

## **Assignment 2**

Exploring tools available on Kali Linux  
for System Security Purpose

Atharva Vaidya  
121942024

Kali Linux is an open source distribution based on Debian focused on providing security auditing tools. Actively developed by Offensive Security, it's one of the most popular security distributions in use by infosec companies and ethical hackers. It includes numerous security-hacker tools for information gathering, vulnerability analysis, wireless attacks, web applications, exploitation tools, stress testing, forensic tools, sniffing and spoofing, password cracking, reverse engineering, hardware hacking and much more.

In this activity students are expected to practically learn following available tools on Kali Linux and prepare abstract report of all the tools and brief report (with screen shorts) of 15 tools (10 Red and 5 of their own choice and exclude this from brief report):

# Contents

NMAP . . . . .	3
Nessus . . . . .	6
Lynis . . . . .	11
John the Ripper . . . . .	17
Apktool . . . . .	21
Hydra . . . . .	25
RainbowCrack . . . . .	28
UnicornScan . . . . .	31
WPScan . . . . .	33
SlowHTTPTest . . . . .	36
Nikto . . . . .	38
Fluxion . . . . .	40
findmyhash . . . . .	41
Wireshark . . . . .	42
Metasploit Framework . . . . .	44
BurpSuite . . . . .	45

# NMAP

Nmap, short for Network Mapper, is a free, open-source tool for vulnerability scanning and network discovery. Network administrators use Nmap to identify what devices are running on their systems, discovering hosts that are available and the services they offer, finding open ports and detecting security risks. Nmap can be used to monitor single hosts as well as vast networks that encompass hundreds of thousands of devices and multitudes of subnets. While the basis of Nmap's functionality is port scanning, it allows for a variety of related capabilities including:

- **Network mapping** : Nmap can identify the devices on a network (also called host discovery), including servers, routers and switches, and how they're physically connected.
- **OS detection** : Nmap can detect the operating systems running on network devices (also called OS fingerprinting), providing the vendor name, the underlying operating system, the version of the software and even an estimate of devices' uptime.
- **Service discovery** : Nmap can not only identify hosts on the network, but whether they're acting as mail, web or name servers, and the particular applications and versions of the related software they're running.
- **Security auditing** : Figuring out what versions of operating systems and applications are running on network hosts lets network managers determine their vulnerability to specific flaws. If a network admin receives an alert about a vulnerability in a particular version of an application, for example, she can scan her network to identify whether that software version is running on the network and take steps to patch or update the relevant hosts. Scripts can also automate tasks such as detecting specific vulnerabilities.

```

root@kali:~# nmap -A -p- 10.1.120.65
Starting Nmap 7.25BETA1 ( https://nmap.org ) at 2019-09-03 06:48 UTC
Nmap scan report for 10.1.120.65
Host is up (0.00040s latency).
Not shown: 65521 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          OpenBSD ftppd 6.4 (Linux port 0.17)
22/tcp    open  ssh          OpenSSH 7.2p2 Ubuntu 4ubuntu2.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 e7:af:45:83:e4:aa:ed:a4:71:a4:68:07:a8:ed:67:9d (RSA)
|   256 38:34:cc:4c:06:54:1c:45:7f:10:la:2c:ca:14:64:19 (ECDSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
| smtp-commands: ac-computer-lab-pc, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
| ssl-cert: Subject: commonName=ubuntu
| Not valid before: 2016-07-23T08:49:16
| Not valid after:  2026-07-21T08:49:16
| ssl-date: TLS randomness does not represent time
80/tcp    open  http         Apache httpd 2.4.18 ((Ubuntu))
| http-server-header: Apache/2.4.18 (Ubuntu)
| http-title: Apache2 Ubuntu Default Page: It works
111/tcp   open  rpcbind     2-4 (RPC #100000)
| rpcinfo:
|   program version  port/proto service
|   100000  2,3,4      111/tcp  rpcbind
|   100000  2,3,4      111/udp rpcbind
|   100003  2,3,4      2049/tcp nfs
|   100003  2,3,4      2049/udp nfs
|   100005  1,2,3      52244/udp mountd
|   100005  1,2,3      59496/tcp mountd
|   100021  1,3,4      40328/tcp nlockmgr
|   100021  1,3,4      50851/udp nlockmgr
|   100227  2,3        2049/tcp nfs_acl
|   100227  2,3        2049/udp nfs_acl
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)

```

Figure 1: Scanning a PC on LAN.(1)

```

root@kali:~# nmap -A -p- 10.1.120.65
Starting Nmap 7.25BETA1 ( https://nmap.org ) at 2019-09-03 06:50 UTC
Nmap scan report for 10.1.120.65
Host is up (0.00040s latency).
Not shown: 65521 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          OpenBSD ftppd 6.4 (Linux port 0.17)
22/tcp    open  ssh          OpenSSH 7.2p2 Ubuntu 4ubuntu2.1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 e7:af:45:83:e4:aa:ed:a4:71:a4:68:07:a8:ed:67:9d (RSA)
|   256 38:34:cc:4c:06:54:1c:45:7f:10:la:2c:ca:14:64:19 (ECDSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
| smtp-commands: ac-computer-lab-pc, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
| ssl-cert: Subject: commonName=ubuntu
| Not valid before: 2016-07-23T08:49:16
| Not valid after:  2026-07-21T08:49:16
| ssl-date: TLS randomness does not represent time
80/tcp    open  http         Apache httpd 2.4.18 ((Ubuntu))
| http-server-header: Apache/2.4.18 (Ubuntu)
| http-title: Apache2 Ubuntu Default Page: It works
111/tcp   open  rpcbind     2-4 (RPC #100000)
| rpcinfo:
|   program version  port/proto service
|   100000  2,3,4      111/tcp  rpcbind
|   100000  2,3,4      111/udp rpcbind
|   100003  2,3,4      2049/tcp nfs
|   100003  2,3,4      2049/udp nfs
|   100005  1,2,3      52244/udp mountd
|   100005  1,2,3      59496/tcp mountd
|   100021  1,3,4      40328/tcp nlockmgr
|   100021  1,3,4      50851/udp nlockmgr
|   100227  2,3        2049/tcp nfs_acl
|   100227  2,3        2049/udp nfs_acl
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
2049/tcp  open  nfs acl    2-3 (RPC #100227)
8080/tcp  open  http        Jetty 9.2.14.v20151106
| http-open-proxy: Proxy might be redirecting requests
| http-server-header: Jetty(9.2.14.v20151106)
| http-title: Welcome to Jetty 9 on Debian
33814/tcp open  mountd     1-3 (RPC #100005)
40328/tcp open  nlockmgr   1-4 (RPC #100021)
59496/tcp open  mountd     1-3 (RPC #100005)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
```

Figure 2: Scanning a PC on LAN.(2)

```

Applications ▾ Places ▾ Terminal ▾ Tue 06:51
root@kali: ~

File Edit View Search Terminal Help

TCP/IP fingerprint:
OS:SCAN(V=7.25BETA1%e=4%D=9/3%0T=21%CT=1%CU=44494%PV=Y%DS=2%DC=T%G=Y%TM=5D6
OS:E0CCF%P=x86_64-pc-linux-gnu)SE0(SP=11%CD=FA00%ISR=9C%TI=1%CI=1%II=I%SS=
OS:%TS=U)OPS(01=M5B4%02=M5B4%03=M5B4%04=M5B4%05=M5B4%06=M5B4)WIN(WI=FFFF%W
OS:2=FFFF%W3=FFFF%W4=FFFF%W5=FFFF%W6=FFFF%W7=FFFF%W8=FFFF%W9=FFFF%W10=FFFF%W
OS:11=FFFF%W12=FFFF%W13=FFFF%W14=FFFF%W15=FFFF%W16=FFFF%W17=FFFF%W18=FFFF%W
OS:19=FFFF%W20=FFFF%W21=FFFF%W22=FFFF%W23=FFFF%W24=FFFF%W25=FFFF%W26=FFFF%W
OS:27=FFFF%W28=FFFF%W29=FFFF%W30=FFFF%W31=FFFF%W32=FFFF%W33=FFFF%W34=FFFF%W
OS:35=FFFF%W36=FFFF%W37=FFFF%W38=FFFF%W39=FFFF%W40=FFFF%W41=FFFF%W42=FFFF%W
OS:43=FFFF%W44=FFFF%W45=FFFF%W46=FFFF%W47=FFFF%W48=FFFF%W49=FFFF%W50=FFFF%W
OS:51=FFFF%W52=FFFF%W53=FFFF%W54=FFFF%W55=FFFF%W56=FFFF%W57=FFFF%W58=FFFF%W
OS:59=FFFF%W60=FFFF%W61=FFFF%W62=FFFF%W63=FFFF%W64=FFFF%W65=FFFF%W66=FFFF%W
OS:67=FFFF%W68=FFFF%W69=FFFF%W70=FFFF%W71=FFFF%W72=FFFF%W73=FFFF%W74=FFFF%W
OS:75=FFFF%W76=FFFF%W77=FFFF%W78=FFFF%W79=FFFF%W80=FFFF%W81=FFFF%W82=FFFF%W
OS:83=FFFF%W84=FFFF%W85=FFFF%W86=FFFF%W87=FFFF%W88=FFFF%W89=FFFF%W90=FFFF%W
OS:91=FFFF%W92=FFFF%W93=FFFF%W94=FFFF%W95=FFFF%W96=FFFF%W97=FFFF%W98=FFFF%W
OS:99=FFFF%W100=FFFF%W101=FFFF%W102=FFFF%W103=FFFF%W104=FFFF%W105=FFFF%W106=FFFF%W
OS:107=FFFF%W108=FFFF%W109=FFFF%W110=FFFF%W111=FFFF%W112=FFFF%W113=FFFF%W114=FFFF%W
OS:115=FFFF%W116=FFFF%W117=FFFF%W118=FFFF%W119=FFFF%W120=FFFF%W121=FFFF%W122=FFFF%W
OS:123=FFFF%W124=FFFF%W125=FFFF%W126=FFFF%W127=FFFF%W128=FFFF%W129=FFFF%W130=FFFF%W
OS:131=FFFF%W132=FFFF%W133=FFFF%W134=FFFF%W135=FFFF%W136=FFFF%W137=FFFF%W138=FFFF%W
OS:139=FFFF%W140=FFFF%W141=FFFF%W142=FFFF%W143=FFFF%W144=FFFF%W145=FFFF%W146=FFFF%W
OS:147=FFFF%W148=FFFF%W149=FFFF%W150=FFFF%W151=FFFF%W152=FFFF%W153=FFFF%W154=FFFF%W
OS:155=FFFF%W156=FFFF%W157=FFFF%W158=FFFF%W159=FFFF%W160=FFFF%W161=FFFF%W162=FFFF%W
OS:163=FFFF%W164=FFFF%W165=FFFF%W166=FFFF%W167=FFFF%W168=FFFF%W169=FFFF%W170=FFFF%W
OS:171=FFFF%W172=FFFF%W173=FFFF%W174=FFFF%W175=FFFF%W176=FFFF%W177=FFFF%W178=FFFF%W
OS:179=FFFF%W180=FFFF%W181=FFFF%W182=FFFF%W183=FFFF%W184=FFFF%W185=FFFF%W186=FFFF%W
OS:187=FFFF%W188=FFFF%W189=FFFF%W190=FFFF%W191=FFFF%W192=FFFF%W193=FFFF%W194=FFFF%W
OS:195=FFFF%W196=FFFF%W197=FFFF%W198=FFFF%W199=FFFF%W200=FFFF%W201=FFFF%W202=FFFF%W
OS:203=FFFF%W204=FFFF%W205=FFFF%W206=FFFF%W207=FFFF%W208=FFFF%W209=FFFF%W210=FFFF%W
OS:211=FFFF%W212=FFFF%W213=FFFF%W214=FFFF%W215=FFFF%W216=FFFF%W217=FFFF%W218=FFFF%W
OS:219=FFFF%W220=FFFF%W221=FFFF%W222=FFFF%W223=FFFF%W224=FFFF%W225=FFFF%W226=FFFF%W
OS:227=FFFF%W228=FFFF%W229=FFFF%W230=FFFF%W231=FFFF%W232=FFFF%W233=FFFF%W234=FFFF%W
OS:235=FFFF%W236=FFFF%W237=FFFF%W238=FFFF%W239=FFFF%W240=FFFF%W241=FFFF%W242=FFFF%W
OS:243=FFFF%W244=FFFF%W245=FFFF%W246=FFFF%W247=FFFF%W248=FFFF%W249=FFFF%W250=FFFF%W
OS:251=FFFF%W252=FFFF%W253=FFFF%W254=FFFF%W255=FFFF%W256=FFFF%W257=FFFF%W258=FFFF%W
OS:259=FFFF%W260=FFFF%W261=FFFF%W262=FFFF%W263=FFFF%W264=FFFF%W265=FFFF%W266=FFFF%W
OS:267=FFFF%W268=FFFF%W269=FFFF%W270=FFFF%W271=FFFF%W272=FFFF%W273=FFFF%W274=FFFF%W
OS:275=FFFF%W276=FFFF%W277=FFFF%W278=FFFF%W279=FFFF%W280=FFFF%W281=FFFF%W282=FFFF%W
OS:283=FFFF%W284=FFFF%W285=FFFF%W286=FFFF%W287=FFFF%W288=FFFF%W289=FFFF%W290=FFFF%W
OS:291=FFFF%W292=FFFF%W293=FFFF%W294=FFFF%W295=FFFF%W296=FFFF%W297=FFFF%W298=FFFF%W
OS:299=FFFF%W300=FFFF%W301=FFFF%W302=FFFF%W303=FFFF%W304=FFFF%W305=FFFF%W306=FFFF%W
OS:307=FFFF%W308=FFFF%W309=FFFF%W310=FFFF%W311=FFFF%W312=FFFF%W313=FFFF%W314=FFFF%W
OS:315=FFFF%W316=FFFF%W317=FFFF%W318=FFFF%W319=FFFF%W320=FFFF%W321=FFFF%W322=FFFF%W
OS:323=FFFF%W324=FFFF%W325=FFFF%W326=FFFF%W327=FFFF%W328=FFFF%W329=FFFF%W330=FFFF%W
OS:331=FFFF%W332=FFFF%W333=FFFF%W334=FFFF%W335=FFFF%W336=FFFF%W337=FFFF%W338=FFFF%W
OS:339=FFFF%W340=FFFF%W341=FFFF%W342=FFFF%W343=FFFF%W344=FFFF%W345=FFFF%W346=FFFF%W
OS:347=FFFF%W348=FFFF%W349=FFFF%W350=FFFF%W351=FFFF%W352=FFFF%W353=FFFF%W354=FFFF%W
OS:355=FFFF%W356=FFFF%W357=FFFF%W358=FFFF%W359=FFFF%W360=FFFF%W361=FFFF%W362=FFFF%W
OS:363=FFFF%W364=FFFF%W365=FFFF%W366=FFFF%W367=FFFF%W368=FFFF%W369=FFFF%W370=FFFF%W
OS:371=FFFF%W372=FFFF%W373=FFFF%W374=FFFF%W375=FFFF%W376=FFFF%W377=FFFF%W378=FFFF%W
OS:379=FFFF%W380=FFFF%W381=FFFF%W382=FFFF%W383=FFFF%W384=FFFF%W385=FFFF%W386=FFFF%W
OS:387=FFFF%W388=FFFF%W389=FFFF%W390=FFFF%W391=FFFF%W392=FFFF%W393=FFFF%W394=FFFF%W
OS:395=FFFF%W396=FFFF%W397=FFFF%W398=FFFF%W399=FFFF%W400=FFFF%W401=FFFF%W402=FFFF%W
OS:403=FFFF%W404=FFFF%W405=FFFF%W406=FFFF%W407=FFFF%W408=FFFF%W409=FFFF%W410=FFFF%W
OS:411=FFFF%W412=FFFF%W413=FFFF%W414=FFFF%W415=FFFF%W416=FFFF%W417=FFFF%W418=FFFF%W
OS:419=FFFF%W420=FFFF%W421=FFFF%W422=FFFF%W423=FFFF%W424=FFFF%W425=FFFF%W426=FFFF%W
OS:427=FFFF%W428=FFFF%W429=FFFF%W430=FFFF%W431=FFFF%W432=FFFF%W433=FFFF%W434=FFFF%W
OS:435=FFFF%W436=FFFF%W437=FFFF%W438=FFFF%W439=FFFF%W440=FFFF%W441=FFFF%W442=FFFF%W
OS:443=FFFF%W444=FFFF%W445=FFFF%W446=FFFF%W447=FFFF%W448=FFFF%W449=FFFF%W450=FFFF%W
OS:451=FFFF%W452=FFFF%W453=FFFF%W454=FFFF%W455=FFFF%W456=FFFF%W457=FFFF%W458=FFFF%W
OS:459=FFFF%W460=FFFF%W461=FFFF%W462=FFFF%W463=FFFF%W464=FFFF%W465=FFFF%W466=FFFF%W
OS:467=FFFF%W468=FFFF%W469=FFFF%W470=FFFF%W471=FFFF%W472=FFFF%W473=FFFF%W474=FFFF%W
OS:475=FFFF%W476=FFFF%W477=FFFF%W478=FFFF%W479=FFFF%W480=FFFF%W481=FFFF%W482=FFFF%W
OS:483=FFFF%W484=FFFF%W485=FFFF%W486=FFFF%W487=FFFF%W488=FFFF%W489=FFFF%W490=FFFF%W
OS:491=FFFF%W492=FFFF%W493=FFFF%W494=FFFF%W495=FFFF%W496=FFFF%W497=FFFF%W498=FFFF%W
OS:499=FFFF%W500=FFFF%W501=FFFF%W502=FFFF%W503=FFFF%W504=FFFF%W505=FFFF%W506=FFFF%W
OS:507=FFFF%W508=FFFF%W509=FFFF%W510=FFFF%W511=FFFF%W512=FFFF%W513=FFFF%W514=FFFF%W
OS:515=FFFF%W516=FFFF%W517=FFFF%W518=FFFF%W519=FFFF%W520=FFFF%W521=FFFF%W522=FFFF%W
OS:523=FFFF%W524=FFFF%W525=FFFF%W526=FFFF%W527=FFFF%W528=FFFF%W529=FFFF%W530=FFFF%W
OS:531=FFFF%W532=FFFF%W533=FFFF%W534=FFFF%W535=FFFF%W536=FFFF%W537=FFFF%W538=FFFF%W
OS:539=FFFF%W540=FFFF%W541=FFFF%W542=FFFF%W543=FFFF%W544=FFFF%W545=FFFF%W546=FFFF%W
OS:547=FFFF%W548=FFFF%W549=FFFF%W550=FFFF%W551=FFFF%W552=FFFF%W553=FFFF%W554=FFFF%W
OS:555=FFFF%W556=FFFF%W557=FFFF%W558=FFFF%W559=FFFF%W560=FFFF%W561=FFFF%W562=FFFF%W
OS:563=FFFF%W564=FFFF%W565=FFFF%W566=FFFF%W567=FFFF%W568=FFFF%W569=FFFF%W570=FFFF%W
OS:571=FFFF%W572=FFFF%W573=FFFF%W574=FFFF%W575=FFFF%W576=FFFF%W577=FFFF%W578=FFFF%W
OS:579=FFFF%W580=FFFF%W581=FFFF%W582=FFFF%W583=FFFF%W584=FFFF%W585=FFFF%W586=FFFF%W
OS:587=FFFF%W588=FFFF%W589=FFFF%W590=FFFF%W591=FFFF%W592=FFFF%W593=FFFF%W594=FFFF%W
OS:595=FFFF%W596=FFFF%W597=FFFF%W598=FFFF%W599=FFFF%W600=FFFF%W601=FFFF%W602=FFFF%W
OS:603=FFFF%W604=FFFF%W605=FFFF%W606=FFFF%W607=FFFF%W608=FFFF%W609=FFFF%W610=FFFF%W
OS:611=FFFF%W612=FFFF%W613=FFFF%W614=FFFF%W615=FFFF%W616=FFFF%W617=FFFF%W618=FFFF%W
OS:619=FFFF%W620=FFFF%W621=FFFF%W622=FFFF%W623=FFFF%W624=FFFF%W625=FFFF%W626=FFFF%W
OS:627=FFFF%W628=FFFF%W629=FFFF%W630=FFFF%W631=FFFF%W632=FFFF%W633=FFFF%W634=FFFF%W
OS:635=FFFF%W636=FFFF%W637=FFFF%W638=FFFF%W639=FFFF%W640=FFFF%W641=FFFF%W642=FFFF%W
OS:643=FFFF%W644=FFFF%W645=FFFF%W646=FFFF%W647=FFFF%W648=FFFF%W649=FFFF%W650=FFFF%W
OS:651=FFFF%W652=FFFF%W653=FFFF%W654=FFFF%W655=FFFF%W656=FFFF%W657=FFFF%W658=FFFF%W
OS:659=FFFF%W660=FFFF%W661=FFFF%W662=FFFF%W663=FFFF%W664=FFFF%W665=FFFF%W666=FFFF%W
OS:667=FFFF%W668=FFFF%W669=FFFF%W670=FFFF%W671=FFFF%W672=FFFF%W673=FFFF%W674=FFFF%W
OS:675=FFFF%W676=FFFF%W677=FFFF%W678=FFFF%W679=FFFF%W680=FFFF%W681=FFFF%W682=FFFF%W
OS:683=FFFF%W684=FFFF%W685=FFFF%W686=FFFF%W687=FFFF%W688=FFFF%W689=FFFF%W690=FFFF%W
OS:691=FFFF%W692=FFFF%W693=FFFF%W694=FFFF%W695=FFFF%W696=FFFF%W697=FFFF%W698=FFFF%W
OS:699=FFFF%W700=FFFF%W701=FFFF%W702=FFFF%W703=FFFF%W704=FFFF%W705=FFFF%W706=FFFF%W
OS:707=FFFF%W708=FFFF%W709=FFFF%W710=FFFF%W711=FFFF%W712=FFFF%W713=FFFF%W714=FFFF%W
OS:715=FFFF%W716=FFFF%W717=FFFF%W718=FFFF%W719=FFFF%W720=FFFF%W721=FFFF%W722=FFFF%W
OS:723=FFFF%W724=FFFF%W725=FFFF%W726=FFFF%W727=FFFF%W728=FFFF%W729=FFFF%W730=FFFF%W
OS:731=FFFF%W732=FFFF%W733=FFFF%W734=FFFF%W735=FFFF%W736=FFFF%W737=FFFF%W738=FFFF%W
OS:739=FFFF%W740=FFFF%W741=FFFF%W742=FFFF%W743=FFFF%W744=FFFF%W745=FFFF%W746=FFFF%W
OS:747=FFFF%W748=FFFF%W749=FFFF%W750=FFFF%W751=FFFF%W752=FFFF%W753=FFFF%W754=FFFF%W
OS:755=FFFF%W756=FFFF%W757=FFFF%W758=FFFF%W759=FFFF%W760=FFFF%W761=FFFF%W762=FFFF%W
OS:763=FFFF%W764=FFFF%W765=FFFF%W766=FFFF%W767=FFFF%W768=FFFF%W769=FFFF%W770=FFFF%W
OS:771=FFFF%W772=FFFF%W773=FFFF%W774=FFFF%W775=FFFF%W776=FFFF%W777=FFFF%W778=FFFF%W
OS:779=FFFF%W780=FFFF%W781=FFFF%W782=FFFF%W783=FFFF%W784=FFFF%W785=FFFF%W786=FFFF%W
OS:787=FFFF%W788=FFFF%W789=FFFF%W790=FFFF%W791=FFFF%W792=FFFF%W793=FFFF%W794=FFFF%W
OS:795=FFFF%W796=FFFF%W797=FFFF%W798=FFFF%W799=FFFF%W800=FFFF%W801=FFFF%W802=FFFF%W
OS:803=FFFF%W804=FFFF%W805=FFFF%W806=FFFF%W807=FFFF%W808=FFFF%W809=FFFF%W810=FFFF%W
OS:811=FFFF%W812=FFFF%W813=FFFF%W814=FFFF%W815=FFFF%W816=FFFF%W817=FFFF%W818=FFFF%W
OS:819=FFFF%W820=FFFF%W821=FFFF%W822=FFFF%W823=FFFF%W824=FFFF%W825=FFFF%W826=FFFF%W
OS:827=FFFF%W828=FFFF%W829=FFFF%W830=FFFF%W831=FFFF%W832=FFFF%W833=FFFF%W834=FFFF%W
OS:835=FFFF%W836=FFFF%W837=FFFF%W838=FFFF%W839=FFFF%W840=FFFF%W841=FFFF%W842=FFFF%W
OS:843=FFFF%W844=FFFF%W845=FFFF%W846=FFFF%W847=FFFF%W848=FFFF%W849=FFFF%W850=FFFF%W
OS:851=FFFF%W852=FFFF%W853=FFFF%W854=FFFF%W855=FFFF%W856=FFFF%W857=FFFF%W858=FFFF%W
OS:859=FFFF%W860=FFFF%W861=FFFF%W862=FFFF%W863=FFFF%W864=FFFF%W865=FFFF%W866=FFFF%W
OS:867=FFFF%W868=FFFF%W869=FFFF%W870=FFFF%W871=FFFF%W872=FFFF%W873=FFFF%W874=FFFF%W
OS:875=FFFF%W876=FFFF%W877=FFFF%W878=FFFF%W879=FFFF%W880=FFFF%W881=FFFF%W882=FFFF%W
OS:883=FFFF%W884=FFFF%W885=FFFF%W886=FFFF%W887=FFFF%W888=FFFF%W889=FFFF%W890=FFFF%W
OS:891=FFFF%W892=FFFF%W893=FFFF%W894=FFFF%W895=FFFF%W896=FFFF%W897=FFFF%W898=FFFF%W
OS:899=FFFF%W900=FFFF%W901=FFFF%W902=FFFF%W903=FFFF%W904=FFFF%W905=FFFF%W906=FFFF%W
OS:907=FFFF%W908=FFFF%W909=FFFF%W910=FFFF%W911=FFFF%W912=FFFF%W913=FFFF%W914=FFFF%W
OS:915=FFFF%W916=FFFF%W917=FFFF%W918=FFFF%W919=FFFF%W920=FFFF%W921=FFFF%W922=FFFF%W
OS:923=FFFF%W924=FFFF%W925=FFFF%W926=FFFF%W927=FFFF%W928=FFFF%W929=FFFF%W930=FFFF%W
OS:931=FFFF%W932=FFFF%W933=FFFF%W934=FFFF%W935=FFFF%W936=FFFF%W937=FFFF%W938=FFFF%W
OS:939=FFFF%W940=FFFF%W941=FFFF%W942=FFFF%W943=FFFF%W944=FFFF%W945=FFFF%W946=FFFF%W
OS:947=FFFF%W948=FFFF%W949=FFFF%W950=FFFF%W951=FFFF%W952=FFFF%W953=FFFF%W954=FFFF%W
OS:955=FFFF%W956=FFFF%W957=FFFF%W958=FFFF%W959=FFFF%W960=FFFF%W961=FFFF%W962=FFFF%W
OS:963=FFFF%W964=FFFF%W965=FFFF%W966=FFFF%W967=FFFF%W968=FFFF%W969=FFFF%W970=FFFF%W
OS:971=FFFF%W972=FFFF%W973=FFFF%W974=FFFF%W975=FFFF%W976=FFFF%W977=FFFF%W978=FFFF%W
OS:979=FFFF%W980=FFFF%W981=FFFF%W982=FFFF%W983=FFFF%W984=FFFF%W985=FFFF%W986=FFFF%W
OS:987=FFFF%W988=FFFF%W989=FFFF%W990=FFFF%W991=FFFF%W992=FFFF%W993=FFFF%W994=FFFF%W
OS:995=FFFF%W996=FFFF%W997=FFFF%W998=FFFF%W999=FFFF%W9999=FFFF%W
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 34.42 seconds

```

Figure 3: Scanning a PC on LAN.3

```

Applications ▾ Places ▾ Terminal ▾ Tue 06:53
root@kali: ~

File Edit View Search Terminal Help

root@kali:~# nmap -A -p- 10.1.121.255
Starting Nmap 7.25BETA1 ( https://nmap.org ) at 2019-09-03 06:53 UTC
Nmap scan report for 10.1.121.255
Host is up (0.00018s latency).
All 65535 scanned ports on 10.1.121.255 are filtered
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
1   0.14 ms  10.0.2.2
2   0.24 ms  10.1.121.255

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.74 seconds
root@kali:~#

```

Figure 4: Scanning a PC on Virtual Machine.

# Nessus

Nessus is a remote security scanning tool, which scans a computer and raises an alert if it discovers any vulnerabilities that malicious hackers could use to gain access to any computer you have connected to a network. It does this by running over 1200 checks on a given computer, testing to see if any of these attacks could be used to break into the computer or otherwise harm it.

Working Each computer has thousands of ports, all of which may or may not have services (ie: a server for a specific high-level protocol) listening on them. Nessus works by testing each port on a computer, determining what service it is running, and then testing this service to make sure there are no vulnerabilities in it that could be used by a hacker to carry out a malicious attack. Nessus is called a "remote scanner" because it does not need to be installed on a computer for it to test that computer. Instead, you can install it on only one computer and test as many computers as you would like.

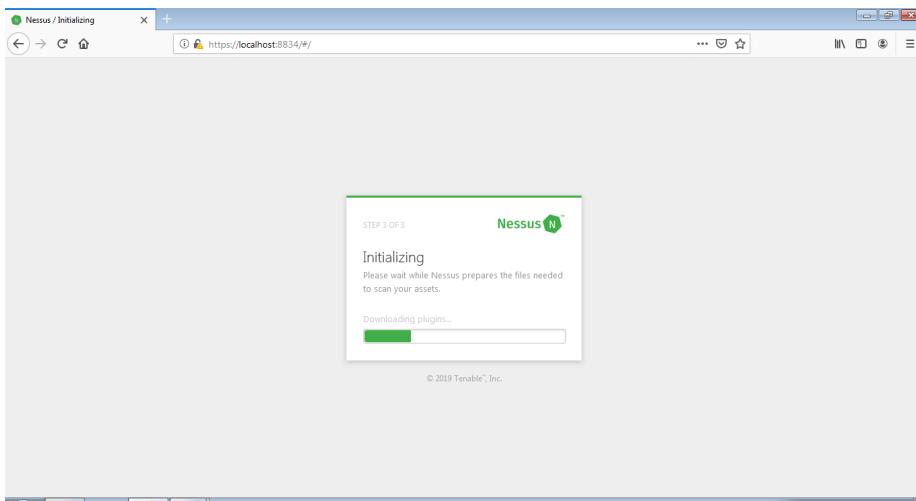


Figure 5: Nessus Initialization

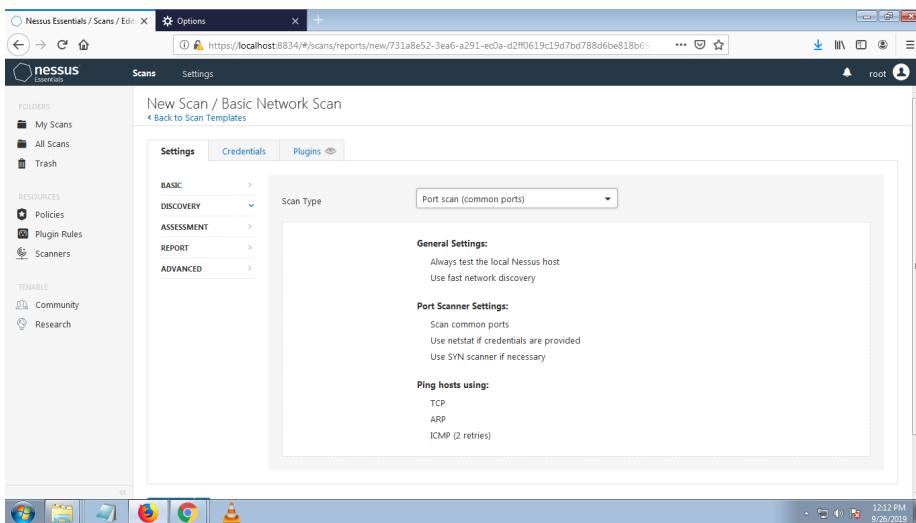


Figure 6: Settings Page

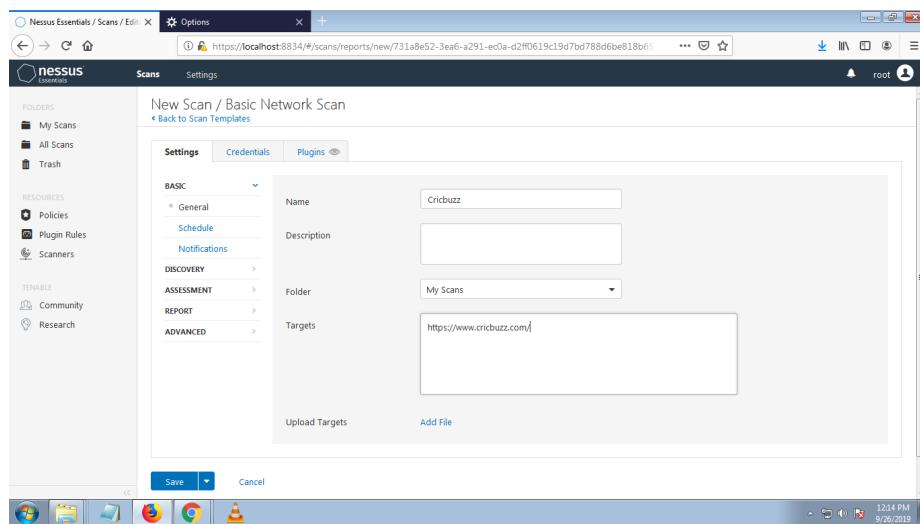


Figure 7: General Settings Page

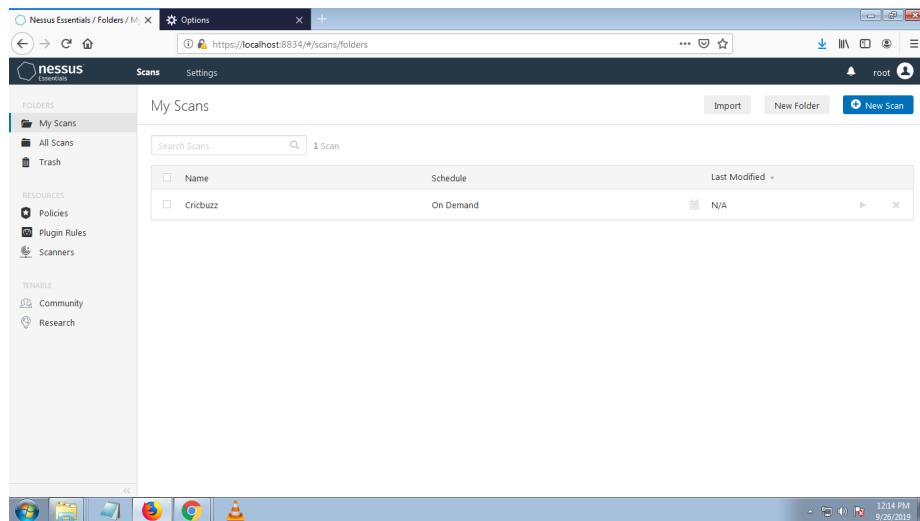


Figure 8: Scans List Page

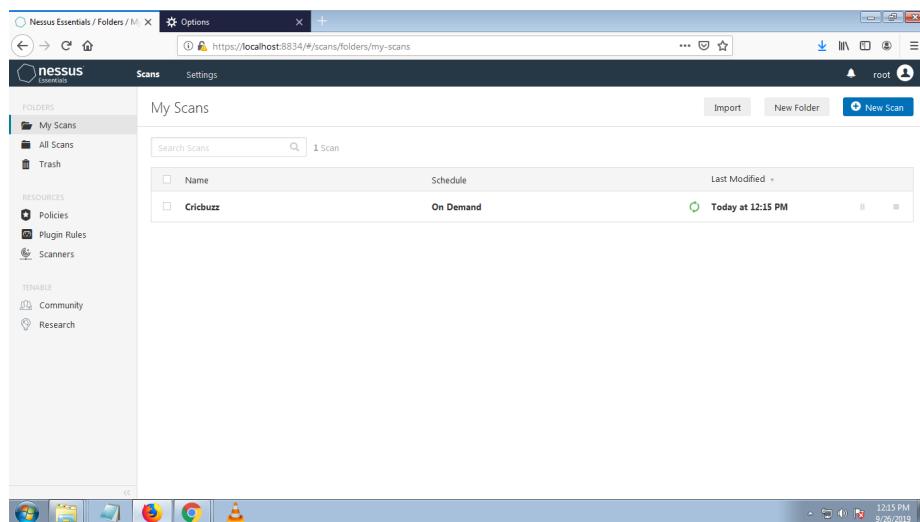


Figure 9: Scan Started

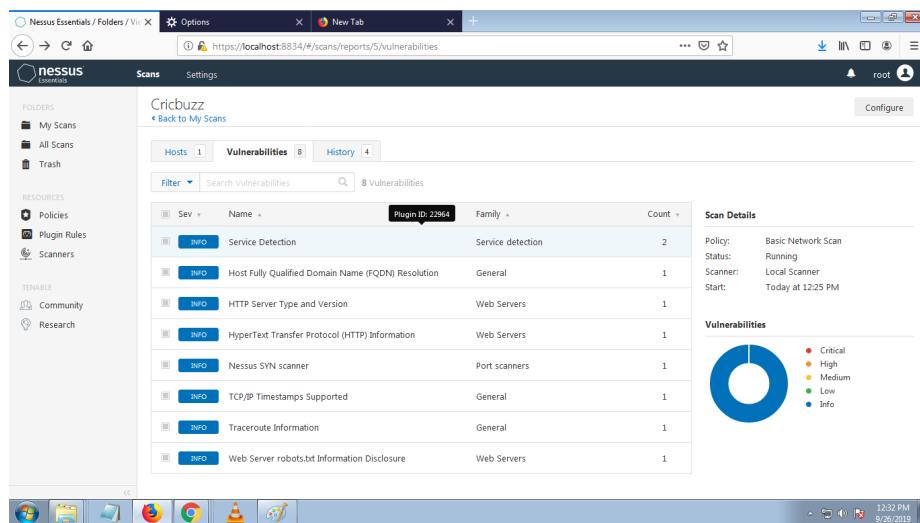


Figure 10: Vulnerabilities List

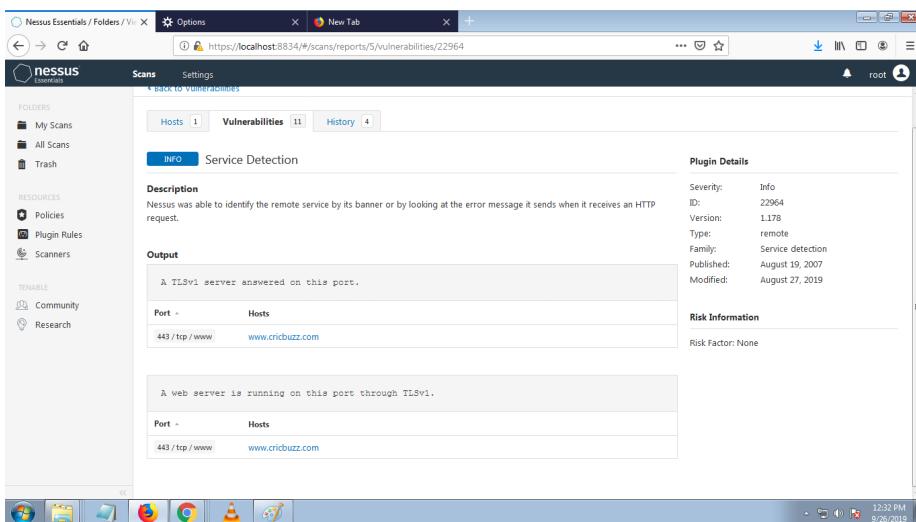


Figure 11: Service Detection Information

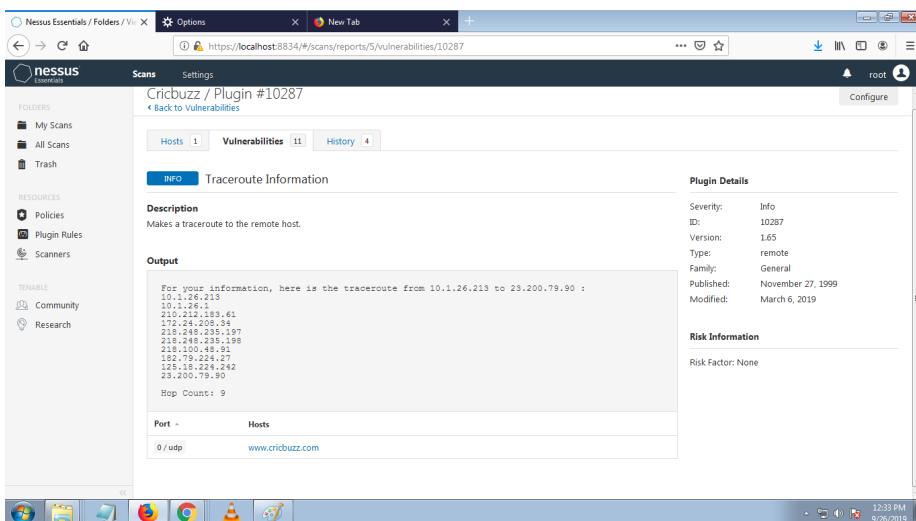


Figure 12: Traceroute Information

# Lynis

Lynis is a security tool for systems running Linux, macOS, or Unix-based operating system. It performs an extensive health scan of your systems to support system hardening and compliance testing.

Lynis scanning is modular. It will only use and test the components that it can find, such as the available system tools and its libraries. No installation of other tools is needed.

```

root@Kali:~# lynis
[ Lynis 2.6.2 ]
#####
Lynis comes with ABSOLUTELY NO WARRANTY. This is free software, and you are
welcome to redistribute it under the terms of the GNU General Public License.
See the LICENSE file for details about using this software.
2007-2018, CISOfy - https://cisoxy.com/lynis/
Enterprise support available (compliance, plugins, interface and tools)
#####

[+] Initializing program

Usage: lynis command [options]

Command:

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

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

  update     : Show update details

Options:

  --no-log          : Don't create a log file
  --pentest+         : Non-privileged scan (useful for pentest)
  --profile <profile> : Scan the system with the given profile file
  --quick (-Q)       : Quick mode, don't wait for user input

root@Kali:~# 

```

Figure 13: Lynis

```

root@Kali:~# lynis audit system --no-log --quick
[ Lynis 2.6.2 ]
#####
Lynis comes with ABSOLUTELY NO WARRANTY. This is free software, and you are
welcome to redistribute it under the terms of the GNU General Public License.
See the LICENSE file for details about using this software.
2007-2018, CISOfy - https://cisoxy.com/lynis/
Enterprise support available (compliance, plugins, interface and tools)
#####

[+] Initializing program
[ DONE ] [ DONE ]
- Detecting OS...
- Checking profiles...
[ DONE ]
Program version: 2.6.2
Operating system: Linux
Operating system name: Debian
Operating system version: kali-rolling
Kernel version: 4.19
Machine platform: x86_64
Hostname: hack3RBOX
Profiles: /etc/lynis/default.prf
Log file: /dev/null
Report file: /var/log/lynis-report.dat
Report version: 1.0
Plugin directory: //usr/share/lynis/plugins
Auditors: [Not Specified]
Test profile: all
Test category: all
Test group: all
[ UNKNOWN ]
- Program update status... [ UNKNOWN ]
[ Lynis update available

root@Kali:~# 

```

Figure 14: Lynis System Audit - OS Detection

```

root : bash — Konsole
File Edit View Bookmarks Settings Help
[+] Plugins (phase 1)
-----
Note: plugins have more extensive tests and may take several minutes to complete
- Plugin: debian
  [
  [+] Debian Tests
  -----
  - Checking for system binaries that are required by Debian Tests...
    - Checking /bin... [ FOUND ]
    - Checking /etc... [ FOUND ]
    - Checking /usr/bin... [ FOUND ]
    - Checking /usr/lib... [ FOUND ]
    - Checking /usr/local/bin... [ FOUND ]
    - Checking /usr/local/lib... [ FOUND ]
  - Authentication
    - PAM (Pluggable Authentication Modules):
      - libpam-dbd... [ Not Installed ]
      - libpam-nsu... [ Not Installed ]
  - File System Checks:
    - DM-Crypt, Cryptsetup & Cryptmount:
      - Crypting / on /dev/sda1 [ NOT ENCRYPTED ]
  - Software:
    - apt-listbugs
    - apt-listchanges
    - checkroot
    - needrestart
    - debsecan
    - debsums
    - fail2ban
  ]
[+] Boot and services
-----
- Service Manager [ systemd ]
- Checking presence GRUB [ DISABLED ]
- Checking for password protection [ FOUND ]
- Check running services (systemctl) [ WARNING ]
- Check startup services (systemctl)
  - Result: found 25 enabled services [ DONE ]
- Check enabled services at boot (systemctl)
  - Result: found 25 enabled services [ OK ]
- Check startup files (permissions)
- Checking sublogin in rescue.service [ NOT FOUND ]

```

Figure 15: Lynis System Audit - Debian Tests and Boot Services

```

root : bash — Konsole
File Edit View Bookmarks Settings Help
[+] Kernel
-----
- Checking default run level [ RUNLEVEL 5 ]
- Checking CPU supports MMX/PES [ FOUND ]
- Checking if PAE and noexec/execute supported [ FOUND ]
- Checking kernel version and release [ DONE ]
- Checking kernel type [ DONE ]
- Checking for 65 active modules [ DONE ]
- Checking Linux kernel configuration file [ FOUND ]
- Checking default I/O kernel scheduler [ FOUND ]
- Checking if /proc/sys/vm/overcommit_update [ OK ]
- Checking core dumps configuration [ DISABLED ]
- Checking if setuid core dumps configuration [ DEFAULT ]
- Check if reboot is needed [ NO ]
[+] Memory and Processes
-----
- Searching /proc/meminfo [ FOUND ]
- Searching for dead/ombie processes [ OK ]
- Searching for IO waiting processes [ OK ]
[+] Users, Groups and Authentication
-----
- Administrator accounts [ OK ]
- Unique group names [ OK ]
- Consistency of group files (grpck) [ OK ]
- Unique group IDs [ OK ]
- Password consistency [ OK ]
- Query system users (non daemons) [ DONE ]
- NIS+ authentication support [ NOT ENABLED ]
- NIS authentication support [ NOT ENABLED ]
- PAM modules [ FOUND ]
- PAM configuration file [ OK ]
- Check suders file permissions [ SUGGESTION ]
- PAM password strength tools [ FOUND ]
- PAM configuration files (pam.conf) [ FOUND ]
- PAM configuration files (pam.d) [ FOUND ]
- PAM modules [ NOT FOUND ]
- Accounts without expire date [ OK ]
- Accounts without password [ OK ]
- Checking user password aging (minimum) [ DISABLED ]
- User password aging (maximum) [ DISABLED ]
- Checking expired passwords [ OK ]

```

Figure 16: Lynis System Audit - Kernel, Memory and Processes, Users and Group

```

root:bash - Konsole
File Edit View Bookmarks Settings Help
[+] Shells
- Checking shells from /etc/shells
  - Recommanded shells (valid shells: 11).
  - Session Timeout settings [ NONE ]
- Checking default umask values [ NONE ]
- Checking default umask in /etc/bash.bashrc [ NONE ]
- Checking default umask in /etc/profile [ NONE ]
[+] File systems
-----[ SUGGESTION ]-----[ SUGGESTION ]-----[ OK ]-----[ OK ]-----[ SUGGESTION ]-----[ NOT DISABLED ]-----[ FOUND ]-----[ FOUND ]
- Checking /home mount point
- Checking /tmp mount point
- Checking /var mount point
- Quotas (check /etc/fstab)
- Testing swap partitions
- Testing /proc mount (inidepid)
- Checking /tmp quota (/tmp)
- Checking /var/tmp sticky bit
- Checking /var/tmp sticky bit
  ACL support root file system [ NOT FOUND ]
  - Checking /var/tmp quota [ NOT FOUND ]
- Checking Locate database
- Disable kernel support of some filesystems
  - discovered kernel modules: freevxf5 hfs hfsplus jffs2 squashfs udf
[+] USB Devices
-----[ SUGGESTION ]-----[ SUGGESTION ]-----[ OK ]-----[ OK ]-----[ SUGGESTION ]-----[ NOT FOUND ]
- Checking storage driver (modprobe config) [ NOT DISABLED ]
- Checking USB devices authorization [ ENABLED ]
- Checking USBDguard [ NOT FOUND ]
[+] Storage
-----[ NOT DISABLED ]
[+] NFS
-----[ NOT FOUND ]
[+] Name services
-----[ FOUND ]
- Searching DNS domain name [ FOUND ]

```

Figure 17: Lynis System Audit - Shell, Filesystems and Devices

```

root:bash - Konsole
File Edit View Bookmarks Settings Help
[+] Software: webserver
-----[ FOUND ]-----[ FOUND ]-----[ FOUND (117) ]-----[ NOT FOUND ]-----[ NOT FOUND ]-----[ NOT FOUND ]-----[ NOT FOUND ]
- Checking Apache (binary) /usr/sbin/apache2
  Info: Configuration file found /etc/apache2/apache2.conf
  Info: No virtual hosts found
  + Loading configuration
    Found 117 loadable modules
      mod_evasive: anti-Dos/brute force [ NOT FOUND ]
      mod_rewrite/mod_dos [ FOUND ]
      ModSecurity: web application firewall [ NOT FOUND ]
- Checking nginx
[+] SSH Support
-----[ NOT FOUND ]
- Checking running SSH daemon [ NOT FOUND ]
[+] SNMP Support
-----[ NOT FOUND ]
[+] Databases
-----No database engines found
[+] LDAP Services
-----[ NOT FOUND ]
- Checking OpenLDAP instance [ NOT FOUND ]
[+] PHP
-----[ NOT FOUND ]
- Checking PHP [ NOT FOUND ]
[+] Squid Support
-----[ NOT FOUND ]
- Checking running Squid daemon [ NOT FOUND ]
[+] Logging and files
-----[ OK ]-----[ NOT FOUND ]-----[ FOUND ]-----[ NOT FOUND ]-----[ FOUND ]-----[ NOT FOUND ]
- Checking for a running log daemon
- Checking Syslog-NG status
- Checking system journal status
- Checking logrotate status
- Checking Syslog status
- Checking RFC 3195 daemon status [ NOT FOUND ]

```

Figure 18: Lynis System Audit - SSH, LDAP, Logging

```

root:bash - Konsole
File Edit View Bookmarks Settings Help
[+] Insecure services
-----  

- Checking inetd status [ NOT ACTIVE ]
[+] Banners and identification
-----  

- /etc/issue [ FOUND ]  

- /etc/issue contents [ WEAK ]  

- /etc/issue.net [ FOUND ]  

- /etc/issue.net contents [ WEAK ]
[+] Scheduled tasks
-----  

- Checking crontab/cronjob [ DONE ]
[+] Accounting
-----  

- Checking accounting information [ NOT FOUND ]  

- Checking sysstat accounting data [ DISABLED ]  

- Checking auditd [ NOT FOUND ]
[+] Time and synchronization
-----  

[+] Cryptography
-----  

- Checking for expired SSL certificates [ 0/2 ] [ NONE ]
[+] Virtualization
-----  

[+] Containers
-----  

[+] Security frameworks
-----  

- Checking presence AppArmor [ FOUND ]  

- Checking AppArmor configuration [ FOUND ]  

- Checking presence SELinux [ NOT FOUND ]  

- Checking presence grsecurity [ NOT FOUND ]  

- Checking for implemented MAC framework [ NONE ]
[+] Software: file integrity
-----  

- Checking file integrity tools [ FOUND ]

```

Figure 19: Lynis System Audit - Security Frameworks

```

root:bash - Konsole
File Edit View Bookmarks Settings Help
- Checking presence integrity tool [ NOT FOUND ]
[+] Software: System tooling
-----  

- Checking automation tooling [ NOT FOUND ]  

- Automation tooling [ NONE ]
[+] Software: Malware
-----  

- Checking chrootkit [ FOUND ]
[+] File Permissions
-----  

- Starting file permissions check
[+] Home directories
-----  

- Checking shell history files [ OK ]
[+] Kernel Hardening
-----  

- Comparing sysctl key pairs with scan profile [ OK ]  

- fs.protected_hardlinks (exp: 1) [ OK ]  

- fs.protected_symlinks (exp: 1) [ OK ]  

- fscrypt_dismount (exp: 0) [ OK ]  

- kernel.core_uses_pid (exp: 1) [ DIFFERENT ]  

- kernel.ctrl-alt-del (exp: 0) [ OK ]  

- kernel.kptr_restrict (exp: 1) [ OK ]  

- kernel.kptr_restrict (exp: 2) [ DIFFERENT ]  

- kernel.randomize_va_space (exp: 2) [ OK ]  

- kernel.sysrq (exp: 1) [ DIFFERENT ]  

- kernel.utsname_scope (exp: 1 2 3) [ DIFFERENT ]  

- net.ipv4.conf.all.accept_redirects (exp: 0) [ DIFFERENT ]  

- net.ipv4.conf.all.accept_source_route (exp: 0) [ OK ]  

- net.ipv4.conf.all.forwarding (exp: 0) [ OK ]  

- net.ipv4.conf.all.log_martians (exp: 0) [ DIFFERENT ]  

- net.ipv4.conf.all.mc_forwarding (exp: 0) [ OK ]  

- net.ipv4.conf.all.secure_redirects (exp: 1) [ OK ]  

- net.ipv4.conf.all.rp_filter (exp: 1) [ DIFFERENT ]  

- net.ipv4.conf.all.send_redirects (exp: 0) [ DIFFERENT ]  

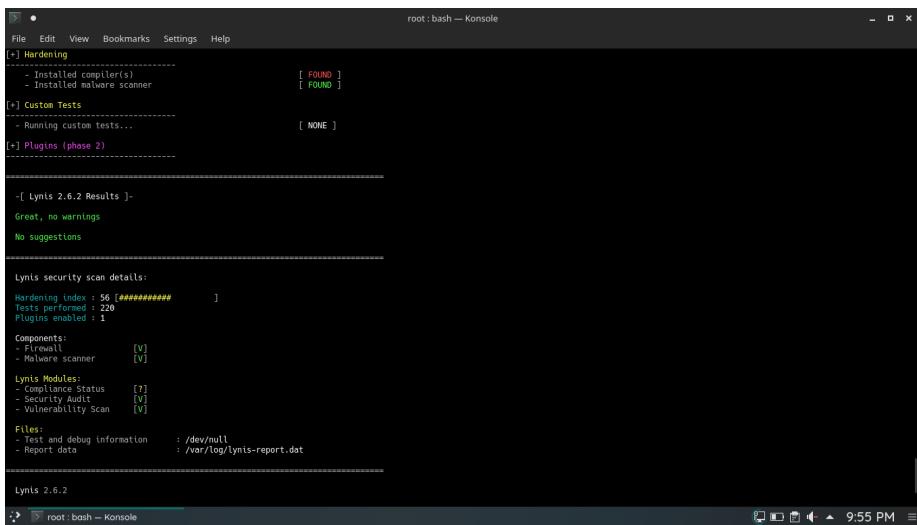
- net.ipv4.conf.all.accept_redirects (exp: 0) [ DIFFERENT ]  

- net.ipv4.conf.default.accept_source_route (exp: 0) [ DIFFERENT ]  

- net.ipv4.conf.default.log_martians (exp: 1) [ DIFFERENT ]

```

Figure 20: Lynis System Audit - Kernel Hardening



The screenshot shows a terminal window titled "root : bash — Konsole". The window displays the output of the Lynis security audit tool. The output is organized into several sections:

- [+] Hardening**:
  - Installed compiler(s) [ FOUND ]
  - Installed malware scanner [ FOUND ]
- [+] Custom Tests**:
  - Running custom tests... [ NONE ]
- [+] Plugins (phase 2)**:
  - [ Lynis 2.6.2 Results ]:
    - Greet, no warnings
    - No suggestions
- Lynis security scan details:**
  - Hardening Index : 56 [##### ]
  - Tests performed : 220
  - Plugins enabled : 1
- Components:**
  - Malware scanner [V]
- Lynis Modules:**
  - Compliance Status [?]
  - Security Audit [V]
  - Vulnerability Scan [V]
- Files:**
  - Test and debug information : /dev/null
  - Report data : /var/log/lynis-report.dat

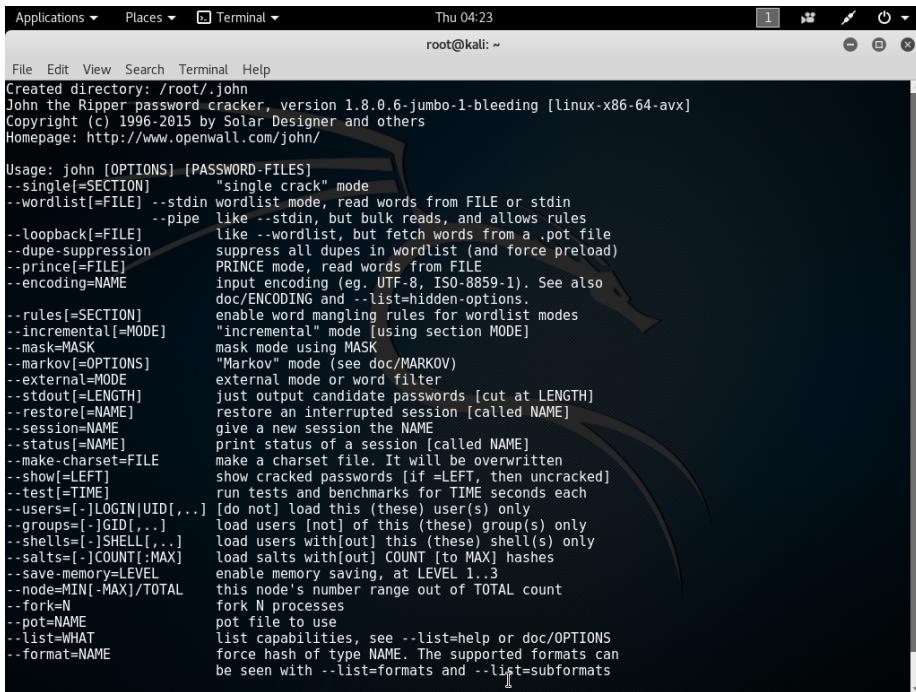
The bottom of the terminal window shows the status bar with "root : bash — Konsole" and the time "9:55 PM".

Figure 21: Lynis System Audit - Suggestions

# John the Ripper

John the Ripper is a fast password cracker. It is an old but a very good password cracker that uses wordlists to crack given hash. This software is available in two versions such as paid version and free version. It is totally cross-platform. John the Ripper works in 3 distinct modes to crack the passwords:

- **Single Crack Mode** : In this mode John the ripper makes use of the information available to it in the form of a username and other information.
- **Wordlist Crack Mode** : In this mode John the ripper uses a wordlist that can also be called a Dictionary and it compares the hashes of the words present in the Dictionary with the password hash. We can use any desired wordlist.
- **Incremental Mode** : This is the most powerful cracking mode, it can try all possible character combinations as passwords. However, it is assumed that cracking with this mode will never terminate because of the number of combinations being too large (actually, it will terminate if you set a low password length limit or make it use a small charset), and you'll have to interrupt it earlier.

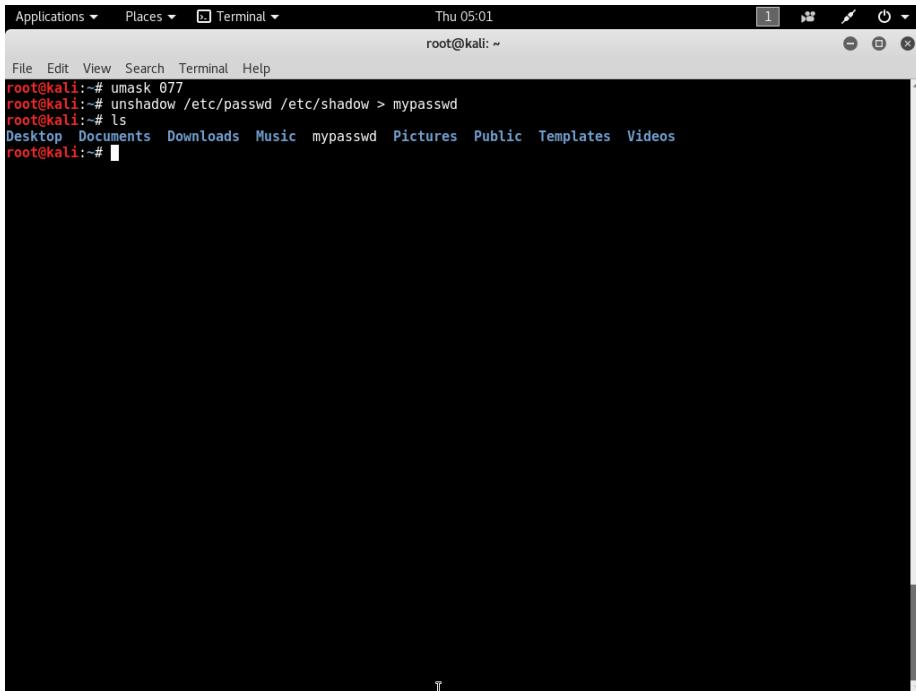


Applications ▾ Places ▾ Terminal ▾ Thu 04:23  
root@kali: ~

```
File Edit View Search Terminal Help
Created directory: /root/.john
John the Ripper password cracker, version 1.8.0.6-jumbo-1-bleeding [linux-x86-64-avx]
Copyright (c) 1996-2015 by Solar Designer and others
Homepage: http://www.openwall.com/john

Usage: john [OPTIONS] [PASSWORD-FILES]
--single[=SECTION]           "single crack" mode
--wordlist[=FILE]             --stdin wordlist mode, read words from FILE or stdin
--loopback[=FILE]              --pipe like --stdin, but bulk reads, and allows rules
--dupe-suppression            like --wordlist, but fetch words from a .pot file
--prince[=FILE]                suppress all dupes in wordlist (and force preload)
--encoding=NAME                 PRINCE mode, read words from FILE
--rules[=SECTION]               input encoding (eg. UTF-8, ISO-8859-1). See also
--incremental[=MODE]             doc/ENCODING and --list=hidden-options.
--mask=MASK                     enable word mangling rules for wordlist modes
--markov[=OPTIONS]              "incremental" mode [using section MODE]
--external[=MODE]                mask mode using MASK
--stdout[=LENGTH]                "Markov" mode (see doc/MARKOV)
--restore[=NAME]                  external mode or word filter
--session=NAME                   just output candidate passwords [cut at LENGTH]
--status[=NAME]                   restore an interrupted session [called NAME]
--make-charset=FILE              give a new session the NAME
--show[=LEFT]                     print status of a session [called NAME]
--test[=TIME]                      make a charset file. It will be overwritten
--users=[-]LOGIN|UID[,...]        show cracked passwords [if =LEFT, then uncracked]
--groups=[-]GID[,...]              run tests and benchmarks for TIME seconds each
--shells=[-]SHELL[,...]            [do not] load this (these) user(s) only
--salts=[-]COUNT[:MAX]            load users [not] of this (these) group(s) only
--save-memory=LEVEL               load salts without COUNT [to MAX] hashes
--node=MIN[!-]MAX/[TOTAL]          enable memory saving, at LEVEL 1..3
--fork=N                          this node's number range out of TOTAL count
--pot=NAME                         fork N processes
--list=WAT                         pot file to use
--format=NAME                      list capabilities, see --list=help or doc/OPTIONS
--format=NAME                      force hash of type NAME. The supported formats can
                                   be seen with --list=formats and --list=subformats
```

Figure 22: John The Ripper



Applications ▾ Places ▾ Terminal ▾ Thu 05:01  
root@kali: ~

```
File Edit View Search Terminal Help
root@kali:~# umask 077
root@kali:~# unshadow /etc/passwd /etc/shadow > mypasswd
root@kali:~# ls
Desktop Documents Downloads Music mypasswd Pictures Public Templates Videos
root@kali:~#
```

Figure 23: Unshadowing the shadow file and copying to PWD

A screenshot of a Kali Linux terminal window titled "Terminal". The window shows the command "john mypasswd -show" being run, which displays a single cracked password hash: "1 password hash cracked, 0 left". The terminal window has a standard Gnome-style interface with a title bar, menu bar, and scroll bars.

```
root@kali:~# # --show is used to Display all cracked passwords
root@kali:~# john mypasswd -show
root:toor:0:0:root:/bin/bash

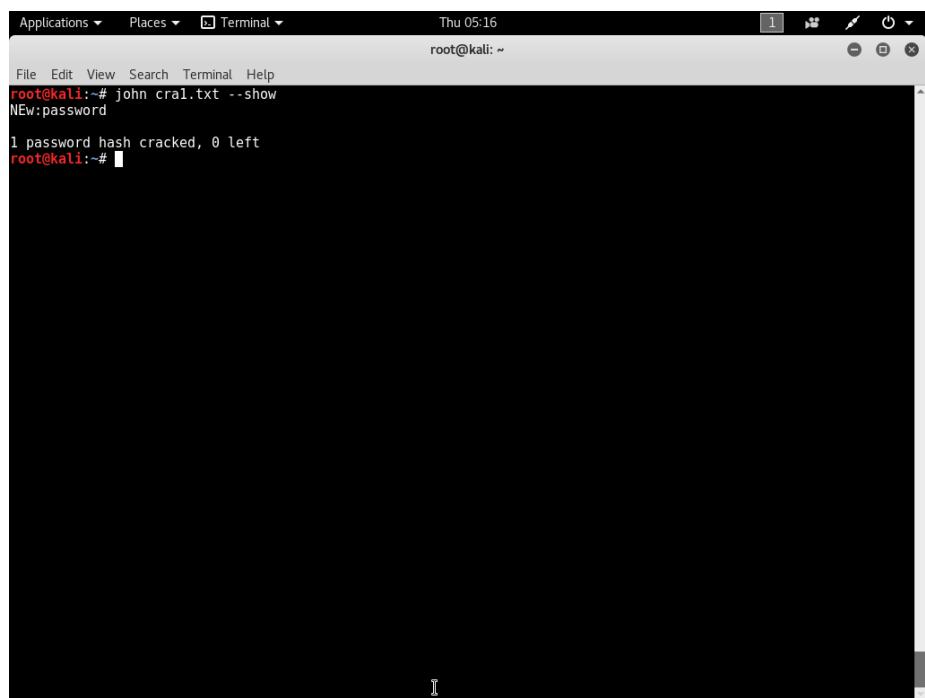
1 password hash cracked, 0 left
root@kali:~#
```

Figure 24: Cracking the Passwords of users

A screenshot of a Kali Linux terminal window titled "Terminal". The window shows the command "nano craf.txt" being run, which displays a single line of text: "NEw:\$2a\$05\$bvIG6Nmld91Mu9RcmWZf05HJIMCT8riNW0hEp8f6/FuA2/mHZFpe". The terminal window has a standard Gnome-style interface with a title bar, menu bar, and scroll bars. A nano editor status bar is visible at the bottom.

```
root@kali:~# File Edit View Search Terminal Help
root@kali:~# nano 2.6.3          File: craf.txt
NEw:$2a$05$bvIG6Nmld91Mu9RcmWZf05HJIMCT8riNW0hEp8f6/FuA2/mHZFpe
```

Figure 25: Contents of a Text File containing Hash



A screenshot of a Kali Linux terminal window. The title bar shows "Applications ▾", "Places ▾", "Terminal ▾", "Thu 05:16", and "root@kali: ~". The terminal window contains the following text:

```
File Edit View Search Terminal Help
root@kali:~# john cral.txt --show
NEmw:password
1 password hash cracked, 0 left
root@kali:~#
```

Figure 26: Cracking the Hash of the Text file

# Apktool

Apktool is a tool for reverse engineering 3rd party, closed, binary Android apps. It can decode resources to nearly original form and rebuild them after making some modifications. It also makes working with an app easier because of the project like file structure and automation of some repetitive tasks like building apk, etc. Decompiling an APK enables to view the content inside an apk file and various aspects of an app or game can be modified. APKTool is used to port applications to previously unsupported devices, theme some of your favorite applications, look into the strings of APK files to see what may be coming in future updates, and provide translations for applications.

## Decode APK Resources

In the illustration below, I am using vulnerable APK file (**diva-beta.apk**) Decoding can be done using the command:

```
apktool d diva-beta.apk
```

The **res** folder contains all the media files including images, ringtones and many more. It also contains the front-end XML files which can be used to alter the look and feel of the app you have decompiled.

**Original** folder contains META-INF and original AndroidManifest.xml, META-INF has APK signatures. After modifying the APK, it is no longer signed.

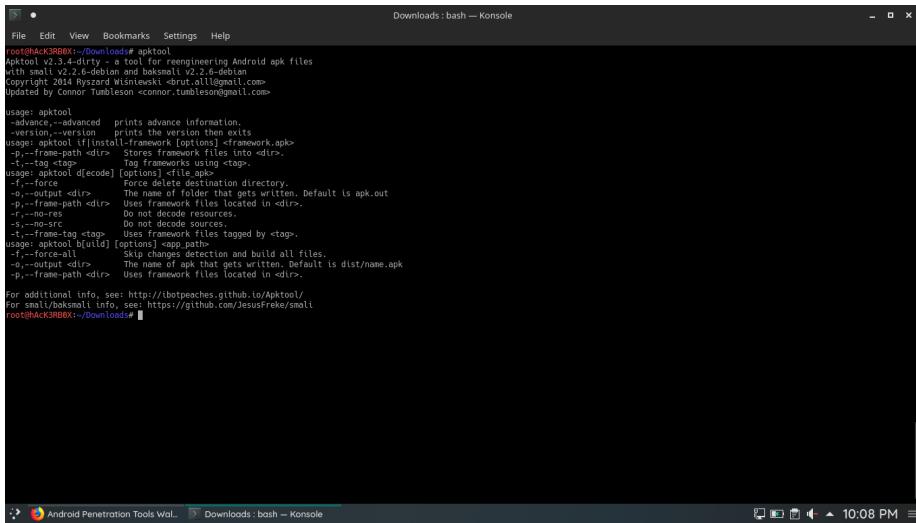
The **AndroidManifest.xml** file describes essential information about your app to the Android build tools, the Android operating system, and Google Play.

The **Smali** folder is the important part, Apktool decodes compiled java files in the form of .smali files (Read more on smali here). These Smali files are responsible for the functionality of the app and handling them would create direct impacts to the app.

## Rebuilding decoded resources back to binary APK

Once done with the modification to the APK source, the APK file can be rebuilt from the source files using the command:

```
apktool b diva-beta
```

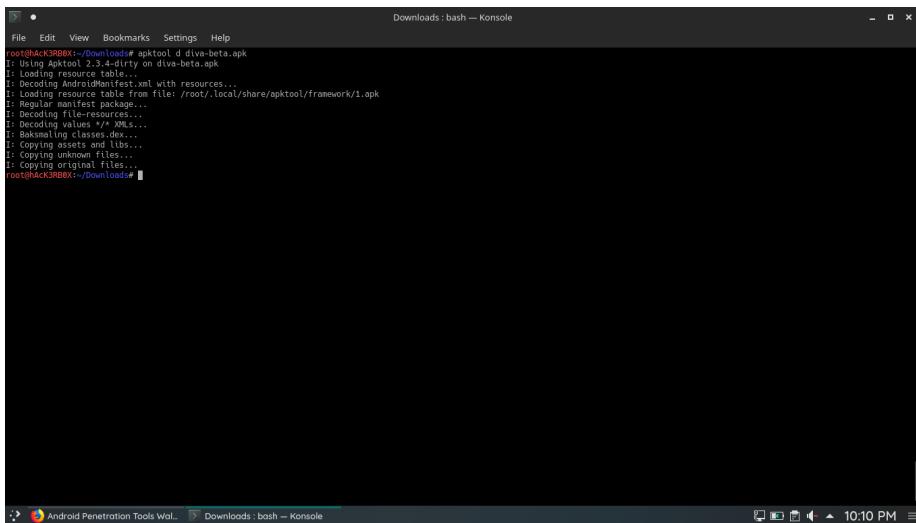


```
root@ACK3R8BX:~/Downloads# apktool
apktool 2.3.4-dirty (https://ibatpaches.github.io/Apktool/)
Copyright 2014 Ryszard Wisniewski <brut.a@illmail.com>
Copyright 2014 Connor Tumleson <connor.tumleson@gmail.com>

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

For additional info, see: https://ibatpaches.github.io/Apktool/
For smali/baksmali info, see: https://github.com/JesusFreke/smali
root@ACK3R8BX:~/Downloads#
```

Figure 27: Apktool



```
root@ACK3R8BX:~/Downloads# apktool d diva-beta.apk
I: Using Apktool 2.3.4-dirty on diva-beta.apk
I: Loading resource table...
I: Loading resource manifest...
I: Loading resource table from file: /root/.local/share/apktool/framework/1.apk
I: Regular manifest package...
I: Decoding resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying original files...
root@ACK3R8BX:~/Downloads#
```

Figure 28: Decoding the diva-beta APK File

Figure 29: Displaying AndroidManifest.xml file

Figure 30: Browsing the Decompiled App Directory

```

root@hAckJ3RBX:~/Downloads# apktool b diva-beta
E: Using Apktool 2.3.4-dirty
brut.directory.DirectoryException: java.nio.file.NoSuchFileException: diva-beta
root@hAckJ3RBX:~/Downloads# ./v2_1_02.sh diva-beta diva-beta.apk hAckJ3RBX.ovpn Kali-Linux-Revealed-1st-edition.pdf
E: Using Apktool 2.3.4-dirty
E: Checking whether apk has changed...
E: Copying apk folder into classes.dex...
E: Checking whether resources has changed...
E: Building resources
brut.common.BruteException: brut.common.BruteException: Could not extract resource: /prebuilt/aapt/linux/aapt (defaulting to $PATH binary)
E: Copying libs... (/lib)
E: Building apk file...
E: Copying unknown files/dir...
E: Copying manifest...
root@hAckJ3RBX:~/Downloads# ls
purpsuite_community_linux_v2_1_02.sh diva-beta diva-beta.apk hAckJ3RBX.ovpn Kali-Linux-Revealed-1st-edition.pdf
total 1540
drwxr-xr-x 3 root root 4096 Sep 29 22:09 .
drwxr-xr-x 29 root root 4096 Sep 29 21:51 ..
drwxr-xr-x 8 root root 146556 Sep 29 22:45 purpsuite_community_linux_v2_1_02.sh
drwxr-xr-x 8 root root 4096 Sep 29 22:17 diva-beta
-rw-r--r-- 1 root root 1475989 Sep 29 22:06 diva-beta.apk
-rw-r--r-- 1 root root 1037239 Aug 3 20:28 hAckJ3RBX.ovpn
-rw-r--r-- 1 root root 10337239 Aug 3 20:28 Kali-Linux-Revealed-1st-edition.pdf
root@hAckJ3RBX:~/Downloads# cd diva-beta/
root@hAckJ3RBX:~/Downloads/diva-beta# ls
apktool build apk build/ dist lib original res small
root@hAckJ3RBX:~/Downloads/diva-beta# cd build/
root@hAckJ3RBX:~/Downloads/diva-beta/build# ls
apk
root@hAckJ3RBX:~/Downloads/diva-beta/build# cd apk/
root@hAckJ3RBX:~/Downloads/diva-beta/build/apk# ls
AndroidManifest.xml classes.dex lib res resources.arsc
root@hAckJ3RBX:~/Downloads/diva-beta/build/apk# cd ..
root@hAckJ3RBX:~/Downloads/diva-beta/build# cd ..
root@hAckJ3RBX:~/Downloads/diva-beta# cd dist/
root@hAckJ3RBX:~/Downloads/diva-beta/dist# ls
dive-beta.apk
root@hAckJ3RBX:~/Downloads/diva-beta/dist# 

```

Figure 31: Recompiling the App from the Decompiled Directory

# Hydra

Hydra (better known as “thc-hydra”) is an online password attack tool. It is very fast and flexible, and new modules are easy to add. This tool makes it possible for researchers and security consultants to show how easy it would be to gain unauthorized access to a system remotely. It brute forces various combinations on live services like telnet, ssh, http, https, smb, snmp, smtp etc. Hydra supports 30+ protocols including their SSL enabled ones. It brute forces on services we specify by using user-lists & wordlists.

Hydra works in 4 modes:

- One username & one password
- User-list & One password
- One username & Password list
- User-list & Password list

```

root@kali:~# hydra
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Syntax: hydra [[[-l LOGIN|-L FILE] [-p PASS|-P FILE]] | [-C FILE]] [-e nsr] [-o FILE] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f]
[-s PORT] [-x MIN:MAX:CHARSET] [-c TIME] [-S0uvv446] [service://server[:PORT]/OPT]

Options:
-l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
-p PASS or -D FILE try password PASS, or load several passwords from FILE
-C FILE colon separated "login:pass" format, instead of -L/-P options
-M FILE list of servers to attack, one entry per line, ':' to specify port
-t TASKS run TASKS number of connects in parallel per target (default: 16)
-u SERVICE specify service to use
-h more command line options (COMPLETE HELP)
-server the target: DNS IP or 192.168.0.0/24 (this OR the -M option)
-service the service to crack (see below for supported protocols)
-OPT some service modules support additional input (-U for module help)

Supported services: adam0500 asterisk cisco cisco-enable cvs firebird ftp[s] http[s]-(head|get|post) http[s]-(get|post)-form http-proxy http-pr
oxy-ssl http[s] irc ldap2[s] ldap3[!(cramp|digest|md5)](s) memcached mongodb mssql mysql nntp oracle-listener oracle-sid pcanywhere pcfns
pgsql[s] postgres radmin2 rdp redis rexec rlogin rpcap rsh rtsp s7-300 sip smb smtp[s] smtp-enum smtp socks5 ssh sshkey svn teamspeak telnet[s]
vsftpd vnc vncpp

Hydra is a tool to guess/crack valid login/password pairs. Licensed under AGPL
v3.0. The newest version is always available at https://github.com/vanhauser-thc/thc-hydra
Don't use in military or secret service organizations, or for illegal purposes.

Example: hydra -l user -P passlist.txt ftp://192.168.0.1
root@kali:#

```

Figure 32: Hydra

```

coep@coep-HP-Compaq-6200-Pro-SFF-PC:~$ sudo systemctl restart vsftpd
coep@coep-HP-Compaq-6200-Pro-SFF-PC:~$ ftp 10.1.26.111
Connected to 10.1.26.111.
220 (vsFTPd 3.0.3)
Name (10.1.26.111:coep): johnwick
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> exit
221 Goodbye.
coep@coep-HP-Compaq-6200-Pro-SFF-PC:~$

```

Figure 33: Creating FTP Server

A screenshot of a Kali Linux desktop environment. A terminal window titled 'root@kali: ~' is open, showing the output of a Hydra attack against an FTP server. The terminal shows two separate runs of Hydra. In the first run, it finds a password for a user named 'johnwick'. In the second run, it finds another password for the same user. Both runs were completed at 2019-09-30 17:49. The terminal window has a dark blue background with a light blue header bar. The Kali logo is visible in the bottom right corner of the desktop.

```
root@kali: ~
File Edit View Search Terminal Help
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-09-30 17:49:31
[DATA] max 16 tasks per 1 server, overall 16 tasks, 3559 login tries (l:1/p:3559
), -223 tries per task
[DATA] attacking ftp://10.1.26.111:21/
[21][ftp] host: 10.1.26.111 login: johnwick password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-09-30 17:49:
root@kali: ~# hydra -l johnwick -P /usr/share/john/password.lst ftp://10.1.26.111
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-09-30 17:50:05
[DATA] max 16 tasks per 1 server, overall 16 tasks, 3559 login tries (l:1/p:3559
), -223 tries per task
[DATA] attacking ftp://10.1.26.111:21/
[21][ftp] host: 10.1.26.111 login: johnwick password: abc123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-09-30 17:50:11
root@kali: ~#
```

Figure 34: Cracking FTP Server

A screenshot of a Kali Linux desktop environment. A terminal window titled 'root@kali: ~' is open, showing the output of a Hydra attack against an FTP server. The terminal shows a single run of Hydra that finds a password for a user named 'johnwick'. The run was completed at 2019-09-30 17:50:45. The terminal window has a dark blue background with a light blue header bar. The Kali logo is visible in the bottom right corner of the desktop.

```
root@kali: ~# hydra -l johnwick -P /usr/share/john/password.lst ftp://10.1.26.111
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret
service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-09-30 17:50:45
[DATA] max 16 tasks per 1 server, overall 16 tasks, 3559 login tries (l:1/p:3559), -223 tries per task
[DATA] attacking ftp://10.1.26.111:21/
[21][ftp] host: 10.1.26.111 login: johnwick password: abc123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-09-30 17:50:52
root@kali: ~#
```

Figure 35: Cracking FTP Server

# **RainbowCrack**

RainbowCrack is a revolutionary hash cracker that we have shared on this page along with Rainbow Tables free to download. It allows user to decrypt a hashed password into understandable plain text. With this software, a password of any length can be cracked in a matter of minutes no matter how strong the opposite security encryption is.

## **Working**

RainbowCrack does not uses brute force attacks like other software. It uses time-memory tradeoff algorithm which uses more storage in order to reduce time. It first requires a pre-computation stage in which all plaintext/hash pairs are computed and stored in rainbow table. There are Terabytes of Rainbow Tables available online, some are free while some paid. It cracks a hash in few minutes that would take up to 4 weeks using brute force attack.



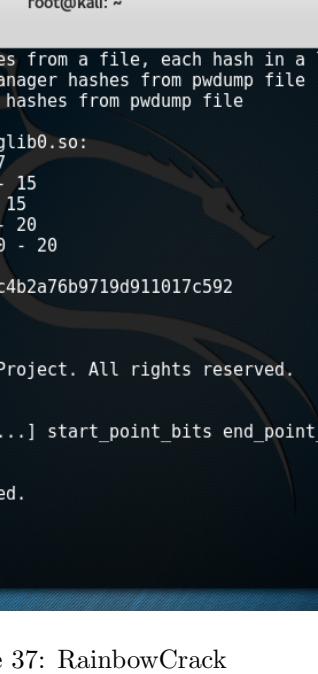
```
root@kali: ~
File Edit View Search Terminal Help
E: Connection timed out
E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?
root@kali:~# rcrack
RainbowCrack 1.6.1
Copyright 2003-2015 RainbowCrack Project. All rights reserved.
http://project-rainbowcrack.com/

usage: rcrack rt_files [rt_files ...] -h hash
       rcrack rt_files [rt_files ...] -l hash_list_file
       rcrack rt_files [rt_files ...] -f pwdump_file
       rcrack rt_files [rt_files ...] -n pwdump_file
rt_files:           path to the rainbow table(s), wildchar(*, ?) supported
-h hash:            load single hash
-l hash_list_file: load hashes from a file, each hash in a line
-f pwdump_file:    load lanmanager hashes from pwdump file
-n pwdump_file:    load ntlm hashes from pwdump file

hash algorithms implemented in alglib0.so:
  lm, plaintext_len limit: 0 - 7
  ntlm, plaintext_len limit: 0 - 15
  md5, plaintext_len limit: 0 - 15
  sha1, plaintext_len limit: 0 - 20
  sha256, plaintext_len limit: 0 - 20

example: rcrack *.rt -h 5d41402abc4b2a76b9719d911017c592
```

Figure 36: RainbowCrack



```
root@kali: ~
File Edit View Search Terminal Help
-l hash_list_file:    load hashes from a file, each hash in a line
-f pwdump_file:       load lanmanager hashes from pwdump file
-n pwdump_file:       load ntlm hashes from pwdump file

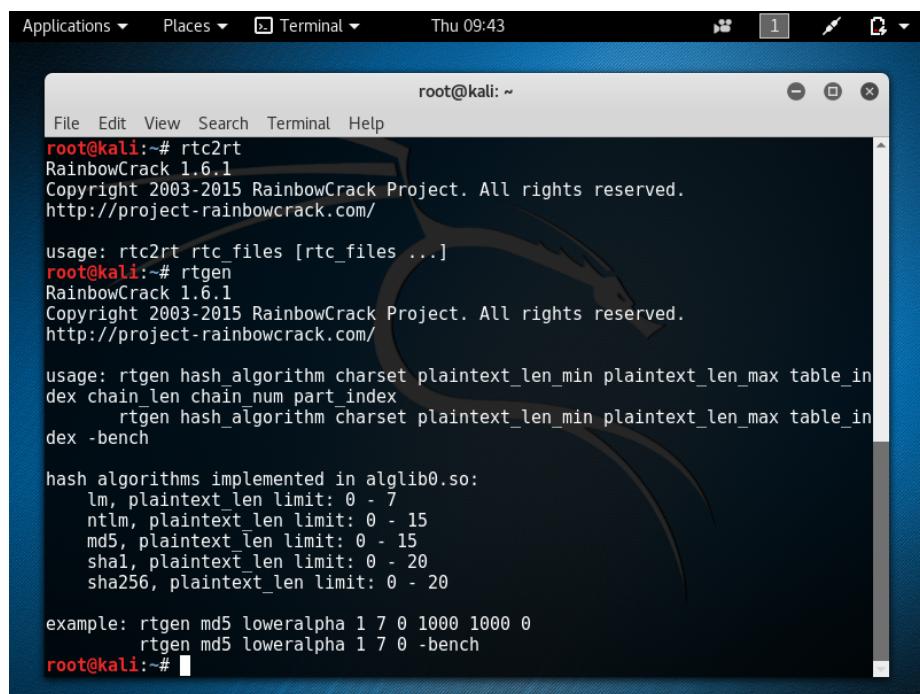
hash algorithms implemented in alglib0.so:
  lm, plaintext_len limit: 0 - 7
  ntlm, plaintext_len limit: 0 - 15
  md5, plaintext_len limit: 0 - 15
  sha1, plaintext_len limit: 0 - 20
  sha256, plaintext_len limit: 0 - 20

example: rcrack *.rt -h 5d41402abc4b2a76b9719d911017c592
          rcrack *.rt -l hash.txt
root@kali:~# rt2rtc
RainbowCrack 1.6.1
Copyright 2003-2015 RainbowCrack Project. All rights reserved.
http://project-rainbowcrack.com/

usage: rt2rtc rt_files [rt_files ...] start_point_bits end_point_bits [-m chunk_size_in_mb] [-p]

Input rainbow tables must be sorted.
1 <= start_point_bits <= 64
1 <= end_point_bits  <= 64
1 <= chunk_size_in_mb
root@kali:~#
```

Figure 37: RainbowCrack



A screenshot of a Kali Linux terminal window titled "root@kali: ~". The window shows the output of the "rtc2rt" and "rtgen" commands. The "rtc2rt" command outputs the RainbowCrack 1.6.1 copyright notice and usage information for "rtc2rt". The "rtgen" command outputs the RainbowCrack 1.6.1 copyright notice and usage information for "rtgen", which includes options for hash algorithm, charset, plaintext length, and table index. The "rtgen" command also lists the hash algorithms implemented in alglib0.so: lm, ntlm, md5, sha1, and sha256, along with their respective plaintext length limits. An example command is shown: "rtgen md5 loweralpha 1 7 0 1000 1000 0".

```
root@kali:~# rtc2rt
RainbowCrack 1.6.1
Copyright 2003-2015 RainbowCrack Project. All rights reserved.
http://project-rainbowcrack.com/

usage: rtc2rt rtc_files [rtc_files ...]
root@kali:~# rtgen
RainbowCrack 1.6.1
Copyright 2003-2015 RainbowCrack Project. All rights reserved.
http://project-rainbowcrack.com/

usage: rtgen hash_algorithm charset plaintext_len_min plaintext_len_max table_in
dex chain_len chain_num part_index
      rtgen hash_algorithm charset plaintext_len_min plaintext_len_max table_in
dex -bench

hash algorithms implemented in alglib0.so:
  lm, plaintext_len limit: 0 - 7
  ntlm, plaintext_len limit: 0 - 15
  md5, plaintext_len limit: 0 - 15
  sha1, plaintext_len limit: 0 - 20
  sha256, plaintext_len limit: 0 - 20

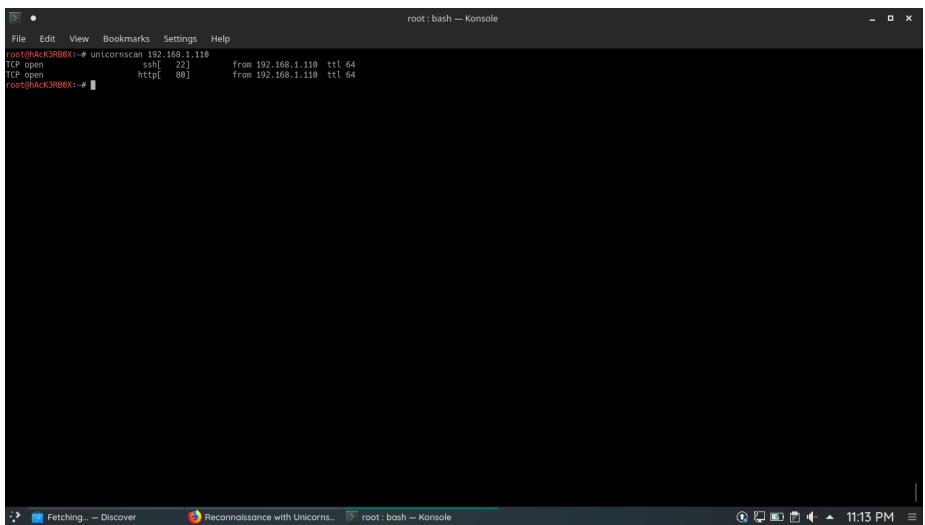
example: rtgen md5 loweralpha 1 7 0 1000 1000 0
          rtgen md5 loweralpha 1 7 0 -bench
root@kali:~#
```

Figure 38: RainbowCrack

# UnicornScan

Unicornscan is a sophisticated, powerful and stateless port scanner that uses stimulus into and measuring a response from any TCP/IP enabled device. One of the key features of unicornscan that sets it apart of nmap and other port scanners is that it has its own TCP/IP stack. The other port scanners all use the underlying host operating system's TCP/IP stack. This enables unicornscan to scan much more quickly than the others as it can, for instance, send out SYN packets with one thread and receive the responses with another thread. This can make a huge difference when scanning very large networks as a security researcher/ pentester where we might be scanning thousands of IP addresses and be even more important to an attacker who may be scanning millions of addresses looking for a particular open port or vulnerability. In addition, because it has its own TCP/IP stack, it is capable of sending packets with different OS fingerprints that the operating system of your host. This can be very useful for obscuring your identity, especially combined with IP spoofing.

This simple syntax will return for us the open TCP ports on the target system, very similar to the nmap -sS scan, but without the default ICMP that nmap uses. As you can see, unicornscan reports back to us that ports 80 are open on the target system.



The screenshot shows a terminal window titled "root:bash — Konsole". The command "unicornscan 192.168.1.118" was run, resulting in the following output:

```
root@Hack3RDBOX:~# unicornscan 192.168.1.118
TCP open          ssh[ 22]      from 192.168.1.110 ttl 64
TCP open          http[ 80]     from 192.168.1.110 ttl 64
root@Hack3RDBOX:~#
```

Figure 39: Unicorn Scan

# WPScan

WPScan is a vulnerability scanner sponsored by SUCURI used to identify the security-related problems on WordPress website. WPScan is useful if the website is on a private network or Intranet where the Internet is not available. WPScan is basically used to find for known vulnerabilities within the core version, plugins, and themes. It is also used to find out if any weak passwords, users, and security configuration issues are present. The database at wpvulndb.com is used to check for vulnerable software and the WPScan team maintains the ever-growing list of vulnerabilities.

Running the basic command above will perform a quick scan of the website to identify your active theme and basic issues, such as exposed WordPress version numbers.

## Checking for Vulnerable Plugins

Adding the `-enumerate vp` argument checks the WordPress website for vulnerable plugins.

If vulnerable plugins are found you will see red exclamation icons and references to further information. Any vulnerable plugin should be replaced and removed if you cannot update it to patch the vulnerability.

## Checking User Enumeration

When hackers know your WordPress usernames it becomes easier for them to perform a successful brute force attack. If attackers gain access to one of your users with sufficient permissions, they can gain control of your WordPress installation.

To find out the login names of users on your WordPress website, we will use the argument `-enumerate u` at the end of the command.

```
File Edit View Bookmarks Settings Help root : bash — Konsole
root@iAeK3RBBX:~# wpscan --url geekflare.com --enumerate vp --random-user-agent
[!] WordPress Security Scanner by the WPScan Team
[!] Version 3.5.3
[!] Sponsored by Sucuri - https://sucuri.net
[!] WPScan_ , @ethicalhacker3r , erwan_lr , @_FireFart_ ,

[+] URL: http://geekflare.com/
[+] Effective URL: https://geekflare.com/
[+] Started: Mon Sep 30 07:43:52 2019

Interesting Finding(s):

[*] https://geekflare.com/
  Interesting Entries:
  | - x-socache-fetch-status: HIT
  | - x-socache-store-status: BYPASS
  | - x-socache-store-time: 2019-09-30T07:43:52Z
  | - via: 1.1 google
  | - alt-svc: h3-23=":443"; ma=86400
  | - expect-ct: max-age=604800; report-uri="https://report-uri.cloudflare.com/cdn-cgi/expect-ct"
  | - report-to: report-to[{"endpoints": [{"url": "https://report-uri.cloudflare.com/cdn-cgi/report-to-endpoint"}]}
  | - cf-ray: 51e20e270ec1d5f4-BOM
  | Found By: Headers (Positive Detection)
  | Confidence: 100%
  | Confidence: 100%

[*] This site has Must Use Plugins: http://geekflare.com/wp-content/mu-plugins/
  | Found By: Direct Access (Aggressive Detection)
  | Confidence: 80%
  | References: http://codex.wordpress.org/Must_Use_Plugins

[*] WordPress version 5.2.3 was identified (Latest, released on 2019-09-05).
  | Details: https://geekflare.com/.htaccess
  | - https://geekflare.com/.htaccess - <generator>https://wordpress.org/?v=5.2.3</generator>
  | - https://geekflare.com/comments/feed/ - <generator>https://wordpress.org/?v=5.2.3</generator>

[*] WordPress theme in use: authority-pro
```

Figure 40: Scanning a Wordpress site for Vulnerability - 1

```
root@localhost:~# ./checkWP.sh geekflare.com
[+] WordPress theme in use: authority-pro
| Location: http://geekflare.com/wp-content/themes/authority-pro/
| Style URL: https://geekflare.com/wp-content/themes/authority-pro/style.css

[+] Detected By: Css Style (Passive Detection)
| The version could not be determined.

[+] Enumerating Vulnerable Plugins (via Passive Methods)
[+] Checking Plugin Versions (via Passive and Aggressive Methods)

[!] Plugin(s) Identified:

[!] elementor
| Location: http://geekflare.com/wp-content/plugins/elementor/
| Latest Version: 2.7.3
| Last Updated: 2019-09-24T13:58:00.000Z

[+] Detected By: Urls In Homepage (Passive Detection)

[!] 1 vulnerability identified:

[!] Title: Elementor Page Builder <= 1.7.12 - Authenticated Unrestricted Editing
| Fixed in: 1.8.0
| References:
| - https://wpvulndb.com/vulnerabilities/9056
| - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-18596
| - http://www.practical-net.net/log/elementor-page-builder-1-8-allows-logged-users-unrestricted-editing

| The version could not be determined.

[+] megamenu
| Location: http://geekflare.com/wp-content/plugins/megamenu/
| Latest Version: 2.7.2
| Last Updated: 2019-09-09T09:43:00.000Z

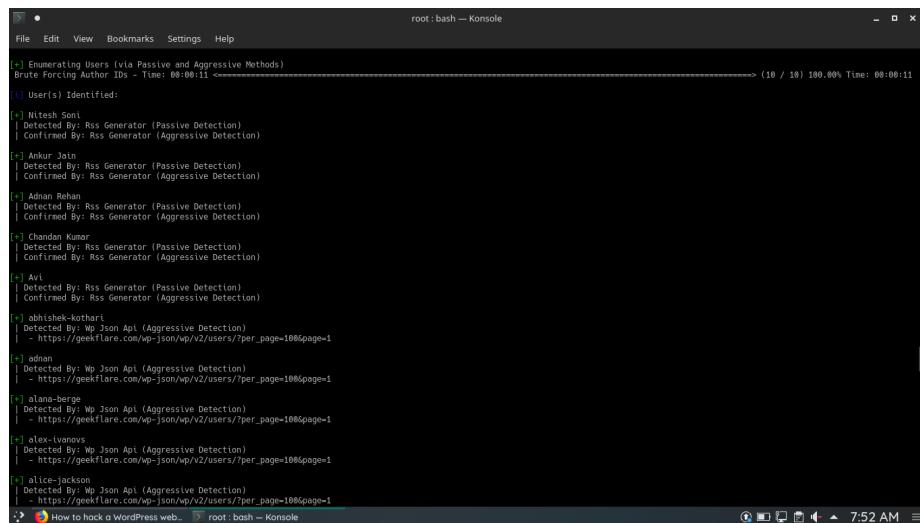
[+] Detected By: Urls In Homepage (Passive Detection)

[!] 1 vulnerability identified:

[!] Title: Max Mega Menu <= 2.3.8 - Authenticated XSS
| Fixed in: 2.4
| References:
| - https://wpvulndb.com/vulnerabilities/9343
| - https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-18525

[+] How to hack a WordPress web... root@localhost:~#
```

Figure 41: Scanning a Wordpress site for Vulnerability - 2



The screenshot shows a terminal window titled "root : bash — Konsole". The window displays the output of a tool for enumerating users via passive and aggressive methods. The output is as follows:

```
[+] Enumerating Users (via Passive and Aggressive Methods)
Brute Forcing Author IDs - Time: 00:00:11 <===== ( 10 / 10 ) 100.00% Time: 00:00:11
[+] User(s) Identified:
[*] Nitesh Soni
| Detected By: Rss Generator (Passive Detection)
| Confirmed By: Rss Generator (Aggressive Detection)
[*] Anil Kumar
| Detected By: Rss Generator (Passive Detection)
| Confirmed By: Rss Generator (Aggressive Detection)
[*] Adnan Rohan
| Detected By: Rss Generator (Passive Detection)
| Confirmed By: Rss Generator (Aggressive Detection)
[*] Chandon Kumar
| Detected By: Rss Generator (Passive Detection)
| Confirmed By: Rss Generator (Aggressive Detection)
[*] Avi
| Detected By: Rss Generator (Passive Detection)
| Confirmed By: Rss Generator (Aggressive Detection)
[*] abhishek-kothari
| Detected By: Wp Json Api (Aggressive Detection)
| - https://geekflare.com/wp-json/wp/v2/users/?per_page=10&page=1
[*] adnan
| Detected By: Wp Json Api (Aggressive Detection)
| - https://geekflare.com/wp-json/wp/v2/users/?per_page=100&page=1
[*] olano-berge
| Detected By: Wp Json Api (Aggressive Detection)
| - https://geekflare.com/wp-json/wp/v2/users/?per_page=100&page=1
[*] alex-ivanovs
| Detected By: Wp Json Api (Aggressive Detection)
| - https://geekflare.com/wp-json/wp/v2/users/?per_page=100&page=1
[*] alice-jackson
| Detected By: Wp Json Api (Aggressive Detection)
| - https://geekflare.com/wp-json/wp/v2/users/?per_page=100&page=1
```

Figure 42: Listing all users from a Particular Wordpress Site

# SlowHTTPTest

SlowHTTPTest is a highly configurable tool that simulates some Application Layer Denial of Service attacks by prolonging HTTP connections in different ways. Currently, the supported attacks by the slowhttptest library are:

- Slowloris
- Slow HTTP POST
- Apache Range Header
- Slow Read

Slowloris and Slow HTTP POST DoS attacks rely on the fact that the HTTP protocol, by design, requires requests to be completely received by the server before they are processed. If an HTTP request is not complete, or if the transfer rate is very low, the server keeps its resources busy waiting for the rest of the data. If the server keeps too many resources busy, this creates a denial of service. This tool is sending partial HTTP requests, trying to get denial of service from target HTTP server.

Slow Read DoS attack aims the same resources as slowloris and slow POST, but instead of prolonging the request, it sends legitimate HTTP request and reads the response slowly. As you can see, our target is our own website, however even with 100 connections, our server doesn't hang at all because we do have protection against this kind of attacks. The service available will be always YES if the target is reachable. You can test with another computer/network if the website is still up indeed. The generate output in HTML created by our options, will be the following one:

```

apt:bash — Konsole
File Edit View Bookmarks Settings Help
root@ACK3RBOX:/etc/apt# slowhttptest -h
Usage: slowhttptest [options ...]
Test modes:
  -S      slow headers a.k.a. SlowWoris (default)
  -B      slow body a.k.a R-U-Dead-Yet
  -R      range attack a.k.a Apache killer
  -X      slow read a.k.a Slow Read
Reporting options:
  -d      generate statistics with socket state changes (off)
  -o file_prefix save statistics output in file.html and file.csv (-o required)
  -v level    verbosity level @=4: Fatal, Info, Error, Warning, Debug
General options:
  -c connections target number of connections (50)
  -t seconds   target test duration in seconds (10)
  -l seconds   target test length in seconds (240)
  -r rate      connections per seconds (50)
  -k bytes     value of Content-Length header if needed (4096)
  -t verb      verb to use, default to GET for
               slow headers and response and to POST for slow body
  -u URL       absolute URL or target (http://localhost/)
  -x bytes     max length of the randomized max-value pair of
               follow-up data per target (0-256 bytes)
               X-xx-xx for header or <xx>xx for body, where x
               is random character (32)
  -f content-type value of Content-Type header (application/x-www-form-urlencoded)
  -m accept    value of Accept header (text/html;q=0.9,soy/polymer_0.5,image/png,*/*;q=0.5)
Probe/Proxy options:
  -d hostport  all traffic directed through HTTP proxy at host:port (off)
  -e hostport  probe traffic directed through HTTP proxy at host:port (off)
  -p seconds   timeout to wait for HTTP response on probe connection,
               after which server is considered inaccessible (5)
Range attack specific options:
  -a start    left boundary of range in range header (5)
  -b bytes    limit for range header right boundary values (2000)
root@ACK3RBOX:/etc/apt# slowhttptest -S -c 100 -t 10 -u http://localhost/
DoS website using slowhttptest... apt:bash — Konsole
10:48 PM
```

Figure 43: SlowHTTPTest

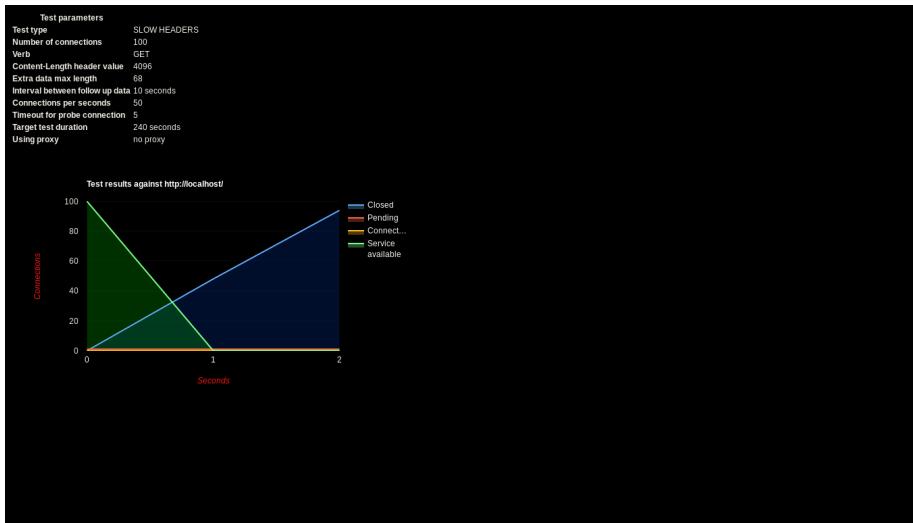


Figure 44: Result Generated after performing a small DOS Attack

# Nikto

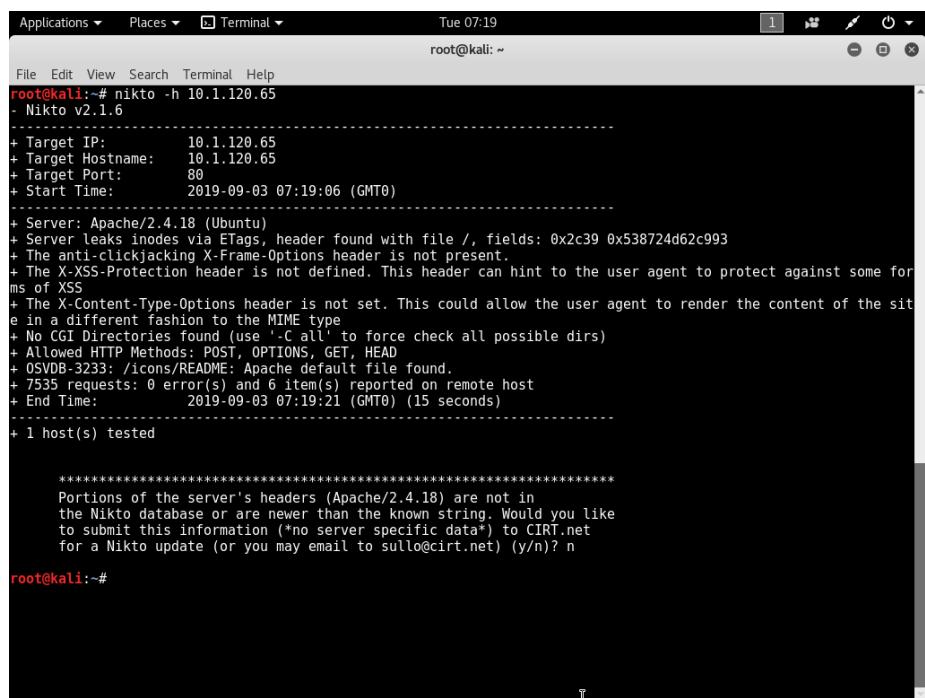
Nikto is an Open Source (GPL) web server scanner which performs comprehensive tests against web servers for multiple items, including over 3500 potentially dangerous files/CGIs, versions on over 900 servers, and version specific problems on over 250 servers. Scan items and plugins are frequently updated and can be automatically updated. Nikto is not designed as an overly stealthy tool. It will test a web server in the shortest timespan possible, and it's fairly obvious in log files.

## Lengthy Nikto run time

Due to the number of security checks that this tool performs a scan can take 45 mins or even longer, depending on the speed of your web server.

## False Positives with Nikto

Nikto does quite well in detecting web server configurations that return HTTP 200 OK on actual “page not found” results. Since Nikto is checking hundreds of URL’s for the presence of old scripts, vulnerable applications and other problems. This can sometimes result in many false positives if the detection of the 404 -*&* 200 is not discovered by Nikto. It is not difficult to spot as you will receive a great deal of invalid urls as positives. These are easily checked manually to ensure they are actual false positives.



The screenshot shows a terminal window titled "Terminal" with the command "nikto -h 10.1.120.65" run by root. The output of the scan is displayed, detailing various findings such as target information, server headers, and security vulnerabilities. A prompt at the end asks if the user wants to submit the findings to CIRT.net.

```
root@kali:~# nikto -h 10.1.120.65
- Nikto v2.1.6
-----
+ Target IP:      10.1.120.65
+ Target Hostname: 10.1.120.65
+ Target Port:    80
+ Start Time:    2019-09-03 07:19:06 (GMT0)
-----
+ Server: Apache/2.4.18 (Ubuntu)
+ Server leaks inodes via ETags, header found with file /, fields: 0x2c39 0x538724d62c993
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Allowed HTTP Methods: POST, OPTIONS, GET, HEAD
+ OSVDB-3233: /icons/README: Apache default file found.
+ 7535 requests: 0 error(s) and 6 item(s) reported on remote host
+ End Time:        2019-09-03 07:19:21 (GMT0) (15 seconds)
-----
+ 1 host(s) tested

*****
Portions of the server's headers (Apache/2.4.18) are not in
the Nikto database or are newer than the known string. Would you like
to submit this information ("no server specific data") to CIRT.net
for a Nikto update (or you may email to sullo@cirt.net) (y/n)? n
root@kali:~#
```

Figure 45: Nikto Sample Scan

# Fluxion

Fluxion is a security auditing and social-engineering research tool. It is a remake of linset by vk496 with (hopefully) fewer bugs and more functionality. The script attempts to retrieve the WPA/WPA2 key from a target access point by means of a social engineering (phishing) attack.

## Working

- Scan for a target wireless network.
- Launch the Handshake Snooper attack.
- Capture a handshake (necessary for password verification).
- Launch Captive Portal attack.
- Spawns a rogue (fake) AP, imitating the original access point.
- Spawns a DNS server, redirecting all requests to the attacker's host running the captive portal.
- Spawns a web server, serving the captive portal which prompts users for their WPA/WPA2 key.
- Spawns a jammer, deauthenticating all clients from original AP and luring them to the rogue AP.
- All authentication attempts at the captive portal are checked against the handshake file captured earlier.
- The attack will automatically terminate once a correct key has been submitted.
- The key will be logged and clients will be allowed to reconnect to the target access point.

# findmyhash

findmyhash is a python script that connects to different online resources to find cracked hashes. It currently supports following hashes:

- MD4
- MD5
- SHA1
- SHA256
- RMD160
- LM
- NTLM
- MYSQL
- CISCO7
- JUNIPER

## Syntax

*python findmyash.py [algorithm] [OPTIONS]*

# Wireshark

Wireshark is a network packet analyzer. A network packet analyzer will try to capture network packets and tries to display that packet data as detailed as possible.

## Features

- Available for UNIX and Windows.
- Capture live packet data from a network interface.
- Open files containing packet data captured with tcpdump/WinDump, Wireshark, and many other packet capture programs.
- Import packets from text files containing hex dumps of packet data.
- Display packets with very detailed protocol information.
- Save packet data captured.
- Export some or all packets in a number of capture file formats.
- Filter packets on many criteria.
- Search for packets on many criteria.
- Colorize packet display based on filters.
- Create various statistics.

## Common Misconceptions of Wireshark

- Wireshark isn't an intrusion detection system. It will not warn you when someone does strange things on your network that he/she isn't allowed to do. However, if strange things happen, Wireshark might help you figure out what is really going on.
- Wireshark will not manipulate things on the network, it will only "measure" things from it. Wireshark doesn't send packets on the network or do other active things (except domain name resolution, but that can be disabled).

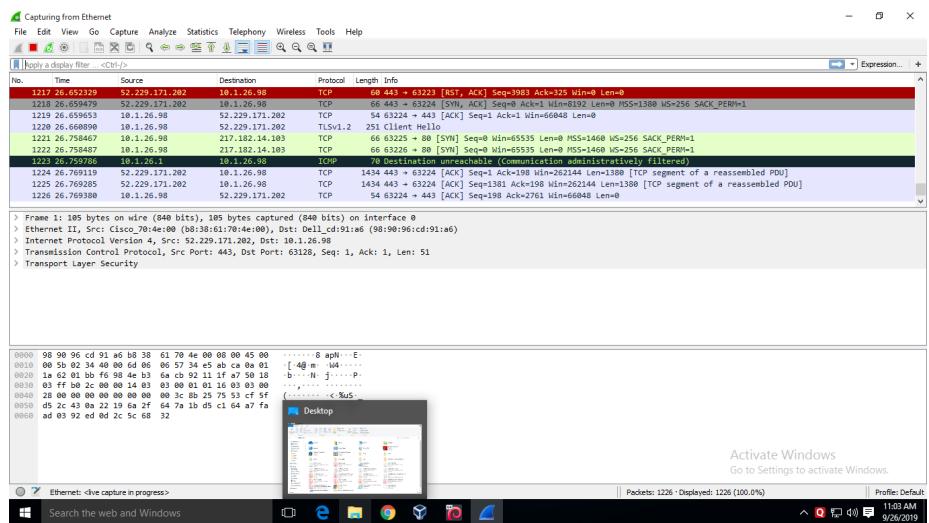


Figure 46: Wireshark Sample Packet Capture

# Metasploit Framework

Metasploit offers developers and ethical hackers with a platform which is dedicated to exploit testing. It is a great platform which offers testers a free of cost but a potent way to Pentest systems, websites, and networks. An expert tester can utilize this tool to check for any vulnerabilities and possible loopholes which can be exploited by malicious attackers.

This amazing platform has the ability to highlight such false positive threats and other loopholes, it identifies such threats in no time. The administrator can opt for automatic analysis of threats and vulnerabilities and prioritize the task accordingly. Even there is an option for real-time scanning and testing.

## Metasploit Pro Features

- It conducts a thorough test of your web app and provides you with a complete audit report.
- Validate vulnerabilities and bugs found in applications.
- You can measure the possible exposure to phishing attacks as well

# BurpSuite

Burp Suite is a Java based Web Penetration Testing framework. Burp Suite helps you identify vulnerabilities and verify attack vectors that are affecting web applications. Burp Suite can be classified as an Interception Proxy. While browsing their target application, a penetration tester can configure their internet browser to route traffic through the Burp Suite proxy server. Burp Suite then acts as a Man In The Middle by capturing and analyzing each request to and from the target web application so that they can be analyzed. Penetration testers can pause, manipulate and replay individual HTTP requests in order to analyze potential parameters or injection points. Injection points can be specified for manual as well as automated fuzzing attacks to discover potentially unintended application behaviors, crashes and error messages.

- **HTTP Proxy** : It operates as a web proxy server, and sits as a man-in-the-middle between the browser and destination web servers. This allows the interception, inspection and modification of the raw traffic passing in both directions.
- **Scanner** : A web application security scanner, used for performing automated vulnerability scans of web applications.
- **Intruder** : This tool can perform automated attacks on web applications. The tool offers a configurable algorithm that can generate malicious HTTP requests. The intruder tool can test and detect SQL injections, cross-site scripting, parameter manipulation and vulnerabilities susceptible to brute-force attacks.
- **Spider** : A tool for automatically crawling web applications. It can be used in conjunction with manual mapping techniques to speed up the process of mapping an application's content and functionality.
- **Repeater** : A simple tool that can be used to manually test an application. It can be used to modify requests to the server, resend them, and observe the results.
- **Decoder** : A tool for transforming encoded data into its canonical form, or for transforming raw data into various encoded and hashed forms. It is capable of intelligently recognizing several encoding formats using heuristic techniques.
- **Comparer** : A tool for performing a comparison (a visual "diff") between any two items of data.
- **Extender** : Allows the security tester to load Burp extensions, to extend Burp's functionality using the security tester's own or third-party code (BAppStore)
- **Sequencer** : A tool for analyzing the quality of randomness in a sample of data items. It can be used to test an application's session tokens or

other important data items that are intended to be unpredictable, such as anti-CSRF tokens, password reset tokens, etc.