**UNDERSTANDING HTTPS AND**

**HOW TO BYPASS IT**

**Everything that we did so far will only work against HTTP pages**

**The reason why it works against HTTP is because as we have seen the data in HTTP is sent as plain text**

Thats why when we are the man in the middle we are able to read this text and if we wanted we are able to modify this text as we wish

**Obviously this was a problem and this problem was fixed in HTTPS**

As we know most websites use **HTTPS** because its a more secure version of **HTTP** and basically it adds an extra layer over **HTTP** which is where the **S** comes from

**So its a secure HTTP protocol and this extra layer will encrypt the plain text data that HTTP sends**

So if a person manages to become the man in the middle, they will be able to read this data but the data will be **gibberish**, it will not be readable

**HTTPS relies on TLS or SSL to encrypt the data**

**TLS == Transport Layer Security**

**SSL == Secure Sockets Layer**

**This is very difficult to break !**

**Therefore, in order to bypass this, the easiest method is to downgrade HTTPS connections to HTTP**

**!! Since we are the man in the middle, we can check if the target is requesting a HTTPS website and instead of giving him the HTTPS version of the website, we will give him the HTTP version !!**

This way the data will be sent in plain text and we will be able to read it

**To do this we will have to manually configure and use a tool called SSL Strip**

**Luckily BetterCAP has a caplet that will do all of this for us**

**The only problem is this caplet does not replace all HTTPS links to HTTP in the loaded pages**

So we are going to take **a modified caplet** to make sure its going to work as expected

**The file is called hstshijacked.zip**

We download it and extract it in Downloads in Kali, then we want to copy the folder and paste it in the correct location where **BetterCAP loads caplets from**

**To go to that location we can press Ctrl + L to open the Path Bar**

**Once we open it, we want to go to:**

usr/share/bettercap/caplets/

We can see we already have this caplet there but this caplet is buggy, it doesnt work as expected so we delete it and we paste the one we copied

Now its all good, we can use this caplet from **BetterCAP**

**But before we do that, we want to change the caplet we created earlier - the spoof caplet, and we want to modify one thing in it**

**We right click it and open it with LeafPad and what we want to modify is we want to add an option to the sniff**

**Before turning on the sniff we want to set an option before it:**

net.sniff.local true

**What this option will do is it will tell BetterCAP to sniff all data even if it thinks this data is local data**

**The reason we do this is because once we use the HTTPS bypass caplet, the data wil seem as if it is being sent from our computer**

**BetterCAP will think these passwords belong to our computer and it will not display it to us on screen**

Thats why we are setting it to **true** so that we can see al of the usernames and the passwords sent on the websites that we will downgrade from **HTTPS** to **HTTP**

**So we save it Ctrl + S and quit it, Ctrl + Q and we are ready to go and use this caplet**