 - sends all ICMPv6 type and code combinations to destination.

USAGE randicmp6 [-s sourceip] interface destination [type [code]]

OPTIONS Option -s sets the source ipv6 address.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

rebind

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

redir6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

redir6 - redirect traffic to you intelligently (man-in-the-middle) with a clever icmp6 redirect spoofer. Implant a route into victim-ip, which redirects all traffic to target-ip to new-ip. You must know the router which would handle the route. If the new-router-mac does not exist, this results in a DOS. If the TTL of the target is not 64, then specify this is the last option.

USAGE redir6 <interface> <victim-ip> <target-ip> <original-router> <new-router> [new-router-mac] [hop-limit]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

sniffjoke

DESCRIPTION **SniffJoke** - transparent TCP connection scrambler**. SniffJoke** is a software able to confuse the Internet traffic analysis, developed with the aim to improve digital privacy in communications and to show and test some security weakness in traffic analysis software. **SniffJoke** - an internet client running **SniffJoke** injects in the transmission flow some packets able to seriously disturb passive analysis like sniffing, interception and low level information theft. No server supports needed!

More info: https://github.com/vecna/sniffjoke

USAGE sniffjoke --location name_of_your_location

USAGE sniffjokectl –stat

USAGE sniffjokectl –start

USAGE sniffjokectl --help

EXAMPLE n/a

sslstrip

DESCRIPTION **sslstrip** provides a demonstration of the HTTPS stripping attacks. It will transparently hijack HTTP traffic on a network, watch for HTTPS links and redirects, then map those links into either look-alike HTTP links or homograph-similar HTTPS links. It also supports modes for supplying a favicon which looks like a lock icon, selective logging, and session denial. First, **arpspoof** convinces a host that our MAC address is the router's MAC address, and the target begins to send us all its network traffic. The kernel forwards everything along except for traffic destined to port 80, which it redirects to $listenPort (10000, for example). At this point, **sslstrip** receives the traffic and does its magic.

USAGE sslstrip.py -l <listenPort>

EXAMPLE

Flip your machine into forwarding mode.

```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Setup iptables to redirect HTTP traffic to sslstrip.

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port <listenPort>
```

Run sslstrip.

```
sslstrip.py -l <listenPort>
```

Run arpspoof to convince a network they should send their traffic to you.

```
arpspoof -i <interface> -t <targetIP> <gatewayIP>
```

tcpreplay

DESCRIPTION **tcpreplay** - replay network traffic stored in pcap files. The basic operation of **tcpreplay** is to resend all packets from the input file(s) at the speed at which they were recorded, or a specified data rate, up to as fast as the hardware is capable. Optionally, the traffic can be split between two interfaces, written to files, filtered and edited in various ways, providing the means to test firewalls, NIDS and other network devices. For more details, please see the **tcpreplay** Manual at: http://tcpreplay.synfin.net/trac/wiki/manual

USAGE tcpreplay [-flag [value]]... [--opt-name [[=|]value]]... <pcap_file(s)>

OPTIONS http://tcpreplay.synfin.net/tcpreplay.html

EXAMPLE tcpreplay --intf1=eth0 sample.pcap (replay a given pcap as it was captured all you need to do is specify the pcap file and the interface to send the traffic out interface 'eth0')

EXAMPLE tcpreplay --topspeed --intf1=eth0 sample.pcap (replay traffic as quickly as possible)

EXAMPLE tcpreplay --loop=10 --intf1=eth0 sample.pcap (replay the sample.pcap file 10 times)

EXAMPLE tcpreplay --oneatatime --verbose --intf1=eth0 sample.pcap (replay packets, one at a time while decoding it (useful for debugging purposes))

EXAMPLE tcpreplay --cachefile=sample.prep --intf1=eth0 --intf2=eth1 sample.pcap (By utilizing tcpprep cache files, tcpreplay can split traffic between two interfaces. This allows tcpreplay to send traffic through a device and emulate both client and server sides of the connection, thereby maintaining state. Using a tcpprep cache file to split traffic between two interfaces (eth0 & eth1) with tcpreplay is simple)

wifi-honey

DESCRIPTION **wifi-honey** works out what encryption a client is looking for in a given network by setting up four fake access points, each with a different type of encryption - None, WEP, WPA and WPA2 - and then observing which of the four the client connects to.

In the case of WPA/WPA2, by running **airodump-ng** along side this you also end up capturing the first two packets of the four way handshake and so can attempt to crack the key with either **aircrack-ng** or **coWPAtty**.

What this script does - is to automate the setup process, it creates five monitor mode interfaces, four are used as APs and the fifth is used for **airodump-ng**. To make things easier, rather than having five windows all this is done in a screen session which allows you to switch between screens to see what is going on. All sessions are labelled so you know which is which.

USAGE ./wifi_honey.sh <essid> <channel> <interface>

USAGE ./wifi_honey.sh fake_wpa_net (start the script with the ESSID of the network you want to impersonate)

USAGE ./wifi_honey.sh fake_wpa_net 1 wlan1 (You can also specify the channel to use and the interface you want to base the whole lot on)

EXAMPLE ./wifi_honey.sh THECRIB 11 wlan2

yersinia

DESCRIPTION **Yersinia** is a network tool designed to take advantage of some weaknesses in different network protocols. It pretends to be a solid framework for analyzing and testing the deployed networks and systems. *Attacks for the following network protocols are implemented: Spanning Tree Protocol (STP), Cisco Discovery Protocol (CDP), Dynamic Trunking Protocol (DTP), Dynamic Host Configuration Protocol (DHCP), Hot Standby Router Protocol (HSRP), IEEE 802.1Q, IEEE 802.1X, Inter-Switch Link Protocol (ISL), VLAN Trunking Protocol (VTP)*

USAGE yersinia [-hVID] [-l logfile] protocol [protocol_options]

OPTIONS

-V Program version.

-h This help screen.

-I Interactive mode (ncurses).

-D Daemon mode.

-l logfile Select logfile.

-c conffile Select config file.

protocol Can be one of the following: cdp, dhcp, dot1q, dtp, hsrp, stp, vtp

EXAMPLE yersinia –D (run in Daemon mode)

[47] VOICE AND SURVEILLANCE

- msgsnarf

msgsnarf

DESCRIPTION **msgsnarf** records selected messages from AOL Instant Messenger, ICQ 2000, IRC, MSN Messenger, or Yahoo Messenger chat sessions.

USAGE **msgsnarf** [**-i** *interface* | **-p** *pcapfile*] [[**-v**] *pattern [expression*]]

OPTIONS

-i interface	Specify the interface to listen on
-p pcapfile	Process packets from the specified PCAP capture file instead of the network.
-v	"Versus" mode. Invert the sense of matching, to select non-matching messages
pattern	Specify regular expression for message matching
expression	Specify a tcpdump(8) filter expression to select traffic to sniff

EXAMPLE msgsnarf -i lo

[48] VOIP TOOLS

- iaxflood
- inviteflood
- ohrwurm
- protos-sip
- rtpbreak
- rtpflood
- rtpinsertsound
- rtpmixsound
- sctpscan
- siparmyknife
- sipp
- sipsak
- svcrack
- svcrash
- svmap
- svreport
- svwar
- voiphopper

iaxflood

DESCRIPTION IAXFlood is a tool for flooding the IAX2 protocol which is used by the Asterisk PBX.

USAGE ./iaxflood sourcename destinationname numpackets

EXAMPLE n/a

inviteflood

DESCRIPTION This tool can be used to flood a target with INVITE requests it can be used to target sip gateways / proxies and sip phones. As long the tool keeps flooding the sip gateway it will prevent users from making phone calls. You can flood the sip proxy with an inexistent extension thus making it generating a 404 not found just to keep it busy.

USAGE ./inviteflood <interface> <target user> <target domain> <ipv4 addr of flood target> <flood stage>

OPTIONS

-a flood tool "From:" alias (e.g. jane.doe)
-i IPv4 source IP address [default is IP address of interface]
-S srcPort (0 - 65535) [default is well-known discard port 9]
-D destPort (0 - 65535) [default is well-known SIP port 5060]
-l lineString line used by SNOM [default is blank]
-s sleep time btwn INVITE msgs (usec)
-h help - print this usage
-v verbose output mode

interface (e.g. eth0)
target user (e.g. "" or john.doe or 5000 or "1+210-555-1212")
target domain (e.g. enterprise.com or an IPv4 address)
IPv4 addr of flood target (ddd.ddd.ddd.ddd)
flood stage (i.e. number of packets)

EXAMPLE ./inviteflood eth0 201 192.168.1.104 192.168.1.104 10000000

ohrwurm

DESCRIPTION **ohrwurm** is a small and simple RTP fuzzer, it has been tested it on a small number of SIP phones, none of them withstood the fuzzing.

Features:

- reads SIP messages to get information of the RTP port numbers
- reading SIP can be omitted by providing the RTP port numbers, so that any RTP traffic can be fuzzed
- RTCP traffic can be suppressed to avoid that codecs learn about the “noisy line”
- special care is taken to break RTP handling itself
- the RTP payload is fuzzed with a constant BER
- the BER is configurable
- requires arpspoof from dsniff to do the MITM attack
- requires both phones to be in a switched LAN (GW operation only works partially)

USAGE n/a

EXAMPLE n/a

protos-sip

DESCRIPTION **PROTOS SIP** test suite is designed to find vulnerabilities in software written for SIP entities. Using this test suite, you can verify the robustness of your software (whether your software can handle incorrect / currupt messages being received from the network). If your software has responded to the test case in a normal fashion(without crashing), then the test case is passed. Test suite doesn't log the test results, you have to observe your software's behaviour to check the test result.

Multiple Cisco products contain vulnerabilities in the processing of Session Initiation Protocol (SIP) INVITE messages. These vulnerabilities were identified by the University of Oulu Secure Programming Group (OUSPG) "**PROTOS" Test Suite for SIP** and can be repeatedly exploited to produce a denial of service.

USAGE n/a

OPTIONS You can enable "-showsent", "-showreply" command line options in the test suite, to check the SIP messages exchanged.

EXAMPLE n/a

rtpbreak

DESCRIPTION With **rtpbreak** you can detect, reconstruct and analyze any RTP session. It doesn't require the presence of RTCP packets and works independently form the used signaling protocol (SIP, H.323, SCCP, ...). The input is a sequence of packets, the output is a set of files you can use as input for other tools (**wireshark/tshark, sox, grep/awk/cut/cat/sed**, ...). It supports also wireless (AP_DLT_IEEE802_11) networks.

USAGE rtpbreak <input><output><select><execution><misc>
OPTIONS http://dallachiesa.com/code/rtpbreak/doc/rtpbreak_en.html

EXAMPLE rtpbreak -i wifi0 -g -m -d logz
EXAMPLE rtpbreak -P2 -t100 -T100 -d logz -r h323.pcap

TIP ***How can I extract audio from the recorded .pcap files?***
It is possible to extract audio using rtpbreak, sox and lame (for mp3).
First, the conversion of each channel in the raw:

```
rtpbreak -W -r longfilename_of_rtpproxy_record.a.rtp
rtpbreak -W -r longfilename_of_rtpproxy_record.o.rtp
```

Then mix in the format wavpcm stereo:

```
sox --combine merge -r 8k -A rtp.0.0.raw -r 8k -A rtp.1.0.raw -t wavpcm -s out.wav
```

and, finally, converting to mp3:

```
lame out.wav out.mp3
```

rtpflood

DESCRIPTION **rtpflood** is used to flood a target IP phone with a UDP packet containing a RTP data. In order to launch a successful attack using **rtpflood** you will need to know the RTP listening port on the remote device you want to attack, for example; x-lite sofphone default rtp port is 8000.

USAGE ./rtpflood sourcename destinationname srcport destport numpackets seqno timestamp SSID

EXAMPLE ./rtpflood 192.168.1.105 192.168.1.118 8000 8002 100000 15000 2000 18800532

rtpinsertsound

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

Here's a baby chameleon instead!

rtpmixsound

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

Here's a baby giraffe instead!

sctpscan

DESCRIPTION **SCTPscan** is a tool to scan SCTP endpoints. SCTP is a protocol that is used to carry SS7 over TCP/IP, among other things. It is part of the SIGTRAN protocol family, for SIGnalling TRANsport. It is a protocol like TCP with built-in support in major OS (Linux kernel 2.6, Solaris 10, FreeBSD 7, Mac OS X with kernel extension, ...). SCTP has some very interesting features (multihoming, multi-stream, resists well to Denial of Service - DoS, high performance). It's used for telecommunication backbone over IP (SS7 over IP aka SIGTRAN), Internet2 transfers, Cluster high-speed communication.

USAGE ./sctpscan [options]

OPTIONS ./sctpscan –h

EXAMPLE ./sctpscan -r 192.168.100.18

siparmyknife

DESCRIPTION **SIP Army Knife** is a fuzzer that searches for cross site scripting, SQL injection, log injection, format strings, buffer overflows, and more. **sipsak** − a utility for various tests on sip servers and user agents.

USAGE **sipsak [-dFGhiILnNMRSTUVvwz] [-a** *PASSWORD* **] [-b** *NUMBER* **] [-c** *SIPURI* **] [-C** *SIPURI* **] [-D** *NUMBER* **] [-e** *NUMBER* **] [-E** *STRING* **] [-f** *FILE* **] [-g** *STRING* **] [-H***HOSTNAME* **] [-l** *PORT* **] [-m** *NUMBER* **] [-o** *NUMBER* **] [-p** *HOSTNAME* **] [-P** *NUMBER* **] [-q** *REGEXP* **] [-r** *PORT* **] [-t** *NUMBER* **] [-u** *STRING* **] [-W** *NUMBER* **] [-x***NUMBER* **] -s** *SIPURI*

OPTIONS http://sipsak.org/man-page.html

EXAMPLE *sipsak -vv -s sip:nobody@foo.bar (Send an OPTIONS request to nobody@foo.bar and display received replies)*

EXAMPLE *sipsak -T -s sip nobody@foo.bar (Trace the SIP path to nobody@foo.bar)*

EXAMPLE *sipsak -U -C sip:me@home -x 3600 -a password -s sip:myself@company (Insert a forwarding contact for myself at work to me at home for one hour and authenticated with password if required)*

EXAMPLE *sipsak -I -C empty -a password -s sip:myself@work (Query the currently registered bindings for myself at work and authenticate with password if required)*

EXAMPLE *sipsak -M -v -s sip:colleaue@work -B "Lunch time!" (Send the instant message "Lunch time!" to the colleague and show result)*

sipp

DESCRIPTION **SIPp** is a free test tool and traffic generator for the SIP protocol. It uses XML format files to define test scenarios. It includes a few basic SipStone user agent scenarios (UAC and UAS) and establishes and releases multiple calls with the INVITE and BYE methods. More info: http://sipp.sourceforge.net/

*It can also read custom XML scenario files describing from very simple to complex call flows. It features the dynamic display of statistics about running tests (call rate, round trip delay, and message statistics), periodic CSV statistics dumps, TCP and UDP over multiple sockets or multiplexed with retransmission management and dynamically adjustable call rates. Other advanced features include support of IPv6, TLS, SCTP, SIP authentication, conditional scenarios, UDP retransmissions, error robustness (call timeout, protocol defense), call specific variable, Posix regular expression to extract and re-inject any protocol fields, custom actions (log, system command exec, call stop) on message receive, field injection from external CSV file to emulate live users. **SIPp** can also send media (RTP) traffic through RTP echo and RTP / pcap replay. Media can be audio or video. While optimized for traffic, stress and performance testing, **SIPp** can be used to run one single call and exit, providing a passed/failed verdict.*

USAGE sipp remote_host[:remote_port] [options]

OPTIONS http://tomeko.net/other/sipp/sipp_cheatsheet.php?lang=pl

EXAMPLE sipp 192.168.1.211 -sf OPTIONS.xml -m 5 -s 30 (Send OPTIONS message 5 times to 30@192.168.1.211)

EXAMPLE sipp 192.168.1.211 -sf OPTIONS_recv_200.xml -m 30 -s 30 (Send OPTIONS message 30 times to 30@192.168.1.211 waiting 200 ms for 200/OK reply each time)

sipsak

DESCRIPTION **sipsak** − a utility for various tests on sip servers and user agents.

USAGE **sipsak [-dFGhiILnNMRSTUVvwz] [-a** *PASSWORD* **] [-b** *NUMBER* **] [-c** *SIPURI* **] [-C** *SIPURI* **] [-D** *NUMBER* **] [-e** *NUMBER* **] [-E** *STRING* **] [-f** *FILE* **] [-g** *STRING* **] [-H***HOSTNAME* **] [-l** *PORT* **] [-m** *NUMBER* **] [-o** *NUMBER* **] [-p** *HOSTNAME* **] [-P** *NUMBER* **] [-q** *REGEXP* **] [-r** *PORT* **] [-t** *NUMBER* **] [-u** *STRING* **] [-W** *NUMBER* **] [-x***NUMBER* **] -s** *SIPURI*

OPTIONS http://sipsak.org/man-page.html

EXAMPLE *sipsak -vv -s sip:nobody@foo.bar* *(Send an OPTIONS request to nobody@foo.bar and display received replies)*

EXAMPLE *sipsak -T -s sip nobody@foo.bar* *(Trace the SIP path to nobody@foo.bar)*

EXAMPLE *sipsak -U -C sip:me@home -x 3600 -a password -s sip:myself@company* *(Insert a forwarding contact for myself at work to me at home for one hour and authenticated with password if required)*

EXAMPLE *sipsak -I -C empty -a password -s sip:myself@work* *(Query the currently registered bindings for myself at work and authenticate with password if required)*

EXAMPLE *sipsak -M -v -s sip:colleaue@work -B "Lunch time!"* (Send the instant message "Lunch time!" to the colleague and show result)

svcrack

DESCRIPTION **SIPVicious suite** is a set of tools that can be used to audit SIP based VoIP systems. It currently consists of four tools:

- svmap - this is a sip scanner. Lists SIP devices found on an IP range
- svwar - identifies active extensions on a PBX
- **svcrack - an online password cracker for SIP PBX**
- svreport - manages sessions and exports reports to various formats
- svcrash - attempts to stop unauthorized svwar and svcrack scans

SIPvicious password cracker is an online password guessing tool for SIP devices.

Read more: https://code.google.com/p/sipvicious/wiki/GettingStarted

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

svcrash

DESCRIPTION **SIPVicious suite** is a set of tools that can be used to audit SIP based VoIP systems. It currently consists of four tools:

- svmap - this is a sip scanner. Lists SIP devices found on an IP range
- svwar - identifies active extensions on a PBX
- svcrack - an online password cracker for SIP PBX
- svreport - manages sessions and exports reports to various formats
- **svcrash - attempts to stop unauthorized svwar and svcrack scans**

Svcrash sends a SIP message response to svwar.py which triggers an unhandled exception. This may allow victims of SIP floods due to attackers using svwar.py to mitigate the attack temporarily. The bug in svwar.py was also fixed. Additionally, the behavior that allowed it to keep sending messages even when not responses are received was also changed.

Read more: https://code.google.com/p/sipvicious/wiki/GettingStarted

USAGE n/a

EXAMPLE n/a

svmap

DESCRIPTION **SIPVicious suite** is a set of tools that can be used to audit SIP based VoIP systems. It currently consists of four tools:

- **svmap - this is a sip scanner. Lists SIP devices found on an IP range**
- svwar - identifies active extensions on a PBX
- svcrack - an online password cracker for SIP PBX
- svreport - manages sessions and exports reports to various formats
- svcrash - attempts to stop unauthorized svwar and svcrack scans

Read more: https://code.google.com/p/sipvicious/wiki/GettingStarted

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

svreport

DESCRIPTION **SIPVicious suite** is a set of tools that can be used to audit SIP based VoIP systems. It currently consists of four tools:

- svmap - this is a sip scanner. Lists SIP devices found on an IP range
- svwar - identifies active extensions on a PBX
- svcrack - an online password cracker for SIP PBX
- **svreport - manages sessions and exports reports to various formats**
- svcrash - attempts to stop unauthorized svwar and svcrack scans

Read more: https://code.google.com/p/sipvicious/wiki/GettingStarted

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

svwar

DESCRIPTION **SIPVicious suite** is a set of tools that can be used to audit SIP based VoIP systems. It currently consists of four tools:

- svmap - this is a sip scanner. Lists SIP devices found on an IP range
- **svwar - identifies active extensions on a PBX**
- svcrack - an online password cracker for SIP PBX
- svreport - manages sessions and exports reports to various formats
- svcrash - attempts to stop unauthorized svwar and svcrack scans

Read more: https://code.google.com/p/sipvicious/wiki/GettingStarted

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

voiphopper

DESCRIPTION **VoIP Hopper** is a GPLv3 licensed security tool, written in C, that rapidly runs a VLAN Hop security test. VoIP Hopper is a VoIP infrastructure security testing tool but also a tool that can be used to test the (in)security of VLANs.

USAGE voiphopper [options] <interface> [options] <mac>

OPTIONS http://voiphopper.sourceforge.net

EXAMPLE voiphopper -i eth0 –z (interactive assessment mode)

EXAMPLE voiphopper -i eth0 -o 00:50:60:03:99:CB (LLDP-MED spoofing: You can spoof LLDP-MED packets to quickly learn the Voice VLAN ID)

EXAMPLE voiphopper -i eth0 -c 0 -m AA:AA:AA:AA:AA:AA (spoof the MAC address of an IP Phone by sniffing for CDP (this changes the MAC address of default interface and new interface))

EXAMPLE voiphopper -d eth0.200 (delete the VoIP interface (eth0.200) created by VoIP Hopper)

[49] WEB SNIFFERS

- burpsuite
- dnsspoof
- driftnet
- ferret
- mitmproxy
- urlsnarf
- webmitm
- webscarab
- webspy
- zaproxy

burpsuite

DESCRIPTION **Burp Suite** is an integrated platform for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Burp Suite *contains the following key components:*

- *An intercepting **Proxy**, which lets you inspect and modify traffic between your browser and the target application.*
- *An application-aware **Spider**, for crawling content and functionality.*
- *An advanced web application **Scanner**, for automating the detection of numerous types of vulnerability.*
- *An **Intruder** tool, for performing powerful customized attacks to find and exploit unusual vulnerabilities.*
- *A **Repeater** tool, for manipulating and resending individual requests.*
- *A **Sequencer** tool, for testing the randomness of session tokens.*
- *The ability to **save your work** and resume working later.*
- ***Extensibility**, allowing you to easily write your own plugins, to perform complex and highly customized tasks within Burp.*

More info: http://portswigger.net/burp/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

dnsspoof

DESCRIPTION **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.

USAGE **dnsspoof** [**-i** *interface*] [**-f** *hostsfile*] [*expression*]

OPTIONS

-i interface Specify the interface to use.

-f hostsfile Specify the pathname of a file in hosts(5) format. Only one hostname allowed per line (no aliases), although hostnames may contain wildcards (such as *.doubleclick.net).

expression Specify a tcpdump(8) filter expression to select traffic to sniff.

If no hostsfile is specified, replies will be forged for all address queries on the LAN with an answer of the local machine's IP address.

EXAMPLE

```
# echo 1 > /proc/sys/net/ipv4/ip_forward (enable port forwarding)
# arpspoof -t 192.168.1.245 192.168.1.5 &;
# arpspoof -t 192.168.1.5 192.168.1.245 &;
# dnsspoof -f  spoofhosts.txt host  192.168.1.245  and  udp  port 53
```

driftnet

DESCRIPTION **Driftnet** watches network traffic, and picks out and displays JPEG and GIF images for display. It is an horrific invasion of privacy and shouldn't be used by anyone anywhere. It is also possible to use driftnet to capture MPEG audio data from the network and play it through a player such as **mpg123**. *Images may be saved by clicking on them.*

USAGE driftnet [*options*] [*filter code*]

OPTIONS

-h Print a summary of usage.

-v Print additional details of packets captured to the terminal.

-i interface Listen to packets on interface. By default, driftnet will try to pick up traffic on all interfaces, but this does not work with all versions of pcap(3); on such systems, an interface must be specified. On some systems, driftnet can only use promiscuous mode if an interface is specified.

-p Do not put the interface into promiscuous mode.

-a Operate in 'adjunct mode', where driftnet gathers images for use by another program, such as Jamie Zawinski's webcollage. In this mode, no window is displayed; images are captured and saved in a temporary directory, and their names written on standard output.

-m number In adjunct mode, silently drop images if there are more than number in the temporary directory. It is assumed that another process will delete images which it has processed.

-x prefix The filename prefix to use when saving images, by default 'driftnet-'.

-d directory Use directory to store temporary files. Driftnet will clear this directory of its own temporary files on exit, but will not delete the directory or any other files.

-s Attempt to capture streamed audio data from the network, and either play it or, in adjunct mode, save it in files. At present this only works with MPEG data.

-S Capture streamed audio data only, ignoring images.

-M command Use the named command to play MPEG audio data. The command, which is executed using the shell, should accept MPEG frames on standard input. The default is 'mpg123 -'.

filter code Additional filter code to restrict the packets captured, in the libpcap syntax. User filter code is evaluated as 'tcp and (filter code)'.

EXAMPLE driftnet -i wlan0

TIP use it together with **arpspoof**

ferret

DESCRIPTION **Ferret** is a copy-detection tool, locating duplicate text or code in multiple text documents or source files. **Ferret** is designed to detect copying (*collusion*) within a given set of files, and distinguishes copying unique to a pair of documents, across a group of documents, and against provided template material.
More info: https://github.com/petercrlane/ferret

USAGE ferret [-h] [-d] [-l] [-a] [-r] [-w] [-p] [-x] [-f] [-u]

OPTIONS

```
-h, --help            displays help on command-line parameters
-d, --data-table      produce similarity table (default)
-l, --list-trigrams   produce trigram list report
-a, --all-comparisons produce list of all comparisons
-r, --remove-common   removes common trigrams
-p, --pdf-report      source-1 source-2 results-file : create pdf report
-x, --xml-report      source-1 source-2 results-file : create xml report
-f, --definition-file use file with document list
-u, --use-stored-data store/retrieve data structure
```

EXAMPLE n/a

mitmproxy

DESCRIPTION **mitmproxy** is an SSL-capable man-in-the-middle HTTP proxy. With **mitmproxy** you can *Intercept HTTP requests and responses and modify them on the fly, Save complete HTTP conversations for later replay and analysis, Replay the client-side of an HTTP conversations, Replay HTTP responses of a previously recorded server, Reverse proxy mode to forward traffic to a specified server, Transparent proxy mode on OSX and Linux, Make scripted changes to HTTP traffic using Python, SSL certificates for interception are generated on the fly, And much, much more.*

More info: mitmproxy.org and https://github.com/cortesi/mitmproxy

USAGE n/a

OPTIONS

```
   [-localHost <host name/ip>]  Default is localhost
   [-localPort <port>]          Default is 8001
   [-keyStore <file>]           Key store details for
   [-keyStorePassword <pass>]   certificates. Equivalent to
   [-keyStoreType <type>]       javax.net.ssl.XXX properties
   [-keyStoreAlias <alias>]     Default is keytool default of 'mykey'
   [-outputFile <filename>]     Default is stdout
   [-v ]                        Verbose proxy output
   [-h ]                        Print this message
```

EXAMPLE n/a

urlsnarf

DESCRIPTION **urlsnarf** outputs all requested URLs sniffed from HTTP traffic in CLF (Common Log Format, used by almost all web servers), suitable for offline post-processing with your favorite web log analysis tool (analog, wwwstat, etc.).

USAGE **urlsnarf** [**-n**] [**-i** *interface*] [[**-v**] *pattern [expression*]]

OPTIONS

-n	Do not resolve IP addresses to hostnames.
-i interface	
-v	"Versus" mode. Invert the sense of matching, to select non-matching URLs. Specify the interface to listen on.
pattern	Specify regular expression for URL matching.
expression	Specify a tcpdump(8) filter expression to select traffic to sniff.

EXAMPLE urlsnarf -i lo

webmitm

DESCRIPTION **webmitm** transparently proxies and sniffs HTTP / HTTPS traffic redirected by **dnsspoof**, capturing most "secure" SSL-encrypted webmail logins and form submissions.

USAGE webmitm [-d] [host]

OPTIONS

-d	Enable debugging mode. May be specified multiple times to greater effect
host	Specify a host to proxy to. If none given, only requests containing an HTTP/1.1 Host: header or absolute URI will be relayed transparently

EXAMPLE webmitm -d

webscarab

DESCRIPTION **WebScarab** is a framework for analysing applications that communicate using the HTTP and HTTPS protocols. It is written in Java, and is thus portable to many platforms. **WebScarab** has several modes of operation, implemented by a number of plugins. In its most common usage, **WebScarab** operates as an intercepting proxy, allowing the operator to review and modify requests created by the browser before they are sent to the server, and to review and modify responses returned from the server before they are received by the browser. **WebScarab** is able to intercept both HTTP and HTTPS communication. The operator can also review the conversations (requests and responses) that have passed through **WebScarab**.

More info: https://www.owasp.org/index.php/WebScarab_Getting_Started

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

webspy

DESCRIPTION **webspy** sends URLs sniffed from a client to your local Netscape browser for display, updated in real-time (as the target surfs, your browser surfs along with them, automagically). Netscape must be running on your local X display ahead of time.

USAGE webspy [-i interface | -p pcapfile] host

OPTIONS

-i interface (Specify the interface to listen on)

-p pcapfile (Process packets from the specified PCAP capture file instead of the network)

Host (Specify the web client to spy on)

EXAMPLE webspy -i eth0 192.168.1.66

zaproxy

DESCRIPTION The **OWASP Zed Attack Proxy (ZAP)** is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications. It is designed to be used by people with a wide range of security experience and as such is ideal for developers and functional testers who are new to penetration testing as well as being a useful addition to an experienced pen testers' toolbox.

More info: https://code.google.com/p/zaproxy/

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[50] MAINTAINING ACCESS: OS BACKDOORS

- cymothoa
- dbd
- intersect
- powersploit
- sbd
- u3-pwn

cymothoa

DESCRIPTION **Cymothoa** is a stealth backdooring tool, that inject backdoor's shellcode into an existing process. The tool uses the **ptrace** library (available on nearly all * nix), to manipulate processes and infect them.

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

OPTIONS http://cymothoa.sourceforge.net/

EXAMPLE cymothoa –S
EXAMPLE cymothoa -p 1140 -s 0 -y 9000

dbd

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

intersect

DESCRIPTION **Intersect** is a Linux post-exploitation framework written in Python. The main goal of this project is to assist penetration testers in the automation of many post exploitation and data exfiltration tasks that they would otherwise perform manually.

With the Intersect framework, users can easily build their own customized scripts from the pre-built templates and modules that are provided or they can write their own modules to add additional or specialized functionality.

The framework is centred around the **Create** application. This is where users can build their Intersect scripts, control the available modules and import their own. Using the Create application provides a very straight forward, menu-driven process for interacting with the framework.

USAGE n/a

OPTIONS http://ohdae.github.io/Intersect-2.5/#Intro

EXAMPLE n/a

powersploit

DESCRIPTION **PowerSploit** is a collection of Microsoft PowerShell modules that can be used to aid reverse engineers, forensic analysts, and penetration testers during all phases of an assessment. More info: https://github.com/mattifestation/PowerSploit#readme

TIP try it with **Metasploit** http://obscuresecurity.blogspot.co.uk/2013/03/powersploit-metasploit-shells.html

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

sbd

DESCRIPTION **SBD** expands to storage-based death, and is named in reference to Novell's Cluster Services, which used **SBD** to exchange poison pill messages.

The **sbd** daemon, combined with the **external/sbd STONITH** agent, provides a way to enable **STONITH** and fencing in clusters without external power switches, but with shared storage.

The **sbd** daemon runs on all nodes in the cluster, monitoring the shared storage. When it either loses access to the majority of **sbd** devices, or sees that another node has written a fencing request to its mailbox slot, the node will immediately fence itself.

sbd can be used in virtual environments where the hypervisor layer is not cluster-enabled, but a shared storage device between the guests is available

More info: http://doc.opensuse.org/products/draft/SLE-HA/SLE-ha-guide_sd_draft/cha.ha.fencing.html , http://doc.opensuse.org/products/draft/SLE-HA/SLE-ha-guide_sd_draft/cha.ha.storage.protect.html and http://www.linux-ha.org/wiki/SBD_Fencing

USAGE n/a

EXAMPLE sbd -d /dev/sbd dump

EXAMPLE sbd -d /dev/sbd message nodea test

u3-pwn

DESCRIPTION **U3-Pwn** is a tool designed to automate injecting executables to Sandisk smart usb devices with default U3 software install. This is performed by removing the original iso file from the device and creating a new iso with autorun features.

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

[51] TUNNELING TOOLS

- cryptcat
- dbd
- dns2tcpc
- dns2tcpd
- iodine
- miredo
- ncat
- proxychains
- proxytunnel
- ptunnel
- pwnat
- sbd
- socat
- sslh
- stunnel4
- updtunnel

cryptcat

DESCRIPTION **Cryptcat** is a simple Unix utility which reads and writes data across network connections, using TCP or UDP protocol while encrypting the data being transmitted. It is designed to be a reliable "back-end" tool that can be used directly or easily driven by other programs and scripts. At the same time, it is a feature-rich network debugging and exploration tool, since it can create almost any kind of connection you would need and has several interesting built-in capabilities.

USAGE connect to somewhere: nc [-options] hostname port[s] [ports] ...

USAGE listen for inbound: nc -l -p port [-options] [hostname] [port]

EXAMPLE cryptcat –h

OPTIONS

-g gateway	source-routing hop point[s], up to 8
-G num	source-routing pointer: 4, 8, 12, ...
-h	this cruft
-i secs	delay interval for lines sent, ports scanned
-l	listen mode, for inbound connects
-n	numeric-only IP addresses, no DNS
-o file	hex dump of traffic
-p port	local port number
-r	randomize local and remote ports
-s addr	local source address
-u	UDP mode
-v	verbose [use twice to be more verbose]
-w secs	timeout for connects and final net reads
-z	zero-I/O mode [used for scanning]

dbd

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

dns2tcpc

DESCRIPTION **dns2tcpc** - A tunnelling tool that encapsulate TCP connections in DNS.

Dns2tcp is a network tool designed to relay TCP connections through DNS traffic. Encapsulation is done on the TCP level, thus no specific driver is needed (i.e., TUN/TAP). **Dns2tcp** is composed of two parts : a server-side tool and a client-side tool. The server has a list of resources specified in a configuration file. Each resource is a local or remote service listening for TCP connections. The client listen on a predefined TCP port and relays each incoming connection through DNS to the final service.

USAGE dns2tcpc -z <dns2tcp_server> [options] [server] (client)

OPTIONS http://www.aldeid.com/wiki/Dns2tcp (clients & server)

EXAMPLE dns2tcpd -d 1 -f ~/.dns2tcpdrc (start the dns2tcpd server as a daemon)

EXAMPLE dns2tcpc -z dns2tcp.aldeid.com -d 3 -k oopsoops (check available resources)

dns2tcpc -z dns2tcp.aldeid.com -k oopsoops -r ssh -l 2222 (select a resource and open a connection)

ssh me@127.0.0.1 -p 2222 -D 8080 (ensure the connection will be encrypted)

dns2tcpd

DESCRIPTION **dns2tcpd** - A tunnelling tool gateway that encapsulate TCP connections in DNS.

Dns2tcp is a network tool designed to relay TCP connections through DNS traffic. Encapsulation is done on the TCP level, thus no specific driver is needed (i.e: TUN/TAP). **Dns2tcp** is composed of two parts : a server-side tool and a client-side tool. The server has a list of resources specified in a configuration file. Each resource is a local or remote service listening for TCP connections. The client listen on a predefined TCP port and relays each incoming connection through DNS to the final service.

USAGE dns2tcpd [-h] [-F] [-i address] [-f config_file] [-d debug_level]

OPTIONS

-h Help Menu

-F Run in foreground

-d debug level Change debug level. Levels available are 1, 2 or 3.

-i IP address IP address to bind (default 0.0.0.0)

-f config file Configuration file to use

EXAMPLE http://manpages.ubuntu.com/manpages/lucid/man1/dns2tcpd.1.html

EXAMPLE http://zitstif.no-ip.org/?tag=dns2tcpd

iodine

DESCRIPTION **iodine** lets you tunnel IPv4 data through a DNS server. This can be useful in situations where Internet access is firewalled, but DNS queries are allowed. It needs a TUN/TAP device to operate. The bandwidth is asymmetrical, with a measured maximum of 680 kbit/s upstream and 2.3 Mbit/s downstream in a wired LAN test network. Realistic sustained throughput on a Wifi network using a carrier-grade DNS cache has been measured at some 50 kbit/s upstream and over 200 kbit/s downstream.

iodine is the client application, **iodined** is the server. More info: http://dev.kryo.se/iodine/wiki/HowtoSetup

Note: server and client are required to speak the exact same protocol. In most cases, this means running the same iodine version. Unfortunately, implementing backward and forward protocol compatibility is usually not feasible.

USAGE iodine (client) and iodined [-v] (server)

USAGE iodine (client) and iodined [-h] (server)

USAGE iodine (client) and iodined [-c] [-s] [-f] [-D] [-u *user*] [-t *chrootdir*] [-d *device*] [-m *mtu*] [-l *listen_ip*] [-p *port*] [-n *external_ip*] [-b *dnsport*] [-P *password*] [-z *context*] [-F *pidfile*] *tunnel_ip* [*/netmask*] *topdomain*] (server)

OPTIONS http://linux.die.net/man/8/iodine

EXAMPLE ./iodined -fP test 10.0.0.1 test.asdf (server)

EXAMPLE ./iodine -fP test 192.168.0.1 test.asdf (client)

miredo

DESCRIPTION **Miredo** is a daemon program providing a Teredo tunnel service compatible with the "Teredo: Tunneling IPv6 over UDP through NATs" Internet proposed standard (RFC 4380). It can provide either Teredo client or Teredo relay functionality.

This is mostly useful to provide IPv6 connectivity to users behind NAT, most of which do not support IPv6 at all. Many NATs do not even support proto-41 forwarding, so it is not possible to set up a 6to4 or point-to-point IPv6-over-IPv4 tunnel through them. A Teredo relay is an IPv6 router which forwards IPv6 packets between the IPv6 Internet and Teredo clients by encapsulating these IPv6 packets over UDP/IPv4. A Teredo client is an IPv6-enabled host which is located behind an IPv4-only Network Address Translator (a.k.a. NAT), and encapsulates its IPv6 traffic inside UDP over IPv4 packets. A Teredo server is a special Teredo relay which is required for Teredo clients to setup their IPv6 connectivity through Teredo. A Teredo server must have to global static subsequent IPv4 addresses. It receives packets from Teredo clients and Teredo relays on UDP port 3544.

USAGE miredo [-c config_file] [-f] [-u user] [-t chrootdir] [server_name]

USAGE miredo [OPTIONS] [SERVER_NAME]

OPTIONS http://linux.die.net/man/8/miredo

EXAMPLE miredo -u miredo

ncat

DESCRIPTION **ncat** is a general-purpose command-line tool for reading, writing, redirecting, and encrypting data across a network. It aims to be your network Swiss Army knife, handling a wide variety of security testing and administration tasks.

Ncat can:

- Act as a simple TCP/UDP/SCTP/SSL client for interacting with web/telnet/mail/TCP/IP servers and services
- Act as a simple TCP/UDP/SCTP/SSL server for offering services to clients, or simply to understand what existing clients are up to by capturing every byte they send.
- Redirect or proxy TCP/UDP/SCTP traffic to other ports or hosts.
- Encrypt communication with SSL, and transport it over IPv4 or IPv6.
- Act as a network gateway for execution of system commands, with I/O redirected to the network.
- Act as a connection broker, allowing two (or far more) clients to connect to each other through a third (brokering) server.

USAGE ncat [options] <url>

EXAMPLE ncat -C mail.example.com 25 (sending email to an SMTP server. Read manual for further steps)

EXAMPLE ncat -l localhost 143 --sh-exec "ncat --ssl imap.example.com 993“ (connecting to an IMPA server that requires SSL . Read manual for further steps)

proxychains

DESCRIPTION **proxychains** - a tool that forces any TCP connection made by any given application to follow through proxy like TOR or any other SOCKS4, SOCKS5 or HTTP(S) proxy. Supported auth-types: "user/pass" for SOCKS4/5, "basic" for HTTP. More info: http://proxychains.sourceforge.net/howto.html

proxyresolv - DNS resolving. Used to resolve host names via proxy or TOR.

USAGE type host port [user pass]

EXAMPLE proxychains telnet targethost.com (in this example it will run telnet through proxy(or chained proxies) specified by proxychains.conf)
EXAMPLE proxyresolv targethost.com (in this example it will resolve targethost.com through proxy(or chained proxies) specified by proxychains.conf)

proxytunnel

DESCRIPTION **proxytunnel** is a program that opens a tunnel through a HTTPS proxy.
More info: http://linux.die.net/man/1/proxytunnel

USAGE **proxytunnel** [*options*]

OPTIONS

-h, --help Print help and exit.
-V, --version kPrint the version of the program and exit.
-i, --inetd Run from inetd. Default is off.
-F STRING, --passfile=STRING The file containing Username & Password to send to HTTPS proxy for authentification. This file uses the same format as .wgetrc, and so can use the credentials in common with wget. This option can be used to at least hide the password from anyone clever enough to use the 'ps' command.
-p STRING, --proxy=STRING The HTTPS Proxy host:port combo to connect to.
-P STRING, --proxyauth=STRING The credentials (user:pass) to use for local HTTP(S) proxy authentication.
-d STRING, --dest=STRING The destination host:port to built the tunnel to.
-r STRING, --remproxy=STRING The second-level proxy host:port to connect to.
-R STRING, --remproxyauth=STRING The credentials (user:pass) to use for remote HTTP(S) proxy authentication.
-v, --verbose Turn on verbosity. Default is off.
-q, --quiet Suppress messages. Default is off.

EXAMPLE proxytunnel -p proxy.customer.com:8080 -u user -s password -d mybox.athome.nl:443

ptunnel

DESCRIPTION Ptunnel is an application that allows you to reliably tunnel TCP connections to a remote host using ICMP echo request and reply packets, commonly known as ping requests and replies.

At first glance, this might seem like a rather useless thing to do, but it can actually come in handy in some cases. The following example illustrates the main motivation in creating ptunnel:

Setting: You're on the go, and stumble across an open wireless network. The network gives you an IP address, but won't let you send TCP or UDP packets out to the rest of the internet, for instance to check your mail. What to do? By chance, you discover that the network will allow you to ping any computer on the rest of the internet. With ptunnel, you can utilize this feature to check your mail, or do other things that require TCP.

More info: https://github.com/madeye/ptunnel

USAGE Client: ./ptunnel -p -lp -da -dp [-c] [-v] [-u] [-x password] Proxy: ./ptunnel [-c] [-v] [-u] [-x password]

EXAMPLE

The following assumes that ptunnel is run as root, both on the proxy and client. To tunnel ssh connections from the client machine via a proxy running on proxy.pingtunnel.com to the computer login.domain.com, the following command line would be used:

ptunnel −p proxy.pingtunnel.com −lp 8000 −da login.domain.com −dp 22

An ssh connection to login.domain.com can now be established as follows:

ssh −p 8000 localhost

If ssh complains about potential man-in-the-middle attacks, simply remove the offending key from the known_hosts file. The warning/error is expected if you have previously ssh'd to your local computer (i.e., ssh localhost), or you have used ptunnel to forward ssh connections to different hosts.

Of course, for all of this to work, you need to start the proxy on your proxy-computer (proxy.pingtunnel.com). Doing this is very simple:

ptunnel

pwnat

DESCRIPTION **pwnat** punches holes in firewalls and NATs allowing any numbers of clients behind NATs to directly connect to a server behind a different NAT with no 3rd party, port forwarding, DMZ or spoofing involved. This will allow you to tunnel any service that you want to run (http, ssh, quake server, IRC, ftp, etc.) through your NAT, or proxy into other remote servers. More info: https://github.com/samyk/pwnat

USAGE ./pwnat <-s | -c> <args>

USAGE <args>: [local ip] <local port> <proxy host> [proxy port (def:2222)] <remote host> <remote port>

USAGE <args>: [local ip] [proxy port (def:2222)] [[allowed host]:[allowed port] ...]

OPTIONS

-c client mode

-s server mode

-6 use IPv6

-v show debug output (up to 2)

-h show this help and exit

EXAMPLE http://www.sumitgupta.net/pwnat-example/

sbd

DESCRIPTION **SBD** expands to storage-based death, and is named in reference to Novell's Cluster Services, which used **SBD** to exchange poison pill messages.

The **sbd** daemon, combined with the **external/sbd STONITH** agent, provides a way to enable **STONITH** and fencing in clusters without external power switches, but with shared storage.

The **sbd** daemon runs on all nodes in the cluster, monitoring the shared storage. When it either loses access to the majority of **sbd** devices, or sees that another node has written a fencing request to its mailbox slot, the node will immediately fence itself.

sbd can be used in virtual environments where the hypervisor layer is not cluster-enabled, but a shared storage device between the guests is available

More info: http://doc.opensuse.org/products/draft/SLE-HA/SLE-ha-guide_sd_draft/cha.ha.fencing.html , http://doc.opensuse.org/products/draft/SLE-HA/SLE-ha-guide_sd_draft/cha.ha.storage.protect.html and http://www.linux-ha.org/wiki/SBD_Fencing

USAGE n/a

EXAMPLE sbd -d /dev/sbd dump

EXAMPLE sbd -d /dev/sbd message nodea test

socat

DESCRIPTION **socat** is a relay for bidirectional data transfer between two independent data channels. *Each of these data channels may be a file, pipe, device (serial line etc. or a pseudo terminal), a socket (UNIX, IP4, IP6 - raw, UDP, TCP), an SSL socket, proxy CONNECT connection, a file descriptor (stdin etc.), the GNU line editor (readline), a program, or a combination of two of these. These modes include generation of "listening" sockets, named pipes, and pseudo terminals.* ***socat*** *can be used, e.g., as TCP port forwarder (one-shot or daemon), as an external socksifier, for attacking weak firewalls, as a shell interface to UNIX sockets, IP6 relay, for redirecting TCP oriented programs to a serial line, to logically connect serial lines on different computers, or to establish a relatively secure environment (su and chroot) for running client or server shell scripts with network connections.*

More info: http://www.dest-unreach.org/socat/doc/README

USAGE socat [options] <address> <address>
USAGE socat -V
USAGE socat -h[h[h]] | -?[?[?]]

OPTIONS http://www.dest-unreach.org/socat/doc/socat.html#OPTIONS

EXAMPLE **socat - TCP4:www.domain.org:80**
EXAMPLE **socat TCP4-LISTEN:www TCP4:www.domain.org:www**
EXAMPLE **socat -,raw,echo=0,escape=0x0f /dev/ttyS0,raw,echo=0,crnl**
EXAMPLE http://www.dest-unreach.org/socat/doc/socat.html#EXAMPLES

sslh

DESCRIPTION **sslh** lets one accept both HTTPS and SSH connections on the same port. It makes it possible to connect to an SSH server on port 443 (e.g. from inside a corporate firewall, which almost never block port 443) while still serving HTTPS on that port. The idea is to have **sslh** listen to the external 443 port, accept the incoming connections, work out what type of connection it is, and then forward to the appropriate server.

USAGE sslh [**−t** ***num***] [**−p** ***listening address***] [**−l** ***target address for SSL***] [**−s** ***target address for SSH***] [**−u** ***username***] [**−P** ***pidfile***] [−v] [−V]

OPTIONS http://manpages.ubuntu.com/manpages/lucid/man8/sslh.8.html

EXAMPLE

```
# configure it in /etc/default/sslh
RUN=yes
DAEMON_OPTS="-u sslh -p 0.0.0.0:443 -s 127.0.0.1:_YOURSSHPORT_ -l 127.0.0.1:443 -P /var/run/sslh.pid“
# start it
/etc/init.d/sslh start
```

stunnel4

DESCRIPTION The **stunnel** program is designed to work as SSL encryption wrapper between remote clients and local (inetd-startable) or remote servers. The concept is that having non-SSL aware daemons running on your system you can easily set them up to communicate with clients over secure SSL channels. **stunnel** can be used to add SSL functionality to commonly used Inetd daemons like POP-2, POP-3, and IMAP servers, to standalone daemons like NNTP, SMTP and HTTP, and in tunnelling PPP over network sockets without changes to the source code.

USAGE stunnel [<filename>] | -fdn | -help | -version | -sockets

OPTIONS http://man.he.net/man8/stunnel4

EXAMPLE

In order to provide SSL encapsulation to your local imapd service, use

```
[imapd]
accept = 993
exec = /usr/sbin/imapd
execargs = imapd
```

If you want to provide tunneling to your pppd daemon on port 2020, use

```
[vpn]
accept = 2020
exec = /usr/sbin/pppd
execargs = pppd local
pty = yes
```

updtunnel

DESCRIPTION This project tunnels TCP data through a UDP tunnel. The executable can act the server or client. The server acts as a proxy for the client, listening on a specified UDP port and creating a connection to a TCP server that the client specifies. The client listens on a TCP port, acting as the server that some TCP client connects to. The client receives any TCP data on that port and sends the data to the **udpserver**, which sends it to the TCP connection it made with the desired TCP server.

USAGE ./udptunnel -<s|c> [-6] <args>
USAGE udptunnel -s [-6] [host] port
USAGE udptunnel -c [-6] [local host] <local port> <proxy host> <proxy port> <remote host> <remote port>
OPTIONS https://github.com/samyk/pwnat/blob/master/README-udptunnel

EXAMPLE

Example for tunneling ssh data through the tunnel between two computers with IP addresses 192.168.1.2 (client) and 192.168.1.1 (server):

server# ./udptunnel -s 192.168.1.1 4444
client# ./udptunnel -c 127.0.0.1 3333 192.168.1.1 4444 127.0.0.1 22
client# ssh -p 3333 user@127.0.0.1

[52] WEB BACKDOORS

- webacoo
- weevely

webacoo

DESCRIPTION **WeBaCoo** - Web Backdoor Cookie Script-Kit. aiming to provide a stealth terminal-like connection over HTTP between client and web server. It is a post exploitation tool to maintain access to a compromised web server. ***WeBaCoo*** *was designed to operate under the radar of modern up-to-dated AV, NIDS, IPS, Network Firewalls and Application Firewalls, proving a stealth mechanism to execute commands to the compromised server. The obfuscated communication is accomplished using HTTP header's Cookie fields under valid client HTTP requests and relative web server's responses.*

USAGE webacoo.pl [options]

OPTIONS https://github.com/anestisb/WeBaCoo

EXAMPLE ./webacoo.pl -g -o backdoor.php (Create 'backdoor.php' obfuscated backdoor with default settings)

EXAMPLE ./webacoo.pl -g -o raw-backdoor.php -f 4 -r (Create 'raw-backdoor.php' un-obfuscated backdoor using 'passthru' function)

EXAMPLE ./webacoo.pl -t -u http://127.0.0.1/backdoor.php (Establish "terminal" connection with remote host using the default setup)

EXAMPLE ./webacoo.pl -t -u http://127.0.0.1/backdoor.php -c "Test-Cookie" -d "TtT" (Establish "terminal" connection with remote host while setting some args)

EXAMPLE ./webacoo.pl -t -u http://10.0.1.13/backdoor.php -p 127.0.0.1:8080 (Establish "terminal" connection with remote host through local http proxy)

EXAMPLE ./webacoo.pl -t -u http://10.0.1.13/backdoor.php -p user:password:10.0.1.8:3128 (Establish "terminal" connection with remote host through http proxy with basic auth)

EXAMPLE ./webacoo.pl -t -u http://example.com/backdoor.php -p tor -l webacoo_log.txt (Establish "terminal" connection with remote host over Tor and log activity)

weevely

DESCRIPTION **Weevely** is a stealth PHP web shell that simulates an SSH-like connection. It is an essential tool for web application post exploitation, and can be used as stealth backdoor or as a web shell to manage legit web accounts, even free hosted ones. Just generate and upload the PHP code on the target web server, and run the **Weevely** client locally to transmit shell commands. More info: https://github.com/epinna/Weevely/wiki/Tutorial

USAGE ./weevely generate <password> [output path]
USAGE ./weevely <url> <password>
USAGE ./weevely <url> <password> "<command>"

OPTIONS ./weevely :help

EXAMPLE ./weevely.py generate p4ssw0rd (Server-side installation)
EXAMPLE ./weevely.py http://target.org/w.php p4ssw0rd "uname“
EXAMPLE ./weevely.py http://target.org/w.php p4ssw0rd ":system.info client_ip"

[53] REVERSE ENGINEERING: DEBUGGERS

- edb-debugger
- ollydbg

edb-debugger

DESCRIPTION **EDB (Evan's Debugger)** is a Qt4 based binary mode debugger with the goal of having usability on par with OllyDbg. It uses a plugin architecture, so adding new features can be done with ease.

USAGE n/a; GUI tool

EXAMPLE ./edb --symbols /lib/libc.so.6 > symbols/libc.so.6.map (installing a plugin)

ollydbg

DESCRIPTION **OllyDbg** is an x86 debugger that emphasizes binary code analysis, which is useful when source code is not available. It traces registers, recognizes procedures, API calls, switches, tables, constants and strings, as well as locates routines from object files and libraries. It has a friendly interface, and its functionality can be extended by third party plugins. **OllyDbg** is often used for reverse engineering of programs. It is often used by crackers to crack software made by other developers. For cracking and reverse engineering, it is often the primary tool because of its ease of use and availability. It is also useful for programmers to ensure that their program is running as intended. Furthermore it can be used for malware analysis purposes as well.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[54] DISASSEMBLY

- jad
- rabin2
- radiff2
- rasm2
- recstudio
- recstudio-cli

jad

DESCRIPTION **Jad (Java Decompiler)** is a currently unmaintained decompiler for the Java programming language. **Jad** provides a command-line user interface to extract source code from class files. A graphical user interface for **Jad** is **JadClipse** which is a plugin to the Eclipse IDE.

USAGE n/a

EXAMPLE n/a

rabin2

DESCRIPTION **rabin2** — Binary program info extractor. This program allows you to get information about ELF/PE/MZ and CLASS files in a simple way.

USAGE **rabin2** [**-eoicsSlrvz**] [**-h**] [**-L** library] [file]

OPTIONS

-e	Show entrypoints for disk and on-memory
-I	Show imports (symbols imported from libraries
-c	Show header checksum (if exist)
-s	Show exported symbols
-o	Show other symbols (not import/export)
-S	Show sections
-l	List linked libraries to the binary
-L library	Show the base address of a library loaded in memory
-r	Show output in radare format
-z	Show strings inside .data section (like gnu strings does)
-v	Display virtual addressing offsets
-h	Show usage help message.

EXAMPLE rabin2 -s a.out (List symbols of a program)

EXAMPLE rabin2 \-e a.out (Get entrypoint)

EXAMPLE rabin2 \-vrsi a.out (Loads symbols and imports from radare)

radiff2

DESCRIPTION **radiff2** - unified binary diffing utility. radiff2 implements many binary diffing algorithms for data and code.

USAGE radiff2 [-cCOdrspvh] [-t 0-100] [-g sym] file1 file2

OPTIONS http://manpages.ubuntu.com/manpages/precise/man1/radiff2.1.html

EXAMPLE radiff2 -g 0x8034804 b a > b.dot
EXAMPLE radiff2 /bin/true /bin/false

rasm2

DESCRIPTION **rasm2** — radare2 patch assembler and disassembler. This tool uses r_asm to assemble and disassemble files or hexpair strings. It supports a large list of architectures which can be listed using the -L flag.

USAGE **rasm2** [**-dDfBCLev**] [**-F** in:out] [**-o** offset] [**-a** arch] [**-b** bits] [**-s** syntax] [**-l** int] [ARG]

OPTIONS http://manpages.ubuntu.com/manpages/precise/man1/rasm2.1.html

EXAMPLE rasm2 -a x86 -b 32 'mov eax, 33' (Assemble opcode)

EXAMPLE rasm2 \-d 90 (Disassemble opcode)

recstudio

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

Here's a baby hippo instead!

recstudio-cli

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

[55] MISC REVERSE ENGINEERING TOOLS

- apktool
- clang
- clang++
- dex2jar
- flasm
- javasnoop
- radare2
- rafind2
- ragg2
- ragg2-cc
- rahash2
- rarun2
- rax2

apktool

DESCRIPTION **APKTool** is an application which decompiles and recompiles android APKs. It 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; it makes possible to debug smali code step by step. Also it makes working with app easier because of project-like files structure and automation of some repetitive tasks like building apk, etc.

USAGE [q|--quiet OR –v|--verbose] COMMAND [...]

OPTIONS https://code.google.com/p/android-apktool/wiki/ApktoolOptions

EXAMPLE apktool if SystemUI.apk
EXAMPLE apktool d SystemUI.apk
EXAMPLE apktool b SystemUI almostdone.apk

clang

DESCRIPTION **The Clang Compiler** is an open-source compiler for the C family of programming languages, aiming to be the best in class implementation of these languages. **Clang** builds on the LLVM optimizer and code generator, allowing it to provide high-quality optimization and code generation support for many targets.
More info: http://clang.llvm.org

USAGE compile + link compile then link debug info enabling optimizations picking a language to use, defaults to C99 by default. Autosenses based on extension. using a makefile

OPTIONS http://clang.llvm.org/docs/UsersManual.html

EXAMPLE clang -x c-header test.h -o test.h.pch
EXAMPLE clang test.c -o test
EXAMPLE clang -include test.h test.c -o test

clang++

DESCRIPTION **The Clang Compiler** is an open-source compiler for the C family of programming languages, aiming to be the best in class implementation of these languages. **Clang** builds on the LLVM optimizer and code generator, allowing it to provide high-quality optimization and code generation support for many targets.
More info: http://clang.llvm.org

USAGE http://developer.apple.com/library/mac/documentation/Darwin/Reference/ManPages/man1/clang++.1.html

OPTIONS http://clang.llvm.org/docs/UsersManual.html

EXAMPLE clang -x c-header test.h -o test.h.pch
EXAMPLE clang test.c -o test
EXAMPLE clang -include test.h test.c -o test

dex2jar

DESCRIPTION **dex2jar** a dex decompiler.

dex2jar contains 4 components:

- **dex-reader** is designed to read the Dalvik Executable (.dex/.odex) format. It has a light weight API similar with ASM. An example here
- **dex-translator** is designed to do the convert job. It reads the dex instruction to **dex-ir** format, after some optimize, convert to ASM format.
- **dex-ir** used by dex-translator, is designed to represent the dex instruction
- **dex-tools** tools to work with .class files.

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

flasm

DESCRIPTION **Flasm** disassembles your entire SWF including all the timelines and events. Looking at disassembly, you learn how the Flash compiler works, which improves your **ActionScript** skills. You can also do some optimizations on the disassembled code by hand or adjust the code as you wish. **Flasm** then applies your changes to the original SWF, replacing original actions.

*It's also possible to embed **Flasm** actions in your ActionScript, making optimizing of large projects more comfortable.*

***Flasm** is not a decompiler. What you get is the human readable representation of SWF bytecodes, not **ActionScript** source. If you're looking for a decompiler, **Flare** may suit your needs. However, **Flare** can't alter the SWF.*

More info: http://flasm.sourceforge.net/

USAGE flasm *option filename*

OPTIONS http://flasm.sourceforge.net/#usage

EXAMPLE flasm -d foo.swf (Disassemble foo.swf to the console)

EXAMPLE flasm -d foo.swf > foo.flm (Disassemble foo.swf, redirect the output to foo.flm)

EXAMPLE flasm -z foo.swf (Compress foo.swf, create .$wf backup. Source SWF doesn't have to be Flash MX file. However, only Flash MX and later players will be able to play the resulting compressed file.)

EXAMPLE flasm -x foo.swf (Decompress foo.swf, create .$wf backup)

javasnoop

DESCRIPTION **JavaSnoop** is a tool for testing (re: hacking) Java desktop applications or applets. More info; http://javasnoop.googlecode.com/svn-history/r32/trunk/resources/README.txt

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

radare2

DESCRIPTION **radare**- the reverse engineering framework. **Radare2** is an open source tool to disassemble, debug, analyze and manipulate binary files. **Radare** project started as a forensics tool, an scriptable commandline hexadecimal editor able to open disk files, but later support for analyzing binaries, disassembling code, debugging programs, attaching to remote gdb servers, ..

USAGE **radare2** [-**s** *addr*] [-**b** *bsize*] [-**e** *k=v*] [-**dwnLV**] *file*

OPTIONS http://www.makelinux.net/man/1/R/radare2

OPTIONS Type '?' for help.

OPTIONS To enter visual mode use the 'V' command. Then press '?' for help.

EXAMPLE r2 -c=H /bin/ls

rafind2

DESCRIPTION afind2 — Advanced commandline hexadecimal editor

USAGE rafind2 [-zXnrhv] [-b *size*] [-f *from*] [-t *to*] [-[m|s|e] *str*] [-x *hex*] *file*

OPTIONS http://manned.org/rafind2.1

EXAMPLE n/a

ragg2

DESCRIPTION **ragg2** — **radare2** utility to run programs in exotic environments.

ragg2 is a frontend for **r_egg**, compile programs into tiny binaries for x86-32/64 and ARM.

This tool is experimental and it is a rewrite of the old **rarc2** and **rarc2-tool** programs as a library and integrated with **r_asm** and **r_bin**.

*Programs generated by **r_egg** are relocatable and can be injected in a running process or on-disk binary file.*

***ragg2-cc** is another tool that comes with **r2** and it is used to generate shellcodes from C code. The final code can be linked with rabin2 and it is relocatable, so it can be used to inject it on any remote process.*

***ragg2-cc** is conceptually based on shellforge4, but only linux/osx x86-32/64 platforms are supported.*

USAGE ragg2 [-a *arch*] [-b *bits*] [-k *kernel*] [-f *format*] [-o *file*] [-i *shellcode*] [-I *path*] [-e *encoder*] [-B *hexpairs*] [-c *k=v*] [-C *file*] [-d *off:dword*] [-D *off:qword*] [-w *off:hexpair*] [-p *padding*] [-FOLsrxvh]

OPTIONS http://manned.org/ragg2.1

EXAMPLE ragg2 -O -F hi.r

EXAMPLE ragg2 hi.c

ragg2-cc

DESCRIPTION **ragg2-cc** - CC frontend for compiling shellcodes. The final code can be linked with **rabin2** and it is relocatable, so it can be used to inject it on any remote process. **ragg2-cc** is conceptually based on **shellforge4**, but only linux/osx x86-32/64 platforms are supported. **ragg2-cc** is a frontend of CC. It is used to creates tiny binaries (1KB) or shellcodes in binary or hexpairs from a C source. The compiler used is the one configured by the CC environment. This has been tested with **gcc**, **llvm-gcc** and **clang**.

USAGE **ragg2-cc** [**-a** arch] [**-b** bits] [**-k** kernel] [**-o** file] [**-dscxvh**]

OPTIONS http://manpages.ubuntu.com/manpages/precise/man1/ragg2-cc.1.html

EXAMPLE ragg2-cc hi.c
EXAMPLE ragg2-cc -x hi.c
EXAMPLE ragg2 -e xor -c key=32 -B ‘ragg2-cc -x hi.c’

rahash2

DESCRIPTION **rahash2** - **radare** tool for creating hashes. **rahash2** is designed to work with blocks like **radare** does. So this way you can generate multiple checksums from a single file, and then make a faster comparison of the blocks to find the part of the file that has changed.

This is useful in forensic tasks, when progressively analysing memory dumps to find the places where it has changed and then use **'radiff**' to get a closer look to these changes.

This is the default work way for **rahash2**. So lets generate a **rahash2** checksumming file and then use it to check if something has changed. The default block size is 32 KBytes. You can change it by using the -b flag.

USAGE rahash2 [-action] [-options] [source] [hash-file]

OPTIONS rahash2 –h

OPTIONS check rahash http://radare.org/doc/html/Section18.1.html

EXAMPLE rahash2 -a md5 -s 'hello world'

rarun2

DESCRIPTION **rarun2** — radare2 utility to run programs in exotic environments. This program is used as a launcher for running programs with different environment, arguments, permissions, directories and overridden default file descriptors. The program just accepts a single argument which is the filename of the configuration file to run the program. It is useful when you have to run a program using long arguments or pass long data to stdin or things like that usually required for exploiting crackmes.

USAGE rarun2 [[script.rr2]]

OPTIONS http://manned.org/rarun2.1

EXAMPLE

```
$ cat foo.rr2
#!/usr/bin/rarun2
program=./pp400
arg0=10
stdin=foo.txt
chdir=/tmp
#chroot=.
./foo.rr2
```

rax2

DESCRIPTION **rax2** — radare base converter. This command is part of the **radare** project. This command allows you to convert values between positive and negative integer, float octal, binary and hexadecimal values.

USAGE rax2 [-ebsSvxkh] [[value] ...]

OPTIONS http://manned.org/rax2.1

EXAMPLE rax2 -s 41 42 43
EXAMPLE rax2 33 0x41 0101b

[56] STRESS TESTING: NETWORK STRESS TESTING

- denial6
- dhcpig
- dos-new-ip6
- flodd_advertise6
- flood_dhcpc6
- flood_mld26
- flood_mld6
- flood_mldrouter26
- flood_router6
- flood_solicitate6
- fragmentation6
- inundator
- kill_router6
- macof
- rsmurf6
- siege
- smurf6
- t50

denial6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

denial6 - tests various known IPv6 vulnerabilities against a target. Performs various denial of service attacks on a target. If a system is vulnerable, it can crash or be under heavy load, so be careful!

USAGE n/a

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

dhcpig

DESCRIPTION text

USAGE text

OPTIONS text

EXAMPLE text

dos-new-ip6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

Dos-new-ip6 - this tool prevents new ipv6 interfaces to come up, by sending answers to duplicate ip6 checks (DAD). This results in a DOS for new ipv6 devices.

USAGE dos-new-ip6 interface

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flodd_advertise6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flodd_advertise6- floods the local network with neighbour advertisements

USAGE flodd_advertise6 interface

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flood_dhcpc6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flood_dhcpc6 - DHCP client flooder. Use to deplete the IP address pool a DHCP6 server is offering. Note: if the pool is very large, this is rather senseless. :-) By default the link-local IP MAC address is random, however this won't work in some circumstances. -n will use the real MAC, -N the real MAC and link-local address. -1 will only solicate an address but not request it. If -N is not used, you should run parasite6 in parallel. Use -d to force DNS updates, you can specify a domain name on the commandline.

USAGE flood_dhcpc6 [-n|-N] [-1] [-d] interface [domain-name]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flood_mld26

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flood_mld26- flood the local network with MLDv2 reports.

USAGE flood_mld26 interface

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flood_mld6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flood_mld6- flood the local network with router advertisements.

USAGE flood_mld6 [-HFD] interface

OPTIONS -F/-D/-H add fragment/destination/hopbyhop header to bypass RA guard security.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flood_mldrouter26

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flood_mldrouter26 - flood the local network with MLD router advertisements.

USAGE flood_mldrouter26 interface

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flood_router6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flood_router6- flood the local network with router advertisements.

USAGE flood_router6 [-HFD] interface

OPTIONS -F/-D/-H add fragment/destination/hopbyhop header to bypass RA guard security.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

flood_solicitate6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

flood_solicitate6- flood the network with neighbor solicitations.

USAGE flood_solicitate6 interface [target]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

fragmentation6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

fragmentation6- this tool prevents new ipv6 interfaces to come up, by sending answers to duplicate ip6 checks (DAD). This results in a DOS for new ipv6 devices.

USAGE fragmentation6[-fp] [-n number] interface destination [test-case-no]

OPTIONS -f activates flooding mode, no pauses between sends; -p disables first and final pings, -n number specifies how often each test is performed. Performs fragment firewall and implementation checks, incl. denial-of-service.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

inundator

DESCRIPTION **Inundator** is a multi-threaded, queue-driven, anonymous intrusion detection false positives generator with support for multiple targets.

Usage

- *Before, during, and after a real attack to bury any potential alerts among a flood of false positives.*
- *Seriously mess with an IDS analyst and keep an InfoSec department busy for days investigating false positives.*
- *Test the effectiveness of an intrusion detection or prevention system. Less alerts means a better product; more alerts means a horrible product.*

USAGE ./inundator.pl --verbose --threads 10 <IP>

EXAMPLE inundator 68.177.102.20

EXAMPLE inundator -r /etc/snort/rules -p localhost:9050 victim_ip

where -r is the path to the snort rules location
where -p is the SOCKS proxy configuration
and the last argument is the victim ip

kill_router6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

kill_router6- announce that a target a router going down to delete it from the routing tables. If you supply a '*' as router-address, this tool will sniff the network for RAs and immediately send the kill packet.

USAGE kill_router6 [-HFD] interface router-address [srcmac [dstmac]]

OPTIONS Option -H adds hop-by-hop, -F fragmentation header and -D dst header.

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

macof

DESCRIPTION **macof** floods the local network with random MAC addresses (causing some switches to fail open in repeating mode, facilitating sniffing). *This program could cause problems on your network. This program could hang, crash or reboot network devices. Switches could start sending packages to all ports making it possible to intercept network traffic.*

USAGE **macof** [**-i** *interface*] [**-s** *src*] [**-d** *dst*] [**-e** *tha*] [**-x** *sport*] [**-y** *dport*] [**-n** *times*]

OPTIONS

-i interface	Specify the interface to send on.
-s src	Specify source IP address.
-d dst	Specify destination IP address.
-e tha	Specify target hardware address.
-x sport	Specify TCP source port.
-y dport	Specify TCP destination port.
-n times	Specify the number of packets to send.

Values for any options left unspecified will be generated randomly.

EXAMPLE ./macof -e <mac_of_def_gate> -n 1000000

EXAMPLE ./macof -r -n 1000000

rsmurf6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

rsmurf6- smurfs the local network of the victim. Note: this depends on an implementation error, currently only verified on Linux. Evil: "ff02::1" as victim will DOS your local LAN completely.

USAGE rsmurf6 interface victim-ip

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

siege

DESCRIPTION **siege** - An HTTP/HTTPS stress tester. **Siege** is a multi-threaded http load testing and benchmarking utility. It was designed to let web developers measure the performance of their code under duress. It allows one to hit a web server with a configurable number of concurrent simulated users. Those users place the webserver "under siege." Performance measures include elapsed time, total data transferred, server response time, its transaction rate, its throughput, its concurrency and the number of times it returned OK. These measures are quantified and reported at the end of each run. Their meaning and significance is discussed below. Siege has essentially three modes of operation: regression (when invoked by bombardment), internet simulation and brute force.

USAGE siege [options] siege [options] [url]

OPTIONS http://linux.die.net/man/1/siege

EXAMPLE siege -c25 -t1M www.example.com

EXAMPLE siege –g www.google.com

smurf6

DESCRIPTION **thc-ipv6 - THC-IPV6-ATTACK-TOOLKIT** - just run the tools without options and they will give you help and show the command line options.

smurf6- smurf the target with icmp echo replies. Target of echo request is the local all-nodes multicast address if not specified.

USAGE smurf6 interface victim-ip [multicast-network-address]

EXAMPLE n/a

TIP DETECTION

Most tools can easily be detected by an IDS or specialized detection software. This is done on purpose to make rogue usage detection easier. The tools either specify a fixed packet signature, or generically sniff for packets (e.g. therefore also answering to icmp6 neighbour solicitations which are sent to a non-existing mac, and are therefore very easy to detect). If you don't want this, change the code.

t50

DESCRIPTION T50 is multi-protocol packet injector too.

Features
- Flooding
- CIDR support
- TCP, UDP, ICMP, IGMPv2, IGMPv3, EGP, DCCP, RSVP, RIPv1, RIPv2, GRE, ESP, AH, EIGRP and OSPF support.
- TCP Options.
- High performance.
- Can hit about 1.000.000 packets per second.

USAGE t50 <host> [/CIDR] [options]

OPTIONS https://github.com/merces/t50/blob/master/src/usage.c
OPTIONS t50 -h

EXAMPLE t50 VICTIM_IP --flood -S –turbo
EXAMPLE t50 VICTIM_IP --flood --turbo --dport (80 443) -S --protocol TCP

[57] VOIP STRESS TESTING

- iaxflood
- invite flood

iaxflood

DESCRIPTION IAXFlood is a tool for flooding the IAX2 protocol which is used by the Asterisk PBX.

USAGE ./iaxflood sourcename destinationname numpackets

EXAMPLE n/a

invite flood

DESCRIPTION **SIP INVITE Flood -** The two attacks above both target HTTP; this one is a VoIP flood that targets SIP. It takes advantage of the normal time lag during the SIP call initiation process to overload a SIP server. Since SIP runs over UDP, a single packet from a caller, hacker, or botnet can start the process of “dialing” and ringing at the beginning of a phone call. In our everyday lives, we don’t think anything of the 20-second delay between entering a phone number and hearing “Hello” or the voicemail prompt from the other end. But that delay, when multiplied across thousands of simultaneous connections, can crash a server and potentially open the door for even more mayhem within a VoIP-based call center. Plus, it’s very difficult to determine in advance which calls are legitimate and which ones are part of a DDoS, especially if an attacker is clever enough to spoof the IP addresses in UDP headers, or to spoof SIP headers so they don’t match the corresponding UDP headers.

USAGE n/a; GUI tool

OPTIONS n/a; GUI tool

EXAMPLE n/a; GUI tool

[58] WEB STRESS TESTING

- thc-ssl-dos

thc-ssl-dos

DESCRIPTION **thc-ssl-dos** - THC has released a DOS tool that exploits SSL renegotiation to perform a denial of service on a given SSL server. It uses renegotiation to constantly trigger new SSL handshakes with the server, using one single TCP connection. **THC-SSL DOS** was developed by a hacking group called The Hacker's Choice (THC), as a proof-of-concept to encourage vendors to patch a serious SSL vulnerability. **THC-SSL-DOS**, as with other "low and slow" attacks, requires only a small number of packets to cause denial-of-service for a fairly large server. It works by initiating a regular SSL handshake and then immediately requesting for the renegotiation of the encryption key, constantly repeating this server resource-intensive renegotiation request until all server resources have been exhausted.

USAGE www.thc.org/thc-ssl-dos/

EXAMPLE ./thc-ssl-dos 127.3.133.7 443

TIP

The average server can do 300 handshakes per second. This would require 10-25% of your laptops CPU

Use multiple hosts (SSL-DOS) if an SSL Accelerator is used

Be smart in target acquisition. The HTTPS Port (443) is not always the best choice. Other SSL enabled ports are more likely to use an SSL Accelerator (like POP3S, SMTPS, .. or the secure database port)

[59] WLAN STRESS TESTING

- mdk3
- reaver

mdk3

DESCRIPTION **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.*

Features:

- *Bruteforce MAC Filters*
- *Bruteforce hidden SSIDs (some small SSID wordlists included)*
- *Probe networks to check if they can hear you*
- *intelligent Authentication-DoS to freeze APs (with success checks)*
- *FakeAP - Beacon Flooding with channel hopping (can crash NetStumbler and some buggy drivers)*
- *Disconnect everything (aka AMOK-MODE) with Deauthentication and Disassociation packets*
- *WPA TKIP Denial-of-Service*
- *WDS Confusion - Shuts down large scale multi-AP installations*

USAGE mdk3 <interface> <test_mode> [test_options]

OPTIONS http://hack-it.org/index.php?title=Mdk3

EXAMPLE mdk3 –fullhelp (for all test options)

reaver

DESCRIPTION **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. Specifically, **Reaver** targets the registrar functionality of WPS, which is flawed in that it only takes 11,000 attempts to guess the correct WPS pin in order to become a WPS registrar. Once registered as a registrar with the access point, the access point will give you the WPA passphrase.

USAGE reaver -i <interface> -b <target bssid> -vv

OPTIONS

-m, --mac=<mac> MAC of the host system (should be resolved automatically)
-e, --essid=<ssid> ESSID of the target AP. Unless cloaked, this will be resolved automatically.
-c, --channel=<channel> Set the 802.11 channel for the interface (implies -f)
-o, --out-file=<file> Send output to a log file [default: stdout]
-f, --fixed Disable channel hopping
-5, --5ghz Use 5GHz 802.11 channels
-v, --verbose Display non-critical warnings (-vv for more)
-q, --quiet Only display critical messages
-i, --interface=<wlan> Name of the monitor-mode interface to use
-b, --bssid=<mac> BSSID of the target AP
-p, --pin=<wps pin> Use the specified WPS pin
-h, --help Show help

EXAMPLE reaver -i mon0 -b 00:01:02:03:04:05

TIP https://code.google.com/p/reaver-wps/wiki/HintsAndTips

[60] HARDWARE HACKING: ANDROID TOOLS

- android-sdk
- apktool
- baksmali
- dex2jar
- smali

android-sdk

DESCRIPTION **The Android SDK** provides you the API libraries and developer tools necessary to build, test, and debug apps for Android.
Read more: http://developer.android.com/sdk/index.html

USAGE n/a; GUI
OPTIONS n/a; GUI
EXAMPLE n/a; GUI

apktool

DESCRIPTION **APKTool** is an application which decompiles and recompiles android APKs. It 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; it makes possible to debug smali code step by step. Also it makes working with app easier because of project-like files structure and automation of some repetitive tasks like building apk, etc.

USAGE [q|--quiet OR –v|--verbose] COMMAND [...]

OPTIONS https://code.google.com/p/android-apktool/wiki/ApktoolOptions

EXAMPLE apktool if SystemUI.apk
EXAMPLE apktool d SystemUI.apk
EXAMPLE apktool b SystemUI almostdone.apk

baksmali

DESCRIPTION **Smali /Baksmali** is an assembler / disassembler for dex file format. When you baksmali (disassemble) it, the tool will disassemble and extract all the classes present in the dex file. Practically you will not get files under .class extension. Rather you will get .smali files, in which you will get code in form of dalvik opcodes or smali syntax. More info: https://code.google.com/p/smali/

USAGE baksmali -a <api_level> -x <odex_file> -d <framework_dir>

EXAMPLE baksmali -x -a 14 -c <copied bootclasspath> ./system/framework/services.odex

-x = odex
-a = api level 14
-c = classes (loaded from the bootclasspath, separated by colon)
If you've done this correctly you will now see a directory called 'out', otherwise verify you've pulled the jars and bootclasspath correctly.

EXAMPLE baksmali -a 15 -x Calculator.odex -d framework -o Calculator

EXAMPLE baksmali -JXmx512m -x blah.odex

EXAMPLE ava -Xmx512m -jar baksmali.jar -x blah.odex

dex2jar

DESCRIPTION **dex2jar** a dex decompiler.

dex2jar contains 4 compments:

- **dex-reader** is designed to read the Dalvik Executable (.dex/.odex) format. It has a light weight API similar with ASM. An example here
- **dex-translator** is designed to do the convert job. It reads the dex instruction to **dex-ir** format, after some optimize, convert to ASM format.
- **dex-ir** used by dex-translator, is designed to represent the dex instruction
- **dex-tools** tools to work with .class files. here are examples

USAGE n/a

OPTIONS n/a

EXAMPLE n/a

smali

DESCRIPTION **Smali /Baksmali** is an assembler / disassembler for dex file format. When you baksmali (disassemble) it, the tool will disassemble and extract all the classes present in the dex file. Practically you will not get files under .class extension. Rather you will get .smali files, in which you will get code in form of dalvik opcodes or smali syntax. More info: https://code.google.com/p/smali/

USAGE baksmali -a <api_level> -x <odex_file> -d <framework_dir>

EXAMPLE java -jar smali.jar -o classes.dex HelloWorld.smali

EXAMPLE java -Xmx512m -jar smali.jar HelloWorld.smali

smali ./out -o classes.dex

[61] ARDUINO TOOLS

- arduino

arduino

DESCRIPTION **Arduino** is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists, and anyone interested in creating interactive objects or environments. The open-source **Arduino** environment makes it easy to write code and upload it to the i/o board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing, avr-gcc, and other open source software.

The **Arduino** development environment contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions, and a series of menus. It connects to the **Arduino** hardware to upload programs and communicate with them.

Software written using **Arduino** are called **sketches**. These sketches are written in the text editor. Sketches are saved with the file extension .ino. It has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the **Arduino** environment including complete error messages and other information. The bottom righthand corner of the window displays the current board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

More info: http://arduino.cc

[62] FORENSICS: ANTI-VIRUS FORENSICS TOOLS

- chrootkit

chrootkit

DESCRIPTION **chkrootkit** - determine whether the system is infected with a rootkit. **chkrootkit** examines certain elements of the target system and determines whether they have been tampered with. Some tools which **chkrootkit** applies while analysing binaries and log files can be found at /usr/lib/chkrootkit.
More info: http://www.spenneberg.org/chkrootkit-mirror/faq/

USAGE **chkrootkit** [OPTION]... [TESTNAME]...

OPTIONS http://manpages.ubuntu.com/manpages/hardy/man1/chkrootkit.1.html

EXAMPLE ./chkrootkit -x | more (see lots of data)
EXAMPLE ./chkrootkit -x | egrep '^/' (Pathnames inside system commands)

[63] DIGITAL ANTI-FORENSICS

- chrootkit

chrootkit

DESCRIPTION **chkrootkit** - determine whether the system is infected with a rootkit. **chkrootkit** examines certain elements of the target system and determines whether they have been tampered with. Some tools which **chkrootkit** applies while analysing binaries and log files can be found at /usr/lib/chkrootkit.
More info: http://www.spenneberg.org/chkrootkit-mirror/faq/

USAGE **chkrootkit** [OPTION]... [TESTNAME]...

OPTIONS http://manpages.ubuntu.com/manpages/hardy/man1/chkrootkit.1.html

EXAMPLE ./chkrootkit -x | more (see lots of data)
EXAMPLE ./chkrootkit -x | egrep '^/' (Pathnames inside system commands)

[64] DIGITAL FORENSICS

- autopsy
- binwalk
- bulk_extractor
- chrootkit
- dc3dd
- dcfldd
- extundelete
- foremost
- fsstat
- galleta
- tsk_comparedir
- tsk_loaddb

autopsy

DESCRIPTION **Autopsy** is a graphical interface to the command line digital investigation analysis tools in **The Sleuth Kit**. Together, they can analyse Windows and UNIX disks and file systems (NTFS, FAT, UFS1/2, Ext2/3).

As **Autopsy** is HTML-based, you can connect to the **Autopsy** server from any platform using an HTML browser. **Autopsy** provides a "File Manager"-like interface and shows details about deleted data and file system structures.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

binwalk

DESCRIPTION **Binwalk** is a tool for searching a given binary image for embedded files and executable code. Specifically, it is designed for identifying files and code embedded inside of firmware images. **Binwalk** uses the **libmagic** library, so it is compatible with magic signatures created for the Unix file utility.

USAGE **binwalk** [OPTIONS] [FILE1] [FILE2] [FILE3] ...

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/binwalk.1.html

EXAMPLE text

bulk_extractor

DESCRIPTION **bulk_extractor** is a C++ program that scans a disk image, a file, or a directory of files and extracts useful information without parsing the file system or file system structures. The results are stored in feature files that can be easily inspected, parsed, or processed with automated tools. **bulk_extractor** also creates histograms of features that it finds, as features that are more common tend to be more important. We have made the following tools available for processing feature files generated by **bulk_extractor**: We have provided a small number of python programs that perform automated processing on feature files.

More info: http://digitalcorpora.org/downloads/bulk_extractor/doc/2012-08-08-bulk_extractor-tutorial.pdf

TIP see BEViewer – GUI for bulk_extractor: https://github.com/simsong/bulk_extractor/wiki/BEViewer

USAGE bulk_extractor [options] imagefile

OPTIONS bulk_extractor -h

EXAMPLE bulk_extractor -p 340731773 /corp/nps/drives/nps-2009-ubnist1/ubnist1.gen3.E01

EXAMPLE bulk_extractor -p 340731773-GZIP-9200 /corp/nps/drives/nps-2009-ubnist1/ubnist1.gen3.E01

EXAMPLE bulk_extractor -o charlie-2009-12-11 drives-redacted/charlie-2009-12-11.E01

chrootkit

DESCRIPTION **chkrootkit** - determine whether the system is infected with a rootkit. **chkrootkit** examines certain elements of the target system and determines whether they have been tampered with. Some tools which **chkrootkit** applies while analysing binaries and log files can be found at /usr/lib/chkrootkit.
More info: http://www.spenneberg.org/chkrootkit-mirror/faq/

USAGE **chkrootkit** [OPTION]... [TESTNAME]...

OPTIONS http://manpages.ubuntu.com/manpages/hardy/man1/chkrootkit.1.html

EXAMPLE ./chkrootkit -x | more (see lots of data)
EXAMPLE ./chkrootkit -x | egrep '^/' (Pathnames inside system commands)

dc3dd

DESCRIPTION **dc3dd (Department of Defense Cyber Crime Center)** is a patched version of **GNU dd** with added features for computer forensics. It is a powerful imaging tool that will create a file that contains an exact replica of a hard drive.

More info: http://www.myfixlog.com/fix.php?fid=33

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

dcfldd

DESCRIPTION **dcfldd** is an enhanced version of **GNU dd** with features useful for forensics and security.

Based on the **dd** program found in the GNU Coreutils package, **dcfldd** has the following additional features:

- Hashing on-the-fly - dcfldd can hash the input data as it is being transferred, helping to ensure data integrity.
- Status output - dcfldd can update the user of its progress in terms of the amount of data transferred and how much longer operation will take.
- Flexible disk wipes - dcfldd can be used to wipe disks quickly and with a known pattern if desired.
- Image/wipe Verify - dcfldd can verify that a target drive is a bit-for-bit match of the specified input file or pattern.
- Multiple outputs - dcfldd can output to multiple files or disks at the same time.
- Split output - dcfldd can split output to multiple files with more configurability than the split command.
- Piped output and logs - dcfldd can send all its log data and output to commands as well as files natively.

USAGE **dcfldd** [*OPTION*]...

OPTIONS http://linux.die.net/man/1/dcfldd

EXAMPLE dcfldd if=/dev/hda1 of=/mnt/data/image.dd hashlog=/mnt/data/md5hash2.txt

extundelete

DESCRIPTION **extundelete** is a utility that can recover deleted files from an ext3 or ext4 partition extundelete uses the information stored in the partition's journal to attempt to recover a file that has been deleted from the partition. There is no guarantee that any particular file will be able to be undeleted, so always try to have a good backup system in place, or at least put one in place after recovering your files!

More info: http://extundelete.sourceforge.net/

USAGE **extundelete** [options] device-file...

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/extundelete.1.html

EXAMPLE extundelete /dev/sda4 --restore-all

foremost

DESCRIPTION Recover files from a disk image based on file types specified by the user using the -t switch. Supports: jpg, gif, png, bmp, avi, exe, mpg, wav, riff, wmv, mov, pdf, ole, doc, zip, rar, htm, cpp)

USAGE **foremost [-h][-V][-d][-vqwQT][-b**<blocksize>**][-o**<dir>**][-t**<type>**][-s**<num>**][-i**<file>**]**

OPTIONS http://manpages.ubuntu.com/manpages/hardy/en/man1/foremost.1.html

EXAMPLE foremost -s 100 -t jpg -i image.dd (Search for jpeg format skipping the first 100 blocks)
EXAMPLE foremost -av image.dd (Only generate an audit file, and print to the screen (verbose mode))
EXAMPLE foremost -t all -i image.dd (Search all defined types)
EXAMPLE foremost -t gif,pdf -i image.dd (Search for gifs and pdfs)
EXAMPLE foremost -vd -t ole,jpeg -i image.dd (Search for office documents and jpeg files in a Unix file system in verbose mode.)
EXAMPLE foremost image.dd (**Run the default case**)

fsstat

DESCRIPTION **fsstat** displays the details associated with a file system. The output of this command is file system specific. At a minimum, the range of meta-data values (inode numbers) and content units (blocks or clusters) are given. Also given are details from the Super Block, such as mount times and features. For file systems that use groups (FFS and EXT2FS), the layout of each group is listed. For a FAT file system, the FAT table is displayed in a condensed format. Note that the data is in sectors and not in clusters.

USAGE **fsstat [-f** *fstype* **] [-i imgtype] [-o imgoffset] [-b dev_sector_size] [-tvV]** *image* [images]

OPTIONS

-t type Print the file system type only.

-f fstype Specify the file system type. Use '-f list' to list the supported file system types. If not given, autodetection methods are used.

-i imgtype Identify the type of image file, such as raw. Use '-i list' to list the supported types. If not given, autodetection methods are used.

-o imgoffset The sector offset where the file system starts in the image.

-b dev_sector_size The size, in bytes, of the underlying device sectors. If not given, the value in the image format is used (if it exists) or 512-bytes is assumed.-vVerbose output of debugging statements to stderr

-V Display version

image [images] The disk or partition image to read, whose format is given with '-i'. Multiple image file names can be given if the image is split into multiple segments. If only one image file is given, and its name is the first in a sequence (e.g., as indicated by ending in '.001'), subsequent image segments will be included automatically.

EXAMPLE fsstat usb.img

galleta

DESCRIPTION **galleta** is a tool to extract valuable information (from a forensics investigator point of view) from MS IE cookie files. It will extract the website name, the variables names and values. The creation and expire time for these variables and also flags.

USAGE galleta [-t] FILE

OPTIONS

-t FD Change the default field delimiter (TAB) to FD.

<file> Cookie file to parse.

EXAMPLE ./galleta antihackertoolkit.txt > cookies.txt

tsk_comparedir

DESCRIPTION **tsk_comparedir** compares the contents of *image* to the contents of *comparison_directory*. This can be useful for detecting rootkits and when testing. Rootkits can be detected by comparing the contents of a local directory and a local raw device. The rootkits typically don't hide data when it is read directly from the raw device.

USAGE **tsk_comparedir [-vV] [-n** *start_inum* **] [-f** *fstype* **] [-i** *imgtype* **] [-b** *dev_sector_size* **] [-o** *sector_offset* **]** *image* [images] comparison_directory

OPTIONS http://www.sleuthkit.org/sleuthkit/man/tsk_comparedir.html

EXAMPLE tsk_comparedir ./image.dd ./directory

tsk_loaddb

DESCRIPTION **tsk_loaddb** loads disk information from *image* to a SQLite database. This database can then be used by tools in other languages for analysis. By default, the database is stored in the same directory as the image with ".db" appended to the name or the database name can be specified with '-d'.

USAGE **tsk_loaddb [-ahkvV] [-i** *imgtype* **] [-b** *dev_sector_size* **] [-i** *imgtype* **] [-d** *database* **]** *image* [images]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/tsk_loaddb.html

EXAMPLE tsk_loaddb ./image.dd

[65] FORENSIC ANALYSIS TOOLS

- affcompare
- affcopy
- affcrypto
- affdiskprint
- affinfo
- affsign
- affstats
- affuse
- affverify
- affxml
- autopsy
- binwalk
- blkcalc
- blkcat
- blkstat
- bulk_extractor
- ffind
- fls
- foremost
- galleta
- hfind
- icat-sleuthkit
- ifind
- ifind
- ils-sleuthkit
- istat
- jcat
- mactime-sleuthkit
- missidentify
- mmcat
- pdgmail
- readpst
- reglookup
- sorter
- srch-strings
- tsk_recover
- vinetto

affcompare

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- ***Compare disk images and report the data or metadata that is different.***
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affcopy

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- ***Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.***
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affcrypto

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affdiskprint

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- ***Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.***

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affinfo

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- ***Print information about a file.***
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affsign

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affstats

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- ***Print detailed statistics about a file***
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affuse

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affverify

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another,* ***with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.***
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- *Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)*
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

affxml

DESCRIPTION **AFFLIBv3 - The Advanced Forensic Format Library and Tools Version 3. AFF Library and Toolkit** is a set of programs for working with computer forensic information.
More info: https://github.com/simsong/AFFLIBv3

Using these tools you can:

- *Interconvert disk images between a variety of formats*
- *Compare disk images and report the data or metadata that is different.*
- *Copy disk images from one location to another, with full verification of data, metadata, and the automatic generation of a chain-of-custody segment.*
- *Find errors in an AFF file and fix them.*
- *Print information about a file.*
- *Print detailed statistics about a file*
- ***Generate an XML representation of a disk image's metadata (for example, acquisition time or the serial number of the acquisition device.)***
- *Produce an XML "diskprint" which allows a disk image to be rapidly fingerprinted without having the computer the SHA1 of the entire disk.*

USAGE n/a
OPTIONS n/a
EXAMPLE n/a

autopsy

DESCRIPTION **Autopsy** is a graphical interface to the command line digital investigation analysis tools in **The Sleuth Kit**. Together, they can analyze Windows and UNIX disks and file systems (NTFS, FAT, UFS1/2, Ext2/3).

As **Autopsy** is HTML-based, you can connect to the **Autopsy** server from any platform using an HTML browser. **Autopsy** provides a "File Manager"-like interface and shows details about deleted data and file system structures.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

binwalk

DESCRIPTION **Binwalk** is a tool for searching a given binary image for embedded files and executable code. Specifically, it is designed for identifying files and code embedded inside of firmware images. **Binwalk** uses the **libmagic** library, so it is compatible with magic signatures created for the Unix file utility.

USAGE **binwalk** [OPTIONS] [FILE1] [FILE2] [FILE3] ...

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/binwalk.1.html

EXAMPLE binwalk firmware.bin (Basic binwalk usage is very simple; just supply it with the path to a target file)
EXAMPLE binwalk -y filesystem firmware.bin (Include Filters)
EXAMPLE binwalk -x jffs2 firmware.bin (Exclude Filters)
EXAMPLE binwalk -y filesystem -x jffs2 firmware.bin (Advanced Filters)
EXAMPLE binwalk -e firmware.bin (Automated Extraction)
EXAMPLE binwalk -f binwalk.log firmware.bin (Logging)
EXAMPLE binwalk --list-plugins (Listing Plugins)

blkcalc

DESCRIPTION **blkcalc** - Converts between unallocated disk unit numbers and regular disk unit numbers. **blkcalc** creates a disk unit number mapping between two images, one normal and another that only contains the unallocated units of the first (the default behaviour of the **blkls** program). One of the -d, -s, or -u options must be given. If the -d option is given, then the **unit_addr** value is the disk unit address in the regular image (i.e. from dd). If the unit is unallocated, its address in an unallocated image is given. If the -u option is given, then the unit_addr value is the disk unit address in the unallocated unit image (i.e. from **blkls**). Its disk unit address in the original image is determined. If the -s option is given, then the unit_addr value is the disk unit address in the slack image (i.e. from **blkls** -s). The image is the full, original image (i.e. from dd). **blkcalc** was called **dcalc** in TSK versions prior to 3.0.0.

USAGE **blkcalc [-dsu unit_addr] [-vV] [-i imgtype] [-o imgoffset] [-b dev_sector_size] [-f fstype] image [images]**

OPTIONS http://www.sleuthkit.org/sleuthkit/man/blkcalc.html

EXAMPLE blkcalc -u 64 images/wd0e

blkcat

DESCRIPTION **blkcat** displays **num** data units (default is one) starting at the unit address **unit_addr** from **image** to stdout in different formats (default is raw). **blkcat** was called **dcat** in TSK versions prior to 3.0.0.

USAGE **blkcat [-ahswvV] [-f fstype] [-u unit_size] [-i imgtype] [-o imgoffset] [-b dev_sector_size]** *image* [images] unit_addr [num]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/blkcat.html

EXAMPLE blkcat -hw image 264
EXAMPLE blkcat -hw image 264 4

blkstat

DESCRIPTION **blkstat** - displays details of a file system data unit (i.e. block or sector) . **blkstat** was called **dstat** in TSK versions prior to 3.0.0.

USAGE **blkstat [-f** *fstype* **] [-i imgtype] [-o imgoffset] [-b dev_sector_size] [-vV]** *image* [images] addr

OPTIONS http://www.sleuthkit.org/sleuthkit/man/blkstat.html

EXAMPLE blkstat imagefile.dd cluster_number
EXAMPLE blkstat $image 28754447

bulk_extractor

DESCRIPTION **bulk_extractor** is a C++ program that scans a disk image, a file, or a directory of files and extracts useful information without parsing the file system or file system structures. The results are stored in feature files that can be easily inspected, parsed, or processed with automated tools. **bulk_extractor** also creates histograms of features that it finds, as features that are more common tend to be more important. We have made the following tools available for processing feature files generated by **bulk_extractor**: We have provided a small number of python programs that perform automated processing on feature files.

More info: http://digitalcorpora.org/downloads/bulk_extractor/doc/2012-08-08-bulk_extractor-tutorial.pdf

TIP see BEViewer – GUI for bulk_extractor: https://github.com/simsong/bulk_extractor/wiki/BEViewer

USAGE bulk_extractor [options] imagefile

OPTIONS bulk_extractor -h

EXAMPLE bulk_extractor -p 340731773 /corp/nps/drives/nps-2009-ubnist1/ubnist1.gen3.E01

EXAMPLE bulk_extractor -p 340731773-GZIP-9200 /corp/nps/drives/nps-2009-ubnist1/ubnist1.gen3.E01

EXAMPLE bulk_extractor -o charlie-2009-12-11 drives-redacted/charlie-2009-12-11.E01

ffind

DESCRIPTION **ffind** finds the names of files or directories that are allocated to *inode* on disk image *image.* By default it only will only return the first name it finds. With some file systems, this will find deleted file names.

USAGE **ffind [-aduvV] [-f fstype] [-i imgtype] [-o imgoffset] [-b dev_sector_size]** *image* [images] inode

OPTIONS http://www.sleuthkit.org/sleuthkit/man/ffind.html

EXAMPLE ffind -a image 212

fls

DESCRIPTION **fls** lists the files and directory names in the *image* and can display file names of recently deleted files for the directory using the given *inode.* If the inode argument is not given, the inode value for the root directory is used. For example, on an NTFS file system it would be 5 and on a Ext3 file system it would be 2.

USAGE **fls [-adDFlpruvV] [-m** *mnt* **] [-z** *zone* **] [-f** *fstype* **] [-s** *seconds* **] [-i** *imgtype* **] [-o** *imgoffset* **] [-b dev_sector_size]** *image* [images] **[** *inode* **]**

OPTIONS http://www.sleuthkit.org/sleuthkit/man/fls.html

EXAMPLE To get a list of all files and directories in an image use: # fls -r image 2
or just (if no inode is specified, the root directory inode is used): # fls -r image

EXAMPLE To get the full path of deleted files in a given directory: # fls -d -p image 29

EXAMPLE To get the mactime output do: # fls -m /usr/local image 2

EXAMPLE If you have a disk image and the file system starts in sector 63, use: # fls -o 63 disk-img.dd

EXAMPLE If you have a disk image that is split use: # fls -i "split" -o 63 disk-1.dd disk-2.dd disk-3.dd

foremost

DESCRIPTION Recover files from a disk image based on file types specified by the user using the -t switch. Supports: jpg, gif, png, bmp, avi, exe, mpg, wav, riff, wmv, mov, pdf, ole, doc, zip, rar, htm, cpp)

USAGE **foremost [-h][-V][-d][-vqwQT][-b**<blocksize>**][-o**<dir>**][-t**<type>**][-s**<num>**][-i**<file>**]**

OPTIONS http://manpages.ubuntu.com/manpages/hardy/en/man1/foremost.1.html

EXAMPLE foremost -s 100 -t jpg -i image.dd (Search for jpeg format skipping the first 100 blocks)
EXAMPLE foremost -av image.dd (Only generate an audit file, and print to the screen (verbose mode))
EXAMPLE foremost -t all -i image.dd (Search all defined types)
EXAMPLE foremost -t gif,pdf -i image.dd (Search for gifs and pdfs)
EXAMPLE foremost -vd -t ole,jpeg -i image.dd (Search for office documents and jpeg files in a Unix file system in verbose mode.)
EXAMPLE foremost image.dd (**Run the default case**)

galleta

DESCRIPTION **galleta** is a tool to extract valuable information (from a forensics investigator point of view) from MS IE cookie files. It will extract the website name, the variables names and values. The creation and expire time for these variables and also flags.

USAGE galleta [-t] FILE

OPTIONS

-t FD Change the default field delimiter (TAB) to FD.

<file> Cookie file to parse.

EXAMPLE ./galleta antihackertoolkit.txt > cookies.txt

hfind

DESCRIPTION **hfind** looks up hash values in a database using a binary search algorithm. This allows one to easily create a hash database and identify if a file is known or not. It works with the NIST National Software Reference Library (NSRL) and the output of **'md5sum**'.

Before the database can be used by **'hfind**', an index file must be created with the '-i' option.

This tool is needed for efficiency. Most text-based databases do not have fixed length entries and are sometimes not sorted. The **hfind** tool will create an index file that is sorted and has fixed-length entries. This allows for fast lookups using a binary search algorithm instead of a linear search such as **'grep**'.

USAGE **hfind [-i** *db_type* **] [-f** *lookup_file* **] [-eq]** *db_file* [hashes]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/hfind.html

EXAMPLE To create an MD5 index file for NIST NSRL: # hfind -i nsrl-md5 /usr/local/hash/nsrl/NSRLFile.txt

EXAMPLE To lookup a value in the NSRL: # hfind /usr/local/hash/nsrl/NSRLFile.txt 76b1f4de1522c20b67acc132937cf82e

EXAMPLE You can even do both SHA-1 and MD5 if you want: # hfind -i nsrl-sha1 /usr/local/hash/nsrl/NSRLFile.txt

EXAMPLE To look entries up, the following will work: # hfind system.md5 76b1f4de1522c20b67acc132937cf82e

icat-sleuthkit

DESCRIPTION **icat** opens the named *image(s)* and copies the file with the specified *inode* number to standard output.

USAGE **icat [-hrsvV] [-f** *fstype* **] [-i** *imgtype* **] [-o** *imgoffset* **] [-b dev_sector_size]** *image* [images] inode

OPTIONS http://www.sleuthkit.org/sleuthkit/man/icat.html

EXAMPLE The following command would display the default data attribute (128-1): # icat -f ntfs ntfs.dd 49

or: # icat -f ntfs ntfs.dd 49-128-1

EXAMPLE The following displays the other data stream: # icat -f ntfs ntfs.dd 49-128-5

EXAMPLE The raw format of the $FILE_NAME attribute can be viewed using: # icat -f ntfs ntfs.dd 49-48-2

ifind

DESCRIPTION **ifind** finds the meta-data structure that has *data_unit* allocated a data unit or has a given file name. In some cases any of the structures can be unallocated and this will still find the results.

USAGE **ifind [-avVl] [-f fstype] [-d data_unit] [-n file] [-p par_inode] [-z ZONE] [-i imgtype] [-o imgoffset] [-b dev_sector_size]** *image* [images]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/ifind.html

EXAMPLE ifind -f fat -d 456 fat-img.dd
EXAMPLE ifind -f linux-ext2 -n "/etc/" linux-img.dd
EXAMPLE ifind -f ntfs -p 5 -l -z EST5EDT ntfs-img.dd

ils-sleuthkit

DESCRIPTION **ils** opens the named *image(s)* and lists inode information. By default, **ils** lists only the inodes of removed files. **ils** lists details about a range of meta data structures in a file system. Its output is in a delimited format that can be further processed.

USAGE **ils [-emOpvV] [-f** *fstype* **] [-s** *seconds* **] [-i** *imgtype* **] [-o** *imgoffset* **] [-b dev_sector_size]** *image* [images] [start-
USAGE stop]**ils [-aAlLvVzZ] [-f** *fstype* **] [-s** *seconds* **] [-i** *imgtype* **] [-o** *imgoffset* **]** *image* [images] [start-stop]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/ils.html

EXAMPLE ils -f openbsd -m images/root.dd >> data/body

istat

DESCRIPTION **istat** displays the uid, gid, mode, size, link number, modified, accessed, changed times, and all the disk units a structure has allocated.

USAGE **istat [-B** *num* **] [-f** *fstype* **] [-i imgtype] [-o imgoffset] [-b dev_sector_size] [-vV] [-z** *zone* **] [-s** *seconds* **]** *image* [images] inode

OPTIONS http://www.sleuthkit.org/sleuthkit/man/istat.html

EXAMPLE istat -f ntfs ntfs.dd 49

jcat

DESCRIPTION **jcat** shows the contents of a journal block in the file system journal. The inode address of the journal can be given or the default location will be used. Note that the block address is a journal block address and not a file system block. The raw output is given to STDOUT.

USAGE **jcat [-f** *fstype* **] [-vV] [-i imgtype] [-o imgoffset] [-b dev_sector_size]** *image* [images] **] [** *inode* **]** *jblk*

OPTIONS http://www.sleuthkit.org/sleuthkit/man/jcat.html

EXAMPLE jcat -f linux-ext3 img.dd 34 | xxd

mactime-sleuthkit

DESCRIPTION **mactime** creates an ASCII time line of file activity based on the body file specified by '-b' or from STDIN. The time line is written to STDOUT. The body file must be in the time machine format that is created by 'ils -m', 'fls -m', or the mac-robber tool.

USAGE **mactime [-b** *body* **] [-g** *group* file **] [-p** *password* file **] [-i** *(day|hour)* index file **] [-dhmVy] [-z** *TIME_ZONE* **] [DATE_RANGE]**

OPTIONS http://www.sleuthkit.org/sleuthkit/man/mactime.html

EXAMPLE mactime -b body.txt -d -i hour data/tl-hour-sum.txt > timeline.txt
EXAMPLE mactime -b body.txt -z EST5EDT 2002-03-01 > tl.03.01.2002.txt
EXAMPLE mactime -b body.txt 2002-03-01 > tl.03.01.2002.txt

missidentify

DESCRIPTION **missidentify** - Find executable files without an executable extension**. Miss Identify** looks at the header of every file it processes and determines if it is a PE executable (Windows executable). Such files can include programs, device drivers, and DLLs. By default the program dis- plays the filename if the extension of the file does not match one of the known executable extensions (.exe, .com, .sys, or .dll). Other options can make the program display the filename of all executable files.

USAGE missidentify [-rqablv] [-s|-S len] [-Vh] [FILES]

OPTIONS http://missidentify.sourceforge.net/manpage.txt

EXAMPLE missidentify –rabv /root/Desktop/WinHDD/ (list files)
EXAMPLE missidentify –rabv /root/Desktop/WinHDD/ > /root/Desktop/list1 (write the found files to list1)
EXAMPLE missidentify –ralv /root/Desktop/WinHDD/ > /root/Desktop/list2 (write all found files to lis2 with the path)

mmcat

DESCRIPTION **mmcat** outputs the contents of a specific volume to stdout. This allows you to extract the contents of a partition to a separate file.

USAGE **mmcat [-t** *mmtype* **] [-o** *offset* **] [-i** *imgtype* **] [-b dev_sector_size] [-vV]** *image [images] part_num*

OPTIONS http://www.sleuthkit.org/sleuthkit/man/mmcat.html

-t mmtype Specify the media management type. Use '-t list' to list the supported types. If not given, autodetection methods are used.

-o offset Specify the offset into the image where the volume containing the partition system starts. The relative offset of the partition system will be added to this value.

-b dev_sector_size The size, in bytes, of the underlying device sectors. If not given, the value in the image format is used (if it exists) or 512-bytes is assumed.

-i imgtype Identify the type of image file, such as raw or split. If not given, autodetection methods are used.

-v Verbose output of debugging statements to stderr

-V Display version

image [images] One (or more if split) disk images whose format is given with '-i'.

part_num Address of partition to process. See the mmls output to determine the address of the partitions.

EXAMPLE n/a

pdgmail

DESCRIPTION **pdgmail** - python script to gather gmail artifacts from a pd process memory dump. pdgmail is a memory forensics tool written in python used to recover Gmail account information from a memory dump. It looks for these things: contacts, last access records, GMail account names, message headers, message bodies

USAGE pdgmail [OPTIONS]

OPTIONS

-f, --file the file to use (stdin if no file given)
-b, --bodies don't look for message bodies (helpful if you're getting too many false positives on the mb regex)
-h, --help prints this
-v,--verbose be verbose (prints filename, other junk)
-V,--version prints just the version info and exits.

EXAMPLE pdgmail -f memorystrings.txt

readpst

DESCRIPTION **readpst** is a program that can read an Outlook PST (Personal Folders) file and convert it into an mbox file, a format suitable for KMail, a recursive mbox structure, or separate emails.

USAGE **readpst** [**-D**] [**-M**] [**-S**] [**-V**] [**-b**] [**-c** *format*] [**-d** *debug-file*] [**-e**] [**-h**] [**-j** *jobs*] [**-k**] [**-o** *output-directory*] [**-q**] [**-r**] [**-t** *output-type-codes*] [**-u**] [**-w**] pstfile

OPTIONS http://linux.die.net/man/1/readpst

EXAMPLE readpst yourfilename.pst
EXAMPLE readpst -k yourfilename.pst
EXAMPLE readpst -S -o out/ outlook.pst

reglookup

DESCRIPTION **reglookup** − Windows NT+ registry reader/lookup tool . The **RegLookup** project is devoted to direct analysis of Windows NT-based registry files. **reglookup** is designed to read Windows registry elements and print them out to stdout in a CSV-like format. It has filtering options to narrow the focus of the output. This tool is designed to work with on Windows NT-based registries.

USAGE **reglookup [options]** ***registry-file***

OPTIONS http://man.cx/reglookup(1)

EXAMPLE To read and print the contents of an entire system registry file: reglookup /mnt/win/c/WINNT/system32/config/system

EXAMPLE To limit the output to just those entries under the Services key: reglookup −p /ControlSet002/Services /mnt/win/c/WINNT/system32/config/system

EXAMPLE To limit the output to all registry values of type BINARY: reglookup −t BINARY /mnt/win/c/WINNT/system32/config/system

EXAMPLE And to limit the output to BINARY values under the Services key: reglookup −t BINARY −p /ControlSet002/Services /mnt/win/c/WINNT/system32/config/system

sorter

DESCRIPTION **sorter** is a Perl script that analyzes a file system to organize the allocated and unallocated files by file type. It runs the 'file' command on each file and organizes the files according to the rules in configuration files. Extension mismatching is also done to identify 'hidden' files. One can also provide hash databases for files that are known to be good and can be ignored and files that are known to be bad and should be alerted.

By default, the program uses the configuration files in the directory where The Sleuth Kit was installed. Those can be overruled with run-time options. There is a standard configuration file for all file system types and then a specific one for a given operating system.

USAGE **[-b** *size* **] [-e] [-E] [-h] [-l] [-md5] [-s] [-sha1] [-U] [-v] [-V] [-a** *hash_alert* **] [-c** *config* **] [-C** *config* **] [-d** *dir* **] [-m** *mnt* **] [-n** *nsrl_db* **] [-x** *hash_exclude* **] [-i imgtype] [-o imgoffset] [-f fstype] image [image] [meta_addr]**

OPTIONS http://www.sleuthkit.org/sleuthkit/man/sorter.html

EXAMPLE

```
# sorter -f ntfs -d data/sorter images/hda1.dd
# sorter -d data/sorter images/hda1.dd
# sorter -i raw -f ntfs -o 63 -d data/sorter images/hda.dd
```

EXAMPLE

```
# sorter -f ntfs -C /usr/local/sleuthkit/share/sort/images.sort -d data/sorter -h -s images/hda1.dd
```

srch-strings

DESCRIPTION no info

USAGE no info

OPTIONS no info

EXAMPLE no info

tsk_recover

DESCRIPTION **tsk_recover** recovers files to the *output_dir* from the *image.* By default recovers only unallocated files. With flags, it will export all files.

USAGE **tsk_recover [-vVae] [-f** *fstype* **] [-i** *imgtype* **] [-b** *dev_sector_size* **] [-o** *sector_offset* **] [-d** *dir_inum* **]** *image* [images] output_dir

OPTIONS http://www.sleuthkit.org/sleuthkit/man/tsk_recover.html

EXAMPLE tsk_recover ./image.dd ./recovered

vinetto

DESCRIPTION **Vinetto** is a forensics tool to examine Thumbs.db files.

USAGE vinetto [OPTIONS] [-s] [-U] [-o DIR] file

OPTIONS

```
--version   show program's version number and exit
 -h, --help  show this help message and exit
 -o DIR      write thumbnails to DIR
 -H          write html report to DIR
 -U          use utf8 encodings
 -s          create symlink of the image realname to the numbered name in
             DIR/.thumbs
```

EXAMPLE How to display metadata contained within a Thumbs.db file

```
vinetto /path/to/Thumbs.db
```

EXAMPLE How to extract the related thumbnails to a directory

```
vinetto -o /tmp/vinetto_output /path/to/Thumbs.db
```

EXAMPLE How to extract the related thumbnails to a directory and produce an html report to preview these thumbnails through your favorite browser.

```
vinetto -Ho /tmp/vinetto_output /path/to/Thumbs.db
```

EXAMPLE How to get a metadata report on all non deleted Thumbs.db files contained within a partition

```
find /mnt/hda2 -iname thumbs.db -printf "\n==\n %p \n\n" -exec vinetto {} \; 2>/tmp/vinetto_err.log
>/tmp/vinetto_hda2.txt
```

[66] FORENSIC CARVING TOOLS

- binwalk
- bulk_extractor
- foremost
- jls
- magicrescue
- pasco
- pev
- recoverjpeg
- fifiuti
- rifiuti2
- safecopy
- scalpel
- scrounge-ntfs

binwalk

DESCRIPTION **Binwalk** is a tool for searching a given binary image for embedded files and executable code. Specifically, it is designed for identifying files and code embedded inside of firmware images. **Binwalk** uses the **libmagic** library, so it is compatible with magic signatures created for the Unix file utility.

USAGE **binwalk** [OPTIONS] [FILE1] [FILE2] [FILE3] ...

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/binwalk.1.html

EXAMPLE text

bulk_extractor

DESCRIPTION **bulk_extractor** is a C++ program that scans a disk image, a file, or a directory of files and extracts useful information without parsing the file system or file system structures. The results are stored in feature files that can be easily inspected, parsed, or processed with automated tools. **bulk_extractor** also creates histograms of features that it finds, as features that are more common tend to be more important. We have made the following tools available for processing feature files generated by **bulk_extractor**: We have provided a small number of python programs that perform automated processing on feature files.

More info: http://digitalcorpora.org/downloads/bulk_extractor/doc/2012-08-08-bulk_extractor-tutorial.pdf

TIP see BEViewer – GUI for bulk_extractor: https://github.com/simsong/bulk_extractor/wiki/BEViewer

USAGE bulk_extractor [options] imagefile

OPTIONS bulk_extractor -h

EXAMPLE bulk_extractor -p 340731773 /corp/nps/drives/nps-2009-ubnist1/ubnist1.gen3.E01

EXAMPLE bulk_extractor -p 340731773-GZIP-9200 /corp/nps/drives/nps-2009-ubnist1/ubnist1.gen3.E01

EXAMPLE bulk_extractor -o charlie-2009-12-11 drives-redacted/charlie-2009-12-11.E01

foremost

DESCRIPTION Recover files from a disk image based on file types specified by the user using the -t switch. Supports: jpg, gif, png, bmp, avi, exe, mpg, wav, riff, wmv, mov, pdf, ole, doc, zip, rar, htm, cpp)

USAGE **foremost [-h][-V][-d][-vqwQT][-b**<blocksize>**][-o**<dir>**][-t**<type>**][-s**<num>**][-i**<file>**]**

OPTIONS http://manpages.ubuntu.com/manpages/hardy/en/man1/foremost.1.html

EXAMPLE foremost -s 100 -t jpg -i image.dd (Search for jpeg format skipping the first 100 blocks)
EXAMPLE foremost -av image.dd (Only generate an audit file, and print to the screen (verbose mode))
EXAMPLE foremost -t all -i image.dd (Search all defined types)
EXAMPLE foremost -t gif,pdf -i image.dd (Search for gifs and pdfs)
EXAMPLE foremost -vd -t ole,jpeg -i image.dd (Search for office documents and jpeg files in a Unix file system in verbose mode.)
EXAMPLE foremost image.dd (**Run the default case**)

jls

DESCRIPTION **jls** lists the records and entries in a file system journal. If index node (inode) is given, then it will look there for a journal. Otherwise, it will use the default location. The output lists the journal block number and a description.

USAGE **jls [-f** fstype **] [-vV] [-i imgtype] [-o imgoffset] [-b dev_sector_size]** image [images] [inode]

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/jls.1.html

EXAMPLE jls -f linux-ext3 img.dd

magicrescue

DESCRIPTION **Magic Rescue** opens devices for reading, scans them for file types it knows how to recover and calls an external program to extract them. It looks at "magic bytes" in file contents, so it can be used both as an undelete utility and for recovering a corrupted drive or partition. It works on any file system, but on very fragmented file systems it can only recover the first chunk of each file. These chunks are sometimes as big as 50MB, however.

*To invoke **magicrescue**, you must specify at least one device and the **-d** and **-r** options. See the "USAGE" section in this manual for getting started.*

USAGE magicrescue [options] devices

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/magicrescue.1.html

EXAMPLE magicrescue -r jpeg-jfif -r jpeg-exif -d ~/output /dev/hdb1

pasco

DESCRIPTION **pasco** is a tool to extract valuable information (from a forensics investigator point of view) from MS IE cache files (index.dat).

USAGE **Pasco** FILE

EXAMPLE pasco index.dat

pev

DESCRIPTION **pev** - show information about MS-Windows excutables files. Makes an analysis and shows useful information of PE32/PE32+ file given.

USAGE **pev** [OPTION]... file

OPTIONS http://manpages.ubuntu.com/manpages/raring/en/man1/pev.1.html

EXAMPLE pev -p putty.exe (To get only the Product Version of **putty.exe** file)
EXAMPLE pev -dc cards.dll (To show DOS and COFF file headers of **cards.dll**)
EXAMPLE pev svchost.exe (Display all possible information about **svchost.exe** file)

recoverjpeg

DESCRIPTION **Recoverjpeg** tries to identify jpeg pictures from a filesystem image. To achieve this goal, it scans the filesystem image and looks for a jpeg structure at blocks starting at 512 bytes boundaries. *Salvaged jpeg pictures are stored by default under the name imageXXXXX.jpg where XXXXX is a five digit number starting at zero. If there are more than 100,000 recovered pictures,* ***recoverjpeg*** *will start using six figures numbers and more as soon as needed, but the 100,000 first ones will use a five figures number. Options -f and -i can override this behaviour.*

USAGE **recoverjpeg** [options] device

OPTIONS http://manpages.ubuntu.com/manpages/hardy/man1/recoverjpeg.1.html

EXAMPLE recoverjpeg /dev/sdc (Recover as many pictures as possible from the memory card located in /dev/sdc)

EXAMPLE recoverjpeg -b 1 /dev/hdb1 (Recover as many pictures as possible from a crashed ReiserFS file system (which does not necessarily store pictures at block boundaries) in /dev/hdb1)

EXAMPLE recoverjpeg -b 1 -r 16m /dev/hdb1 (Do the same thing in a memory constrained environment where no more than 16MB of RAM can be used for the operation)

fifiuti

DESCRIPTION no info

USAGE no info

EXAMPLE no info

rifiuti2

DESCRIPTION **Rifiuti2** is a rewrite of **rifiuti**, a great tool from Foundstone folks for analyzing Windows Recycle Bin INFO2 file. Analysis of Windows Recycle Bin is usually carried out during Windows computer forensics. **Rifiuti2** can extract file deletion time, original path and size of deleted files and whether the deleted files have been moved out from the recycle bin since they are trashed. **Rifiuti2** supports the INFO2 file format found in Windows up to Windows XP and the new file format found in Vista, and the program is fully internationalized. If you need to analyse recycle bins of Windows Vista and Windows Server 2008, you should use the **rifiuti-vista** command, for other Windows platforms, you should use the **rifiuti** command.

USAGE **rifiuti** [**-x**] [**-tnl8**] [**-o** outfile] filename

USAGE **rifiuti-vista** [**-x**] [**-n8**] [**-o** outfile] file_or_directory

OPTIONS http://manpages.ubuntu.com/manpages/lucid/man1/rifiuti2.1.html

EXAMPLE rifiuti2 INFO2

EXAMPLE rifiuti2 –x INFO2

EXAMPLE rifiuti-vista win7recycle/

EXAMPLE rifiuti-vista -h

safecopy

DESCRIPTION **Safecopy** is a data recovery tool which tries to extract as much data as possible from a seekable, but problematic (i.e. damaged sectors) source - like floppy drives, hard disk partitions, CDs, ..., where other tools like dd would fail doe to I/O errors. **Safecopy** tries to get as much data from the source as possible without device dependent tricks. For example to get an ISO image from a copy protected or otherwise damaged CD-ROM, **cdrdao** and **bin2iso** would possibly do a better and faster job. **Safecopy** comes with preset options (named stages) to ease its use. These presets can be overridden by individual options.

USAGE safecopy [options] <source> <target>

OPTIONS http://manpages.ubuntu.com/manpages/lucid/man1/safecopy.1.html

EXAMPLE safecopy image1.dat combined.dat -I image2.badblocks -i blocksize2 \ -X image1.badblocks -x blocksize1

EXAMPLE safecopy /dev/filesystem -b <bsize> -s <X/bsize> -l <Y/bsize> (create an image of a device that starts at X and is Y in size)

MORE EXAMPLES http://safecopy.sourceforge.net/

scalpel

DESCRIPTION **Scalpel** is a fast file carver that reads a database of header and footer definitions and extracts matching files from a set of image files or raw device files. **Scalpel** is file system-independent and will carve files from FATx, NTFS, ext2/3, or raw partitions. It is useful for both digital forensics investigation and file recovery.

USAGE **scalpel** [**-b**] [**-c** <file>] [**-d**] [**-h**] [**-i** <file>] [**-m** <blocksize>] [**-n**] [**-o** <dir>] [**-O**] [**-p**] [**-r**] [**-s** <num>] [**-t**] [**-u**] [**-V**] [**-v**] [FILES]...

OPTIONS http://manpages.ubuntu.com/manpages/lucid/man1/scalpel.1.html

EXAMPLE scalpel /dev/sda1 -o output

scrounge-ntfs

DESCRIPTION Data recovery program for NTFS file systems. Reads each block of the hard disk to and retrieves rebuilds file system tree on another partition. It writes the files retrieved to another working file system. Certain information about the partition needs to be known in advance.

TIP You should have your partition information stored away in advance. This allows reliable retrieval of file info. If you don't however, there's some hope. You can guess at it in many cases.

USAGE **scrounge-ntfs -l** *disk*
USAGE **scrounge-ntfs -s** *disk*
USAGE **scrounge-ntfs** [**-m** *mftoffset*] [**-c** *clustersize*] [**-o** *outdir*] *disk start end*

OPTIONS http://thewalter.net/stef/software/scrounge/scrounge-ntfs.html

EXAMPLE scrounge-ntfs -l /dev/sdb

[67] FORENSIC HASHING TOOLS

- md5deep
- rahash2

md5deep

DESCRIPTION **md5deep** is a set of programs to compute MD5, SHA-1, SHA-256, Tiger, or Whirlpool message digests on an arbitrary number of files.

md5deep is similar to the **md5sum** program found in the GNU **Coreutils** package, but has the following additional features:

- Recursive operation - md5deep is able to recursive examine an entire directory tree. That is, compute the MD5 for every file in a directory and for every file in every subdirectory.
- Comparison mode - md5deep can accept a list of known hashes and compare them to a set of input files. The program can display either those input files that match the list of known hashes or those that do not match. Hashes sets can be drawn from Encase, the National Software Reference Library, iLook Investigator, Hashkeeper, md5sum, BSD md5, and other generic hash generating programs. Users are welcome to add functionality to read other formats too!
- Time estimation - md5deep can produce a time estimate when it's processing very large files.
- Piecewise hashing - Hash input files in arbitrary sized blocks
- File type mode - md5deep can process only files of a certain type, such as regular files, block devices, etc.

USAGE n/a

EXAMPLE n/a

rahash2

DESCRIPTION **rahash2** - **radare** tool for creating hashes. **rahash2** is designed to work with blocks like **radare** does. So this way you can generate multiple checksums from a single file, and then make a faster comparision of the blocks to find the part of the file that has changed.

This is useful in forensic tasks, when progressively analyzing memory dumps to find the places where it has changed and then use **'radiff**' to get a closer look to these changes.

This is the default work way for **rahash2**. So lets generate a **rahash2** checksumming file and then use it to check if something has changed. The default block size is 32 KBytes. You can change it by using the -b flag.

USAGE rahash2 [-action] [-options] [source] [hash-file]

OPTIONS rahash2 –h

OPTIONS check rahash http://radare.org/doc/html/Section18.1.html

EXAMPLE rahash2 -a md5 -s 'hello world'

[68] FORENSIC IMAGING TOOLS

- affcat
- affconvert
- blkls
- dc3dd
- dcfldd
- ddrescue
- ewfacquire
- ewfacquirestream
- ewfexport
- ewfinfo
- ewfverify
- fsstat
- guymager
- img_cat
- img_stat
- mmls
- mmstat
- tsk_gettimes

affcat

DESCRIPTION The **Advanced Forensic Format (AFF)** is on-disk format for storing computer forensic information. Critical features of **AFF** include:

- **AFF** allows you to store both computer forensic data and associated metadata in one or more files.
- **AFF** allows files to be digital singed, to provide for chain-of-custody and long-term file integrity.
- **AFF** allows for forensic disk images to stored encrypted and decrypted on-the-fly for processing. This allows disk images containing privacy sensitive material to be stored on the Internet.
- **AFF** is an open format unencumbered by copyright or patent protection. The **AFFLIB** library that implements **AFF** is available for use in both Open Source and proprietary tools.

AFF Library and Toolkit is a set of programs for working with computer forensic information.

affcat - outputs the contents of an image file to stdout. Image files that are not raw but are recognized by **AFF** will be output in raw format. Missing pages will not be padded, but the fact that they are missing will be noted on STDERR.

USAGE https://github.com/simsong/AFFLIBv3/blob/master/man/affcat.1

EXAMPLE n/a

affconvert

DESCRIPTION The **Advanced Forensic Format (AFF)** is on-disk format for storing computer forensic information. Critical features of **AFF** include:

- **AFF** allows you to store both computer forensic data and associated metadata in one or more files.
- **AFF** allows files to be digital singed, to provide for chain-of-custody and long-term file integrity.
- **AFF** allows for forensic disk images to stored encrypted and decrypted on-the-fly for processing. This allows disk images containing privacy sensitive material to be stored on the Internet.
- **AFF** is an open format unencumbered by copyright or patent protection. The **AFFLIB** library that implements **AFF** is available for use in both Open Source and proprietary tools.

AFF Library and Toolkit is a set of programs for working with computer forensic information.

affconvert - *converts raw -> aff, aff -> raw, aff -> aff (recompressing/uncompressing)*

USAGE https://github.com/simsong/AFFLIBv3/blob/master/tools/affconvert.cpp

EXAMPLE n/a

blkls

DESCRIPTION **blkls** opens the named *image(s)* and copies file system data units (blocks). By default, **blkls** copies the contents of unallocated data blocks. **blkls** was called **dls** in TSK versions prior to 3.0.0. **blkls** was called **unrm** in TCT. **blkls** lists details about file system data units. In its default mode, it outputs the unallocated data unit contents to STDOUT. It can also list the details about which are allocated and which are not.

USAGE **blkls [-aAelsvV] [-f** *fstype* **] [-i** *imgtype* **] [-o** *imgoffset* **]** *[-b* dev_sector_size] image [images] [start-stop]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/blkls.html

EXAMPLE blkls images/wd0e.dd > output/wd0e.blkls

EXAMPLE blkls images/hda1.dd 32768-65535 > output/hda1-grp1.blkls

dc3dd

DESCRIPTION **dc3dd (Department of Defense Cyber Crime Center)** is a patched version of **GNU dd** with added features for computer forensics. It is a powerful imaging tool that will create a file that contains an exact replica of a hard drive.

More info: http://www.myfixlog.com/fix.php?fid=33

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

dcfldd

DESCRIPTION **dcfldd** is an enhanced version of **GNU dd** with features useful for forensics and security.

Based on the **dd** program found in the GNU Coreutils package, **dcfldd** has the following additional features:

- Hashing on-the-fly - dcfldd can hash the input data as it is being transferred, helping to ensure data integrity.
- Status output - dcfldd can update the user of its progress in terms of the amount of data transferred and how much longer operation will take.
- Flexible disk wipes - dcfldd can be used to wipe disks quickly and with a known pattern if desired.
- Image/wipe Verify - dcfldd can verify that a target drive is a bit-for-bit match of the specified input file or pattern.
- Multiple outputs - dcfldd can output to multiple files or disks at the same time.
- Split output - dcfldd can split output to multiple files with more configurability than the split command.
- Piped output and logs - dcfldd can send all its log data and output to commands as well as files natively.

USAGE **dcfldd** [*OPTION*]...

OPTIONS http://linux.die.net/man/1/dcfldd

EXAMPLE dcfldd if=/dev/hda1 of=/mnt/data/image.dd hashlog=/mnt/data/md5hash2.txt

ddrescue

DESCRIPTION **ddrescue** is a raw disk imaging tool that "copies data from one file or block device to another, trying hard to rescue data in case of read errors." The application is developed as part of the GNU project and has written with UNIX/Linux in mind.

More info: http://www.gnu.org/software/ddrescue/manual/ddrescue_manual.html

USAGE ddrescue [options] infile outfile [logfile]

OPTIONS http://www.forensicswiki.org/wiki/Ddrescue

EXAMPLE Rescue an entire hard disk /dev/sda to another disk /dev/sdb

ddrescue -n /dev/sda /dev/sdb rescue.log (copy the error free areas first)
ddrescue -r 1 /dev/sda /dev/sdb rescue.log (attempt to recover any bad sectors)

EXAMPLE Rescue a CD-ROM in /dev/cdrom

ddrescue -b 2048 /dev/cdrom cdimage logfile (write cdimage to a blank CD-ROM)

EXAMPLE Rescue an ext2 partition in /dev/hda2 to /dev/hdb2

ddrescue -r3 /dev/hda2 /dev/hdb2 logfile
e2fsck -v -f /dev/hdb2
mount -t ext2 -o ro /dev/hdb2 /mnt
(This will overwrite ALL data on the partition you are copying to. If you do not want to do that, rather create an image of the partition to be rescued)

ewfacquire

DESCRIPTION **ewfacquire** is a utility to acquire media data from a *source* and store it in EWF format (Expert Witness Compression Format). **ewfacquire** acquires media data in a format equivalent to EnCase and FTK imager, including meta data. Under Linux, FreeBSD, NetBSD, OpenBSD, MacOS-X/Darwin **ewfacquire** supports reading directly from device files. On other platforms **ewfacquire** can convert a raw (dd) image into the EWF format.

ewfacquire *is part of the* ***libewf*** *package.* ***libewf*** *is a library to support the Expert Witness Compression Format (EWF).* ***libewf*** *supports both the SMART format (EWF-S01) and the EnCase format (EWF-E01).* ***libewf*** *currently does not support the Logical Volume format (EWF-L01). EWF-X is an expirimental format intended for testing purposes to enhance the EWF format.* ***libewf*** *allows you to read and write media data in the EWF format.*

USAGE **ewfacquire** [**-A** *codepage*] [**-b** *amount_of_sectors*] [**-B** *amount_of_bytes*] [**-c** *compression_type*] [**-C** *case_number*] [**-d** *digest_type*] [**-D** *description*] [**-e** *examiner_name*] [**-E***evidence_number*] [**-f** *format*] [**-g** *amount_of_sectors*] [**-l** *log_filename*] [**-m** *media_type*] [**-M** *media_flags*] [**-N** *notes*] [**-o** *offset*] [**-p** *process_buffer_size*] [**-P** *bytes_per_sector*] [**-r***read_error_retries*] [**-S** *segment_file_size*] [**-t** *target*] [**-2** *secondary_target*] [**-hqRsuvVw**] *source*

OPTIONS http://linux.die.net/man/1/ewfacquire

EXAMPLE ewfacquire /dev/fd0

ewfacquirestream

DESCRIPTION **ewfacquirestream** is a utility to acquire media data from stdin and store it in EWF format (Expert Witness Format). **ewfacquirestream** acquires media data in a format equivalent to EnCase and FTK imager, including meta data. Under Linux, FreeBSD, NetBSD, OpenBSD, MacOS-X/Darwin

ewfacquirestream *is part of the* ***libewf*** *package.* ***libewf*** *is a library to support the Expert Witness Compression Format (EWF).* ***libewf*** *supports both the SMART format (EWF-S01) and the EnCase format (EWF-E01).* ***libewf*** *currently does not support the Logical Volume format (EWF-L01). EWF-X is an expirimental format intended for testing purposes to enhance the EWF format.* ***libewf*** *allows you to read and write media data in the EWF format.*

USAGE **ewfacquirestream** [**-A** *codepage*] [**-b** *amount_of_sectors*] [**-B** *amount_of_bytes*] [**-c** *compression_type*] [**-C** *case_number*] [**-d** *digest_type*] [**-D** *description*] [**-e** *examiner_name*] [**-E** *evidence_number*] [**-f** *format*] [**-l** *log_filename*] [**-m** *media_type*] [**-M** *media_flags*] [**-N** *notes*] [**-o** *offset*] [**-p** *process_buffer_size*] [**-S** *segment_file_size*] [**-t** *target*] [**-2***secondary_target*] [**-hqsvVw**]

OPTIONS http://linux.die.net/man/1/ewfacquirestream

EXAMPLE ewfacquirestream -C 1 -D Floppy -E 1.1 -e 'John D.' -N 'Just a floppy in my system' -m removable -M physical -t floppy </dev/fd0

ewfexport

DESCRIPTION **ewfexport** is a utility to export media data stored in EWF files.

ewfexport *is part of the* ***libewf*** *package.* ***libewf*** *is a library to support the Expert Witness Compression Format (EWF).* ***libewf*** *supports both the SMART format (EWF-S01) and the EnCase format (EWF-E01).* ***libewf*** *currently does not support the Logical Volume format (EWF-L01). EWF-X is an expirimental format intended for testing purposes to enhance the EWF format.* ***libewf*** *allows you to read and write media data in the EWF format.*

USAGE **ewfexport** [**-A** *codepage*] [**-B** *amount_of_bytes*] [**-c** *compression_type*] [**-d** *digest_type*] [**-f** *format*] [**-l** *log_filename*] [**-o** *offset*] [**-p** *process_buffer_size*] [**-S** *segment_file_size*] [**-t***target*] [**-hqsuvVw**] *ewf_files*

OPTIONS http://linux.die.net/man/1/ewfexport

EXAMPLE ewfexport floppy.E01

ewfinfo

DESCRIPTION **ewfinfo** is a utility to show meta data stored in EWF files.

ewfinfo *is part of the* ***libewf*** *package.* ***libewf*** *is a library to support the Expert Witness Compression Format (EWF).* ***libewf*** *supports both the SMART format (EWF-S01) and the EnCase format (EWF-E01).* ***libewf*** *currently does not support the Logical Volume format (EWF-L01). EWF-X is an expirimental format intended for testing purposes to enhance the EWF format.* ***libewf*** *allows you to read and write media data in the EWF format.*

USAGE **ewfinfo** [**-A** *codepage*] [**-d** *date_format*] [**-ehimvV**] *ewf_files*

OPTIONS http://linux.die.net/man/1/ewfinfo

EXAMPLE ewfinfo -d dm floppy.E01

ewfverify

DESCRIPTION **ewfverify** is a utility to verify media data stored in EWF files.

ewfverify *is part of the* ***libewf*** *package.* ***libewf*** *is a library to support the Expert Witness Compression Format (EWF).* ***libewf*** *supports both the SMART format (EWF-S01) and the EnCase format (EWF-E01).* ***libewf*** *currently does not support the Logical Volume format (EWF-L01). EWF-X is an expirimental format intended for testing purposes to enhance the EWF format.* ***libewf*** *allows you to read and write media data in the EWF format.*

USAGE **ewfverify** [**-A** *codepage*] [**-d** *digest_type*] [**-l** *log_filename*] [**-p** *process_buffer_size*] [**-hqvVw**] *ewf_files*

OPTIONS http://linux.die.net/man/1/ewfverify

EXAMPLE ewfverify floppy.E01

fsstat

DESCRIPTION **fsstat** displays the details associated with a file system. The output of this command is file system specific. At a minimum, the range of meta-data values (inode numbers) and content units (blocks or clusters) are given. Also given are details from the Super Block, such as mount times and features. For file systems that use groups (FFS and EXT2FS), the layout of each group is listed. For a FAT file system, the FAT table is displayed in a condensed format. Note that the data is in sectors and not in clusters.

USAGE **fsstat [-f** *fstype* **] [-i imgtype] [-o imgoffset] [-b dev_sector_size] [-tvV]** *image* [images]

OPTIONS

-t type Print the file system type only.

-f fstype Specify the file system type. Use '-f list' to list the supported file system types. If not given, autodetection methods are used.

-i imgtype Identify the type of image file, such as raw. Use '-i list' to list the supported types. If not given, autodetection methods are used.

-o imgoffset The sector offset where the file system starts in the image.

-b dev_sector_size The size, in bytes, of the underlying device sectors. If not given, the value in the image format is used (if it exists) or 512-bytes is assumed.-vVerbose output of debugging statements to stderr

-V Display version

image [images] The disk or partition image to read, whose format is given with '-i'. Multiple image file names can be given if the image is split into multiple segments. If only one image file is given, and its name is the first in a sequence (e.g., as indicated by ending in '.001'), subsequent image segments will be included automatically.

EXAMPLE fsstat usb.img

guymager

DESCRIPTION **Guymager** is an open source forensic imager. It focuses on user friendliness and high speed. It is one of the first forensic imaging tools to utilize mutli-threading for the imaging process. **guymager** is a free forensic imager for media acquisition. Guymager can generate flat (dd), EWF (E01) and AFF images and it supports disk cloning.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

img_cat

DESCRIPTION **img_cat** outputs the contents of an image file. Image files that are not raw will have embedded data and metadata. **img_cat** will output only the metadata. This allows you to convert an embedded format to raw or to calculate the MD5 hash of the data by piping the output to the appropriate tool.

USAGE **img_cat [-i imgtype] [-b dev_sector_size] [-b start_sector] [-e stop_sector] [-vV]** *image* [images]

USAGE img_cat [-i imgtype] [-vV] image [images]

OPTIONS

-i imgtype Identify the type of image file, such as raw or aff. Use '-i list' to list the supported types. If not given, autodetection methods are used.

-b dev_sector_size The size, in bytes, of the underlying device sectors. If not given, the value in the image format is used (if it exists) or 512-bytes is assumed.

-s start_sector The sector number to start at.

-e stop_sector The sector number to stop at.

-v Verbose output of debugging statements to stderr

-V Display version

image [images] The disk or partition image to read, whose format is given with '-i'. Multiple image file names can be given if the image is split into multiple segments. If only one image file is given, and its name is the first in a sequence (e.g., as indicated by ending in '.001'), subsequent image segments will be included automatically.

EXAMPLE img_stat usb.img

img_stat

DESCRIPTION **img_stat** displays the details associated with an image file. The output of this command is image format specific. At a minimum, the size will be given and the byte range of each file will be given for split image formats.

USAGE **img_stat [-i imgtype] [-b dev_sector_size] [-tvV]** *image* [images]

OPTIONS

-i imgtype Identify the type of image file, such as raw. Use '-i list' to list the supported types. If not given, autodetection methods are used.

-b dev_sector_size The size, in bytes, of the underlying device sectors. If not given, the value in the image format is used (if it exists) or 512-bytes is assumed.

-t Print the image type only.

-v Verbose output of debugging statements to stderr

-V Display version

Image [images] The disk or partition image to read, whose format is given with '-i'. Multiple image file names can be given if the image is split into multiple segments. If only one image file is given, and its name is the first in a sequence (e.g., as indicated by ending in '.001'), subsequent image segments will be included automatically.

EXAMPLE img_stat usb.img (Display image type and size of image file usb.img)

mmls

DESCRIPTION **mmls** displays the contents of a volume system (media management). In general, this is used to list the partition table contents so that you can determine where each partition starts. The output identifies the type of partition and its length, which makes it easy to use 'dd' to extract the partitions. The output is sorted based on the starting sector so it is easy to identify gaps in the layout.

USAGE **mmls [-t** *mmtype* **] [-o** *offset* **] [-i** *imgtype* **] [-b dev_sector_size] [-BrvV] [-aAmM]** *image* [images]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/mmls.html

EXAMPLE mmls -t dos part2

EXAMPLE mmls -t list

mmstat

DESCRIPTION **mmstat** - display details about the media management system (partition tables). **mms** displays the general details of the media management systems, which include partition tables and disk labels. Mainly, the type is given. **mmstat** simply displays the system volume information.

USAGE **mmstat [-t** mmtype **] [-o** offset **] [-i** imgtype **] [-vV]** image [images]

OPTIONS

-t mmtype Specify the media management type. Use the -? option for supported types.

-o offset Specify the offset into the image where the volume containing the partition system starts. The relative offset of the partition system will be added to this value.

-i imgtype Identify the type of image file, such as raw or split. Raw is the default.

-v Verbose output of debugging statements to stderr

-V Display version

image [images] One (or more if split) disk images whose format is given with '-i'.

EXAMPLE mmstat part2 (display system volume information for partition part2)

tsk_gettimes

DESCRIPTION **tsk_gettimes** examines each of the file systems in a disk image and returns the data about them in the MACtime body format (the same as running 'fls -m' on each file system). The output of this can be used as input to mactime to make a timeline of file activity. The data is printed to STDOUT, which can then be redirected to a file.

USAGE **tsk_gettimes [-vV] [-f** *fstype* **] [-i** *imgtype* **] [-b** *dev_sector_size* **] [-z** *zone* **] [-s** *seconds* **]** *image* [images]

OPTIONS http://www.sleuthkit.org/sleuthkit/man/tsk_gettimes.html

EXAMPLE tsk_gettimes ./image.dd > body.txt (collect data about image image.dd)

[69] FORENSIC SUITES

- autopsy
- dff

autopsy

DESCRIPTION **Autopsy** is a graphical interface to the command line digital investigation analysis tools in **The Sleuth Kit**. Together, they can analyze Windows and UNIX disks and file systems (NTFS, FAT, UFS1/2, Ext2/3).

As **Autopsy** is HTML-based, you can connect to the **Autopsy** server from any platform using an HTML browser. **Autopsy** provides a "File Manager"-like interface and shows details about deleted data and file system structures.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

dff

DESCRIPTION DFF (dff.pl) is a Perl script that is a transparent wrapper around dvipdfm(1), dvips(1) and pdftex(1), fixing many of their bugs concerning Type1 fonts, thus providing a consistent environment for creating PDFs with subsetted vector Type1 fonts using TeX. DFF should be used with teTeX.

USAGE dff.pl [dff-options] dvipdfm [dvipdfm-options] <input.dvi>
USAGE dff.pl [dff-options] dvips [dvips-options] <input.dvi>
USAGE dff.pl [dff-options] pdf[la]tex [pdflatex-options] <input.tex>
USAGE dff.pl [dff-options] xdvi{|gz|cfg} [xdvi-options] <input.dvi>
USAGE dff.pl [dff-options] ps2pk # create dff_psk.map

```
The dff-options are:
--dff-do[=1]                ignored
--dff-force-paper[=1]       force paper size for dvips(1)
--dff-onepass[=1]           run dvips/dvipdfm/pdftex only once, if applicable
[--dff-basefonts=guess]     upload base 35 fonts: adobe-kb urw-kb urw-urw
[--dff-map=FONT.MAP ...]    load this Type1 font map file
[--dff-map-nc=FONT.MAP ...] load this Type1 font map file, but not from curdir
[--dff-map-dir=DIR ...]     load all .map files in this dir (recursively)
[--dff-map-dir-if=FILE ...] load all .map files from dir of FILE if FILE
[--dff-base14-upload=0]     upload base 14 fonts to printer (!adobe-bi)
[--dff-maps-only=0]         generate only dff_*.map, don't run driver
--dff-type1fix[=1]          use type1fix.pl to fix all Type1 fonts
--dff-to-curdir[=1]         put target PDF to curdir (not DVI dir)
[--dff-path-maps=1]         use font maps found in \$DFF_MAP_PATH
[--dff-dvips-maps=0]        use font maps found in dvips config.ps
--dff-pdftex-maps[=1]       use font maps found in pdftex pdftex.cfg
[--dff-dvipdfm-maps=0]      use font maps found in dvipdfm config
[--dff-find-fonts=1]        find fonts other than in dff.fnt/
[--dff-driver=DRIVER]       driver is dvips or `strace dvips' etc.
[--dff-subset=MODE]         MODE: subset, whole, none, [unchanged]
```

EXAMPLE dff.pl -dff-map-dir=mydir -dff-map-nc=ps2pk.map -dff-pdftex-maps=0 ps2pk

[70] NETWORK FORENSICS

- p0f

p0f

DESCRIPTION **p0f** uses a fingerprinting technique based on analysing the structure of a TCP/IP packet to determine the operating system and other configuration properties of a remote host. The process is completely passive and does not generate any suspicious network traffic. The other host has to either:

connect to your network - either spontaneously or in an induced manner, for example when trying to establish a ftp data stream, returning a bounced mail, performing auth lookup, using IRC DCC, external html mail image reference and so on, or be contacted by some entity on your network using some standard means (such as a web browsing); it can either accept or refuse the connection.

The method can see thru packet firewalls and does not have the restrictions of an active fingerprinting. The main uses of passive OS fingerprinting are attacker profiling (IDS and honeypots), visitor profiling (content optimization), customer/user profiling (policy enforcement), pen-testing, etc.

USAGE p0f [-f file] [-i device] [-s file] [-o file]
[-w file] [-Q sock [-0]] [-u user] [-FXVNDUKASCMROqtpvdlrx]
[-c size] [-T nn] [-e nn] ['filter rule']

OPTIONS http://www.aldeid.com/wiki/P0f

EXAMPLE p0f -i eth1 –vt (The following command will start p0f)

EXAMPLE p0f -i eth1 –vto output.txt (The output of the ingerprint information can also be directed to a file using the –o option)

[71] PASSWORD FORENSIC TOOLS

- chntpw

chntpw

DESCRIPTION **chntpw** is a Linux utility to (re)set the password of any user that has a valid (local) account on your WinNT or Win2000 system, by modifying the crypted password in the registry's SAM file. You do not need to know the old password to set a new one. It works offline (i.e., you have to shutdown your computer and boot off a linux floppy disk). The bootdisk includes stuff to access NTFS partitions and scripts to glue the whole thing together. This utility works with SYSKEY and includes the option to turn it off. A bootdisk image is provided.

USAGE chntpw [options] <systemfile> [securityfile] [otherreghive] [...]

OPTIONS: chntpw -h

EXAMPLE chntpw -i sam (starts the program in the interactive mode and specifies the name of the Windows sam file)

[72] PDF FORENSIC TOOLS

- pdf-parser
- peepdf

pdf-parser

DESCRIPTION This tool will parse a PDF document to identify the fundamental elements used in the analyzed file. It will not render a PDF document. It provides features to extract raw data from PDF documents, like compressed images. **pdf-parser** can deal with malicious PDF documents that use obfuscation features of the PDF language.

USAGE pdf-parser.py [options] pdf-file

OPTIONS

--versions how program's version number and exit
-h, --helps how this help message and exit
-s SEARCH, --search=SEARCH string to search in indirect objects (except streams)
-f, --filter pass stream object through filters (FlateDecode, ASCIIHexDecode, ASCII85Decode, LZWDecode and RunLengthDecode only)
-o OBJECT, --object=OBJECTid of indirect object to select (version independent)
-r REFERENCE, --reference=REFERENCE id of indirect object being referenced (version independent)
-e ELEMENTS, --elements=ELEMENTS type of elements to select (cxtsi)
-w, --raw raw output for data and filters
-a, --stats display stats for pdf document
-t TYPE, --type=TYPE type of indirect object to select
-v, --verbose display malformed PDF elements
-x EXTRACT, --extract=EXTRACT filename to extract to
-H, --hash display hash of objects
-n, --nocanonicalizedoutput do not canonicalize the output
-d DUMP, --dump=DUMP filename to dump stream content to
-D, --debug display debug info

EXAMPLE pdf-parser.py http://example.com/doc.pdf
EXAMPLE pdf-parser.py maldoc.zip

peepdf

DESCRIPTION **peepdf** is a Python tool to explore PDF files in order to find out if the file can be harmful or not. The aim of this tool is to provide all the necessary components that a security researcher could need in a PDF analysis without using 3 or 4 tools to make all the tasks. With **peepdf** it's possible to see all the objects in the document showing the suspicious elements, supports all the most used filters and encodings, it can parse different versions of a file, object streams and encrypted files. With the installation of **SpiderMonkey** and **Libemu** it provides JavaScript and Shell code analysis wrappers too. Apart of this it's able to create new PDF files and to modify existent ones.

USAGE ./peepdf.py [options] PDF_file

OPTIONS

-h, --help show this help message and exit
-i, --interactive Sets console mode.
-f, --force-mode Sets force parsing mode to ignore errors.
-l, --loose-mode Sets loose parsing mode to catch malformed objects.
-s SCRIPTFILE, --load-script=SCRIPTFILE Load the commands stored in the specified file and execute them.

EXAMPLE ./peepdf.py –l

EXAMPLE ./peepdf.py -f fcexploit.pdf File: fcexploit.pdf

[73] RAM FORENSIOC TOOLS

- volafox
- volatility

volafox

DESCRIPTION Volafox is a Mac OS X memory analysis tool based on volatility.

USAGE python volafox.py -i MEMORY_IMAGE -s KERNEL_IMAGE -o INFORMATION

EXAMPLE volafox.py -i MemoryImage.mem -s mach_kernel -o machine_info - display mac os x version info
EXAMPLE volafox.py -i MemoryImage.mem -s mach_kernel -o mount_info - dispaly mounted device info
EXAMPLE volafox.py -i MemoryImage.mem -s mach_kernel -o proc_info - process list information
EXAMPLE volafox.py -i MemoryImage.mem -s mach_kernel -o proc_info -x [PID] - more info from a process with

volatility

DESCRIPTION The **Volatility Framework** is a completely open collection of tools, implemented in Python under the GNU General Public License, for the extraction of digital artefacts from volatile memory (RAM) samples. The extraction techniques are performed completely independent of the system being investigated but offer unprecedented visibility into the runtime state of the system. The framework is intended to introduce people to the techniques and complexities associated with extracting digital artefacts from volatile memory samples and provide a platform for further work into this exciting area of research.

USAGE python vol.py -f <path to mem image> --profile=<profile_name> plugin_name <plugin_options>

OPTIONS ./vol.py -h

EXAMPLE ./vol.py imageinfo -f /var/forensics/images/WV01_clean.dd

EXAMPLE ./vol.py --profile=WinXPSP3x86 pslist -f /var/forensics/images/WV01_clean.dd

EXAMPLE ./vol.py --profile=WinXPSP3x86 pslist -f /var/forensics/images/WV01_clean.dd | egrep '(notepad.exe|sol.exe|cmd.exe|nc.exe|dd.exe|iexplore.exe|helix.exe)'

EXAMPLE ./vol.py --profile=WinXPSP3x86 **connections** -f /var/forensics/images/WV01_clean.dd

[74] REPORTING TOOLS: EVIDENCE MANAGEMENT

- casefile
- keepnote
- magictree
- maltego
- metagoofil
- truecrypt

casefile

DESCRIPTION **CaseFile** gives you the ability to quickly add, link and analyze data having the same graphing flexibility and performance as **Maltego** without the use of transforms. Combining **Maltego's** fantastic graph and link analysis this tool allows for analysts to examine links between manually added data to mind map your information.

- **CaseFile** is a visual intelligence application that can be used to determine the relationships and real world links between hundreds of different types of information.
- It gives you the ability to quickly view second, third and n-th order relationships and find links otherwise undiscoverable with other types of intelligence tools.
- **CaseFile** comes bundled with many different types of entities that are commonly used in investigations allowing you to act quickly and efficiently. **CaseFile** also has the ability to add custom entity types allowing you to extend the product to your own data sets.

USAGE n/a; GUI tool

EXAMPLE n/a, GUI tool

keepnote

DESCRIPTION **KeepNote** is a note taking application that works on Windows, Linux, and MacOS X. With **KeepNote**, you can store your class notes, TODO lists, research notes, journal entries, paper outlines, etc. in a simple notebook hierarchy with rich-text formatting, images, and more. Using full-text search, you can retrieve any note for later reference.

Features

- *Rich-text formatting (e.g. Bullet point lists, Inline images)*
- *Hierarchical organization for notes*
- *Web links and note-to-note links*
- *Full-text search*
- *Integrated screenshot*
- *File attachments*
- *Spell checking (via gtkspell)*
- *Auto-saving*
- *Built-in backup and restore (archive to zip files)*
- *Extensions (i.e. "plugins")*
- *Cross-platform (Linux, Windows, MacOS X)*

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

magictree

DESCRIPTION **MagicTree**, a Java application created by Gremwell, is an actively supported data collection and reporting tool. It manages data using nodes in a tree-structure. **MagicTree** allows for XML data imports and has XSLT transforms for many popular formats such as:

- Nessus (v1 and v2)
- Nikto
- Nmap
- Burp
- Qualys
- Imperva Scuba
- OpenVas

More info: http://www.gremwell.com/magictreedoc/6fabd1f6.html

USAGE n/a; GUI tool

EXAMPLE n/a GUI tool

maltego

DESCRIPTION **Maltego** is a unique platform developed to deliver a clear threat picture to the environment that an organization owns and operates. **Maltego** can locate, aggregate and visualize this information. **Maltego** is a program that can be used to determine the relationships and real world links between people, groups of people (social networks), companies, organizations, web sites, phrases, affiliations, documents and files, internet infrastructure (domains, DNS names, netblocks, IP addresses).

USAGE n/a, GUI tool

EXAMPLE n/a, GUI tool

metagoofil

DESCRIPTION **Metagoofil** is an information gathering tool designed for extracting metadata of public/indexed documents (pdf,doc,xls,ppt,odp,ods) available in the target/victim websites.
The output is a file that can reveal:

- relevant metadata information
- usernames (potential targets for brute force attacks on open services like ftp, pop3, auths in web apps, ...)
- list of disclosed paths in the metadata

USAGE python metagoofil.py <option>

OPTIONS

- **-d <domain>** Domain to search
- **-f <type>** Filetype to download (all,pdf,doc,xls,ppt,odp,ods, etc)
- **-l <number>** Limit of results to work with (default 100)
- **-o <path>** Output file (html format)
- **-t <path>** Target directory to download files

EXAMPLE python metagoofil.py \ -d ******club.net \ -l 100 \ -f all \ -o output.html \ -t output-files

truecrypt

DESCRIPTION text

USAGE TrueCrypt [/a [devices|favorites]] [/b] [/c [y|n]] [/d [*drive letter*]] [/e] [/f] [/h [y|n]] [/k *keyfile or search path*] [/l *drive letter*] [/m {bk|rm|recovery|ro|sm|ts}] [/p*password*] [/q [background|preferences]] [/s] [/v *volume*] [/w]

OPTIONS http://www.truecrypt.org/docs/?s=command-line-usage

EXAMPLE truecrypt /q /v d:\myvolume (Mount the volume *d:\ myvolume* as the first free drive letter, using the password prompt (the main program window will not be displayed))

EXAMPLE truecrypt /q /dx (Dismount a volume mounted as the drive letter X (the main program window will not be displayed))

EXAMPLE truecrypt /v myvolume.tc /lx /a /p MyPassword /e /b (Mount a volume called myvolume.tc using the password MyPassword, as the drive letter X. TrueCrypt will open an explorer window and beep; mounting will be automatic)

[75] MEDIA CAPTURE

- cutycapt
- recordmydesktop

cutycapt

DESCRIPTION **CutyCapt** is a small cross-platform command-line utility to capture WebKit's rendering of a web page into a variety of vector and bitmap formats, including SVG, PDF, PS, PNG, JPEG, TIFF, GIF, and BMP.

USAGE CutyCapt --url=<target ip> --out=<output file>.<extension>

OPTIONS http://cutycapt.sourceforge.net/

EXAMPLE CutyCapt --url=http://www.example.org/ --out=localfile.png

TIP

Using CutyCapt without X server

You cannot use CutyCapt without an X server, but you can use e.g. Xvfb as light-weight server if you are not running an interactive graphical desktop environment.

For example, you could use:

- *% xvfb-run --server-args="-screen 0, 1024x768x24" ./CutyCapt --url=... --out=...*

recordmydesktop

DESCRIPTION **recordMyDesktop** produces a file (default out.ogv) that contains a video and audio recording of a linux desktop session. The default behaviour of recording is to mark areas that have changed (through libxdamage) and update the frame. This behaviour can be changed (option --full-shots) to produce a more accurate result or capture windows that do not generate events on change (windows with accelerated 3d context) but this will notably increase the workload.

USAGE recordmydesktop [Options]^ filename

EXAMPLE

recordMyDesktop doesn't have a command line interface.
After startup, it can be controlled only through the following signals:

SIGUSR1 causes the program to pause if it's currently recording, and vice-versa.

SIGTERM causes normal termination of the recording.

SIGINT also causes normal termination.

SIGABRT terminates the program and removes the specified output file.

[76] SYSTEM SERVICES: HTTP

- apache2 restart
- apache2 start
- apache2 stop

apache2 restart

DESCRIPTION **Apache** is probably the most popular Linux-based Web server application in use. Once you have DNS correctly setup and your server has access to the Internet, you'll need to configure **Apache** to accept surfers wanting to access your Web site.

USAGE http://httpd.apache.org/docs/

EXAMPLE apache restart

apache2 start

DESCRIPTION **Apache** is probably the most popular Linux-based Web server application in use. Once you have DNS correctly setup and your server has access to the Internet, you'll need to configure **Apache** to accept surfers wanting to access your Web site.

USAGE http://httpd.apache.org/docs/

EXAMPLE apache start

apache2 stop

DESCRIPTION **Apache** is probably the most popular Linux-based Web server application in use. Once you have DNS correctly setup and your server has access to the Internet, you'll need to configure **Apache** to accept surfers wanting to access your Web site.

USAGE http://httpd.apache.org/docs/

EXAMPLE apache stop

[77] METASPLOIT

- community / pro start
- community / pro stop

community / pro start

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

Metasploit Pro, an open-core commercial **Metasploit** edition for penetration testers. **Metasploit Pro** includes all features of Metasploit Express and adds web application scanning and exploitation, social engineering campaigns, and VPN pivoting.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

community / pro stop

DESCRIPTION

The Metasploit Project is a computer security project which provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

Metasploit Community Edition, a free, web-based user interface for **Metasploit. Metasploit Community** is based on the commercial functionality of the paid-for editions with a reduced set of features, including network discovery, module browsing, and manual exploitation. **Metasploit Community** is included in the main installer.

USAGE n/a; GUI tool

EXAMPLE n/a; GUI tool

[78] MYSQL

- mysql restart
- mysql start
- mysql stop

mysql restart

DESCRIPTION **mysql** is a simple SQL shell with input line editing capabilities. It supports interactive and non-interactive use. When used interactively, query results are presented in an ASCII-table format. When used non-interactively (for example, as a filter), the result is presented in tab-separated format. The output format can be changed using command options.

USAGE mysql [options]

USAGE mysql --help

EXAMPLE mysql restart

mysql start

DESCRIPTION **mysql** is a simple SQL shell with input line editing capabilities. It supports interactive and non-interactive use. When used interactively, query results are presented in an ASCII-table format. When used non-interactively (for example, as a filter), the result is presented in tab-separated format. The output format can be changed using command options.

USAGE mysql [options]

USAGE mysql --help

EXAMPLE mysql start

mysql stop

DESCRIPTION **mysql** is a simple SQL shell with input line editing capabilities. It supports interactive and non-interactive use. When used interactively, query results are presented in an ASCII-table format. When used non-interactively (for example, as a filter), the result is presented in tab-separated format. The output format can be changed using command options.

USAGE mysql [options]

USAGE mysql --help

EXAMPLE mysql stop

[79] SSHD

- sshd restart
- sshd start
- sshd stop

sshd restart

DESCRIPTION **sshd** (OpenSSH Daemon) is the daemon program for ssh. Together these programs replace rlogin and rsh, and provide secure encrypted communications between two untrusted hosts over an insecure network. **sshd** listens for connections from clients. It is normally started at boot from */etc/rc*. It forks a new daemon for each incoming connection. The forked daemons handle key exchange, encryption, authentication, command execution, and data exchange. **sshd** can be configured using command-line options or a configuration file (by default sshd_config); command-line options override values specified in the configuration file. **sshd** rereads its configuration file when it receives a hangup signal, SIGHUP, by executing itself with the name and options it was started with, e.g. */usr/sbin/sshd*.

USAGE **sshd** [**-46DdeiqTt**] [**-b** *bits*] [**-C** *connection_spec*] [**-c** *host_certificate_file*] [**-E** *log_file*] [**-f** *config_file*] [**-g** *login_grace_time*] [**-h** *host_key_file*] [**-k** *key_gen_time*] [**-o** *option*] [**-p** *port*] [**-u** *len*]

OPTIONS http://www.openbsd.org/cgi-bin/man.cgi?query=sshd&sektion=8

EXAMPLE sshd restart

sshd start

DESCRIPTION **sshd** (OpenSSH Daemon) is the daemon program for ssh. Together these programs replace rlogin and rsh, and provide secure encrypted communications between two untrusted hosts over an insecure network. **sshd** listens for connections from clients. It is normally started at boot from */etc/rc*. It forks a new daemon for each incoming connection. The forked daemons handle key exchange, encryption, authentication, command execution, and data exchange. **sshd** can be configured using command-line options or a configuration file (by default sshd_config); command-line options override values specified in the configuration file. **sshd** rereads its configuration file when it receives a hangup signal, SIGHUP, by executing itself with the name and options it was started with, e.g. */usr/sbin/sshd*.

USAGE **sshd** [**-46DdeiqTt**] [**-b** *bits*] [**-C** *connection_spec*] [**-c** *host_certificate_file*] [**-E** *log_file*] [**-f** *config_file*] [**-g** *login_grace_time*] [**-h** *host_key_file*] [**-k** *key_gen_time*] [**-o** *option*] [**-p** *port*] [**-u** *len*]

OPTIONS http://www.openbsd.org/cgi-bin/man.cgi?query=sshd&sektion=8

EXAMPLE sshd start

sshd stop

DESCRIPTION **sshd** (OpenSSH Daemon) is the daemon program for ssh. Together these programs replace rlogin and rsh, and provide secure encrypted communications between two untrusted hosts over an insecure network. **sshd** listens for connections from clients. It is normally started at boot from */etc/rc*. It forks a new daemon for each incoming connection. The forked daemons handle key exchange, encryption, authentication, command execution, and data exchange. **sshd** can be configured using command-line options or a configuration file (by default sshd_config); command-line options override values specified in the configuration file. **sshd** rereads its configuration file when it receives a hangup signal, SIGHUP, by executing itself with the name and options it was started with, e.g. */usr/sbin/sshd*.

USAGE **sshd** [**-46DdeiqTt**] [**-b** *bits*] [**-C** *connection_spec*] [**-c** *host_certificate_file*] [**-E** *log_file*] [**-f** *config_file*] [**-g** *login_grace_time*] [**-h** *host_key_file*] [**-k** *key_gen_time*] [**-o** *option*] [**-p** *port*] [**-u** *len*]

OPTIONS http://www.openbsd.org/cgi-bin/man.cgi?query=sshd&sektion=8

EXAMPLE sshd stop

references

- http://www.aldeid.com
- http://www.morningstarsecurity.com
- http://www.hackingdna.com
- http://zer0byte.com/2013/03/19/kali-linux-complete-tools-list-installation-screen-shots/
- http://www.monkey.org/~dugsong/fragroute/
- http://www.sans.org/security-resources/idfaq/fragroute.php
- http://flylib.com/books/en/3.105.1.82/1/
- http://www.darknet.org.uk/2008/04/cdpsnarf-cdp-packet-sniffer/
- http://mateslab.weebly.com/dnmap-the-distributed-nmap.html
- http://www.tuicool.com/articles/raimMz
- http://backtrackwasneversoeasy.blogspot.co.uk/2012/02/terminating-internet-of-whole-network.html
- http://www.ethicalhacker.net
- http://nmap.org/ncat/guide/ncat-tricks.html
- http://nixgeneration.com/~jaime/netdiscover/
- http://csabyblog.blogspot.co.uk
- http://thehackernews.com
- https://code.google.com/p/wol-e/wiki/Help
- http://linux.die.net/man/1/xprobe2
- http://www.digininja.org/projects/twofi.php
- https://code.google.com/p/intrace/wiki/intrace
- https://github.com/iSECPartners/sslyze/wiki
- http://www.securitytube-tools.net/index.php@title=Braa.html
- http://security.radware.com

references

- http://www.kali.org/
- www.backtrack-linux.org
- http://www.question-defense.com
- http://www.vulnerabilityassessment.co.uk/torch.htm
- http://myexploit.wordpress.com/network-copy-router-config-pl-merge-router-config-pl/
- http://www.securitytube.net
- http://www.rutschle.net/tech/sslh.shtml
- http://althing.cs.dartmouth.edu/local/www.thoughtcrime.org/ie.html
- http://www.thoughtcrime.org/software/sslstrip/
- http://ucsniff.sourceforge.net/ace.html
- http://www.phenoelit.org/irpas/docu.html
- http://www.forensicswiki.org/wiki/Tcpflow
- http://linux.die.net/man/1/wireshark
- http://www.nta-monitor.com/tools-resources/security-tools/ike-scan
- http://www.vulnerabilityassessment.co.uk/cge.htm
- http://www.yersinia.net
- http://www.cqure.net/wp/tools/database/dbpwaudit/
- https://code.google.com/p/hexorbase/
- http://sqlmap.org/
- http://sqlsus.sourceforge.net/
- http://www.jammed.com/~jwa/hacks/security/tnscmd/tnscmd-doc.html
- http://mazzoo.de/blog/2006/08/25#ohrwurm
- http://securitytools.wikidot.com

references

- https://www.owasp.org
- http://www.powerfuzzer.com
- http://sipsak.org/
- http://resources.infosecinstitute.com/intro-to-fuzzing/
- http://www.rootkit.nl/files/lynis-documentation.html
- http://www.cirt.net/nikto2
- http://pentestmonkey.net/tools/audit/unix-privesc-check
- http://www.openvas.org
- http://blindelephant.sourceforge.net/
- code.google.com/p/plecost
- http://packetstormsecurity.com/files/94305/UA-Tester-User-Agent-Tester-1.03.html
- http://portswigger.net/burp/
- http://sourceforge.net/projects/websploit/
- http://www.edge-security.com/wfuzz.php
- https://code.google.com/p/wfuzz
- http://xsser.sourceforge.net/
- http://www.testingsecurity.com/paros_proxy
- http://www.parosproxy.org/
- http://www.edge-security.com/proxystrike.php
- http://www.hackingarticles.in
- http://tipstrickshack.blogspot.co.uk/2012/11/how-to-use-websploit.html
- http://cutycapt.sourceforge.net/
- http://dirb.sourceforge.net

references

- http://www.skullsecurity.org/
- http://deblaze-tool.appspot.com
- http://www.securitytube-tools.net/index.php@title=Grabber.html
- http://rgaucher.info/beta/grabber/
- http://howtohack.poly.edu/wiki/Padding_Oracle_Attack
- http://blog.gdssecurity.com/labs/2010/9/14/automated-padding-oracle-attacks-with-padbuster.html
- https://code.google.com/p/skipfish/
- http://w3af.org/
- http://wapiti.sourceforge.net/
- http://www.scrt.ch/en/attack/downloads/webshag
- http://www.hackingdna.com/2013/01/webshag-on-backtrack-5.html
- http://www.digininja.org/projects/cewl.php
- http://hashcat.net
- https://code.google.com/p/pyrit
- http://www.securiteam.com/tools/5JP0I2KFPA.html
- http://freecode.com/projects/chntpw
- http://whatisgon.wordpress.com/2010/01/28/chntpw-tutorial-resetting-windows-passwords-editing-registry-linux/
- http://www.cgsecurity.org/cmospwd.txt
- http://adaywithtape.blogspot.co.uk/2011/05/creating-wordlists-with-crunch-v30.html
- http://hashcat.net
- http://ixplizit.wordpress.com/2012/04/08/hashcat-the-very-basic/
- https://code.google.com/p/hash-identifier/
- http://www.osix.net/modules/article/?id=455

references

- http://cse.spsu.edu/raustin2/coursefiles/forensics/How_to_use_Volatility_v2.pdf
- http://thesprawl.org/projects/pack/#maskgen
- http://dev.man-online.org/man1/ophcrack-cli/
- http://ophcrack.sourceforge.net/
- http://manned.org
- http://www.onlinehashcrack.com/how_to_crack_windows_passwords.php
- http://project-rainbowcrack.com
- http://www.randomstorm.com/rsmangler-security-tool.php
- http://pentestn00b.wordpress.com
- http://bernardodamele.blogspot.co.uk/2011/12/dump-windows-password-hashes.html
- http://manpages.ubuntu.com/manpages/natty/man1/sipcrack.1.html
- http://www.leidecker.info/projects/sucrack.shtml
- http://santoshdudhade.blogspot.co.uk/2012/12/findmyhash-112-python-script-to-crack.html
- http://www.foofus.net/jmk/medusa/medusa.html#how
- http://www.irongeek.com/i.php?page=backtrack-r1-man-pages/medusa
- http://nmap.org/ncrack/man.html
- http://leidecker.info/projects/phrasendrescher.shtml
- http://wiki.thc.org/BlueMaho
- http://flylib.com/books/en/3.418.1.83/1/
- http://www.hackfromacave.com
- http://www.pentest.co.uk/downloads.html?cat=downloads&section=01_bluetooth
- https://github.com/rezeusor/killerbee
- https://code.google.com/p/nfc-tools/source/browse/trunk/mfoc/src/mfoc.c?r=977

references

- http://nfc-tools.org
- http://www.binarytides.com/hack-windows-social-engineering-toolkit-java-applet/
- http://seclists.org
- http://www.openbsd.org/cgi-bin/man.cgi?query=sshd&sektion=8
- http://recordmydesktop.sourceforge.net/manpage.php
- http://www.truecrypt.org
- http://keepnote.org
- http://apache.org
- https://github.com/simsong/AFFLIBv3
- http://www.computersecuritystudent.com/FORENSICS/VOLATILITY
- http://csabyblog.blogspot.co.uk/2013/01/backtrack-forensics-volafox.html
- http://www.sleuthkit.org/autopsy/desc.php
- http://sysforensics.org/2012/02/sleuth-kit-part-2-mmls-and-mmstat.html
- http://guymager.sourceforge.net/
- http://www.myfixlog.com/fix.php?fid=33
- http://www.gnu.org/software/ddrescue/manual/ddrescue_manual.html
- http://www.spenneberg.org/chkrootkit-mirror/faq/
- www.aircrack-ng.org/
- https://sites.google.com/site/clickdeathsquad/Home/cds-wpacrack
- http://www.willhackforsushi.com
- http://www.ciscopress.com
- http://openmaniak.com/kismet_platform.php
- http://sid.rstack.org/static/

references

- http://www.digininja.org
- http://thesprawl.org/projects/dnschef/
- http://hackingrelated.wordpress.com
- http://r00tsec.blogspot.co.uk/2011/07/hacking-with-evilgrade-on-backtrack5.html
- https://github.com/vecna/sniffjoke
- http://tcpreplay.synfin.net
- http://dallachiesa.com/code/rtpbreak/doc/rtpbreak_en.html
- http://tomeko.net/other/sipp/sipp_cheatsheet.php?lang=pl
- http://sipp.sourceforge.net/
- https://code.google.com/p/sipvicious/wiki/GettingStarted
- http://voiphopper.sourceforge.net/
- http://ohdae.github.io/Intersect-2.5/#Intro
- http://obscuresecurity.blogspot.co.uk/2013/03/powersploit-metasploit-shells.html
- http://dev.kryo.se/iodine/wiki/HowtoSetup
- http://proxychains.sourceforge.net/
- http://man.cx/ptunnel(8)
- http://www.sumitgupta.net/pwnat-example/
- https://github.com/
- http://www.dest-unreach.org/socat/doc/README
- https://bechtsoudis.com/webacoo/
- http://inundator.sourceforge.net/
- http://vinetto.sourceforge.net/
- http://www.elithecomputerguy.com/classes/hacking/