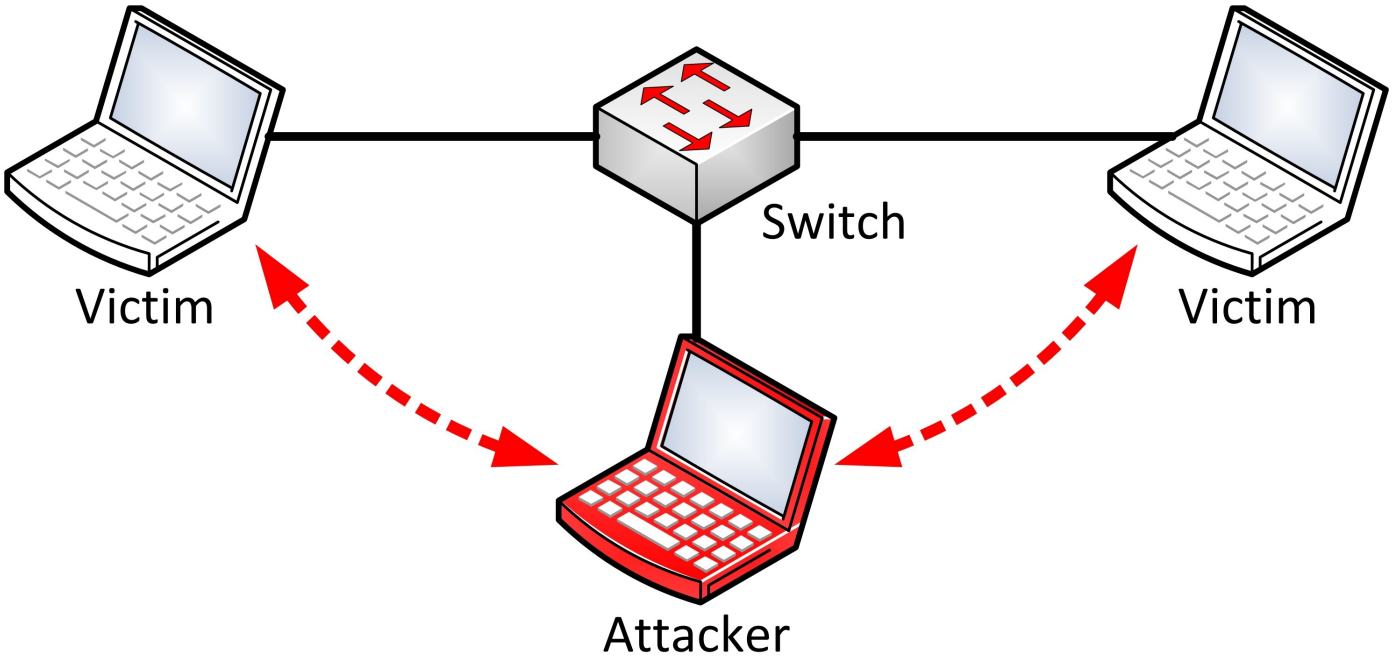
BETTER CAP

BETTER CAP

Introduction

BetterCAP is a powerful, flexible and portable tool created to perform various types of **MITM** attacks against a network, manipulate **HTTP**, **HTTPS** and **TCP** traffic in realtime, sniff for credentials and much more.

Nevertheless we can simplify the concept with an example. When you connect to some network (your home network, some public WiFi, StarBucks, etc), the router/switch is responsible for forwarding all of your packets to the correct destination, during a MITM attack we "force" the network to consider our device as the router (we "spoof" the original router/switch address in some way):



Once this happens, all of the network traffic goes through your computer instead of the legit router/switch and at that point you can do pretty much everything you want, from just sniffing for specific data (emails, passwords, cookies, etc of other people on your network) to actively intercepting and proxying all the requests of some specific protocol in order to modify them on the fly (you can, for instance, replace all images of all websites being visited by everyone, kill connections, etc).

BetterCap is responsible for giving the security researcher everything he needs in **one single tool** which simply works, on GNU/Linux, Mac OS X and OpenBSD systems.

Use Cases

You might think that BetterCAP is just another tool which helps script-kiddies to harm networks ... but it's much more than that, its use cases are many, for instance:

- Many professional penetration testers find a great companion in bettercap since its very first release.
- Reverse engineers are using it in order to reverse or modify closed network protocols.
- Mobile/IoT security researchers are exploiting bettercap capabilities to test the security of mobile systems.

Installation

BetterCap comes packaged as a **Ruby** gem, meaning you will need a Ruby interpreter (>= 1.9) and a RubyGems environment installed. Moreover, it is **fully compatible with GNU/Linux, Mac OS X and OpenBSD platforms**.

Installing on Kali Linux

Kali Linux has bettercap packaged and added to the **kali-rolling** repositories. To install bettercap and all dependencies in one fell swoop on the latest version of Kali Linux:

apt-get update

apt-get install bettercap

You can also clone the source code from the github repository:

```
git clone https://github.com/evilsocket/bettercap
cd bettercap
bundle install
gem build bettercap.gemspec
sudo gem install bettercap*.gem
```

Quick Start

Once you've installed bettercap, quickly get started with:

```
bettercap --help
```

The help menu will show you every available command line option and a few examples.

The following are the main options that determine the general behaviour of BetterCap, **these options are not mandatory**, in fact bettercap will automatically detect everything it needs in order to work, you just might need to use one or more of the following options to specify some custom behaviour in specific cases.

Examples

Attack specific targets:

`sudo bettercap -T 192.168.1.10,192.168.1.11`

Attack a specific target by its MAC address:

`sudo bettercap -T 01:23:45:67:89:10`

Attack a range of IP addresses:

`sudo bettercap -T 192.168.1.1-30`

Attack a specific subnet:

`sudo bettercap -T 192.168.1.1/24`

Randomize the interface MAC address during the attack:

`sudo bettercap --random-mac`

Options

`-I, --interface IFACE`

BetterCAP will automatically detect your default network interface and use it, if you want to make it use another interface (when you have more than one, let's say `eth0` and `wlan0`) you can use this option.

`--use-mac ADDRESS`

Change the interface MAC address to this value before performing the attack.

`--random-mac`

Change the interface MAC address to a random one before performing the attack.

`-G, --gateway ADDRESS`

The same goes for the gateway, either let bettercap automatically detect it or manually specify its address.

`-T, --target ADDRESS1,ADDRESS2`

If no specific target is given on the command line, bettercap will spoof every single address on the network. There are cases when you already know the IP or MAC address of your target(s), in such cases you can use this option.

`--ignore ADDRESS1,ADDRESS2`

Ignore these IP addresses if found while searching for targets.

`--no-discovery`

Do not actively search for hosts, just use the current ARP cache, default to `false`.

`--no-target-nbns`

Disable target NBNS hostname resolution.

`--packet-throttle NUMBER`

Number of seconds (can be a decimal number) to wait between each packet to be sent.

`--check-updates`

Will check if any update is available and then exit.

`-h, --help`

Display the available options.

Logging

These options determine how bettercap console logger is going to behave.

Examples

Save log output to the `out.log` file:

```
sudo bettercap --log out.log
```

Save log output to the `out.log` file and suppress terminal output:

```
sudo bettercap --log out.log --silent
```

Save log output to the `out-ts.log` file and enable timestamps for each line:

```
sudo bettercap --log-timestamp --log out-ts.log
```

Options

`-O, --log LOG_FILE`

Log all messages into a file, if not specified the log messages will be only print into the shell.

`--log-timestamp`

Enable logging with timestamps for each line, disabled by default.

`-D, --debug`

Enable debug logging, **it is good practice to use this option while reporting a bug** in order to have the full debug log of the program.

`--silent`

Suppress every message which is not an error or a warning, default to `false`.

Spoofing

As previously described in the introduction section, spoofing is the very hearth of every MITM attack. These options will determine which spoofing technique to use and how to use it.

BetterCap already includes an ARP spoofer (working both in full duplex and half duplex mode which is the default), a **DNS** spoofer and **the first, fully working and completely automatized ICMP DoubleDirect spoofer in the world**

Examples

Use the good old ARP spoofing:

`sudo bettercap` or `sudo bettercap -S ARP` or `sudo bettercap --spoofer ARP`

Use a *full duplex ICMP redirect* spoofing attack:

`sudo bettercap -S ICMP` or `sudo bettercap --spoofer ICMP`

Disable spoofing:

`sudo bettercap -S NONE` or `sudo bettercap --spoofer NONE` or `sudo bettercap --no-spoofing`

No dear 192.168.1.2, you won't connect to anything anymore :D

`sudo bettercap -T 192.168.1.2 --kill`

Options

`-S, --spoofer NAME`

Spoofer module to use, available: `ARP`, `ICMP`, `NONE` - default: `ARP`.

`--no-spoofing`

Disable spoofing, alias for `--spoofer NONE` / `-S NONE`.

`--kill`

Instead of forwarding packets, this switch will make targets connections to be killed.

`--full-duplex`

Enable full-duplex MITM, this will make bettercap attack both the target(s) and the router.