**ARP SPOOFING**

**USING BETTERCAP**

**Framework to run network attacks**

**Can be used to:**

**- ARP Spoof tagets(redirect the flow of packets)**

**- Sniff data(urls, username, passwords)**

**- Bypass HTTPS**

**- Redirect domain requests(DNS Spoofing)**

**- Inject code in loaded pages**

**And more!**

**To run BetterCAP:**

bettercap -iface eth0

(tool)

(-iface to specify the interface that is connected to the network that i want to run the attacks against)

(the interface)

We will have a different prompt now in which we can use the commands of BetterCAP

**We can type help to list all comands**

**There is also Modules which are important**

These are all the Modules that we can use

At first only **events.stream** should work which is basically the Module that runs in the background to handle all the events

**You can type help followed by the name of any Module you want and this will show you a help menu - how to use this specific Module**

We will see the **net.probe** and **net.recon** Modules

**So we type:**

help net.probe

We will get description of what this Module does

Basically it keeps sending UDP packets to discover devices on the same network and we can do:

**net.probe on**

**to turn on the Module and net.probe off to turn it off**

You can also see all the options that you can modify for this Module, we will keep them default for now

When we turn it on we can see that this will automatically start discovering clients connected to the same network

This is a way of discovering connected clients quickly using **BetterCAP**

When we started **net.probe** it automatically started **net.recon**

We can check it with typing **help**

We will see **net.probe** running which we turned on manually and **net.recon** which got turned on automatically by **BetterCAP**

The reason fo this is because the net.probe sends probe requests to all possible IPs and then if we get a response, the net.recon will be the one detecting this response by monitoring our ARP cache and then adding all of these IPs in a nice list so we can target them

**So now when net.recon is running we can do:**

net.show

**To see all of the connected clients** and we can see a nice list of all of the connected clients, we can see **their IPs**, we can see the corresponding **MAC Addresses** for these clients and it also can show you info below Name about each one of these IPs

You can see also below Vendor its attempting to discover the manufacturer of the hardware used in each of these clients

**Now to ARP attack**

**The first thing we need to do is to use a Module called ARP spoof**

use help arp.spoof to see how to use it and see all the options we can set for it

**As usual we can do arp.spoof on to turn this Module on**

**And arp.spoof off to turn it off**

We can do **arp.ban on** - this will literally just cut the connection of the target

And **arp.ban off** to turn the ban off

Anything under **Parameters** in help section are the options that we can set for this specific Module

The options are in yellow that we can set or change and after them is their description and the default value

**For example we have arp.spoof.fullduplex**

Basically what this option will do if you set it to true, it will spoof both the router and the target

This would be useful if the router has some sort of protection against ARP spoofing attacks because you wont be interacting with the router at all but its also limiting because the router will communicate with the target device directly so we wont be able to inject stuff in the responses that the router sends to the target device

**If you leave it on false it will only spoof the target machine**

Now we will change this to **true** and the method we are going to use can be used to change any option in any Module in **BetterCAP**

set arp.spoof.fullduplex true

(set)(option name)(the value you want to set)

Now the option we want to change is the **targets**

In the description its telling us there are the targets that we want to run the attacks against and we can use a **comma( , )** **if i wanted to target more than one IP at the same time**

**So we do the same:**

set arp.spoof.targets 10.0.2.7

(set)(the option)(the IP of the target)

**Now we are ready to run the tool:**

arp.spoof on

**And to sniff the information(to see all the requests):**

arp.sniff on

Now it should tell us that the Module is running

If we do **help** again, we can see that **arp.spoof is running**

**! Make sure that the net.probe and the net.recon are running !**

Right now, BetterCAP should be doing what ARP spoofing was doing, fooling both the router and the target device and putting us in the middle of the connection

Now every time our target wants to send something to the router, it will send it to the Kali machine and because we set the full duplex option on in BetterCAP, the router also thinks that this Kali machine is the target machine, therefore anytime it needs to send a response to our target, it will actually send it to BetterCAP and as we said before that means every username, password, URL, anything the target sends or receives, will have to go through the Kali machine where we are going to be able to read it, modify it, or drop it