Harry Pittar, Trent Lim, Levi Schimanski, Jada Mataroa

TLS SECURITY REPORT

What is the TLS attack?

A TLS attack is a type of man in the middle (MITM) attack or more commonly known as a three way handshake. The TLS protocol begins with a TLS handshake where each connection begins. This is where the technical underpinnings of TLS is established. A "handshake" is a description of how protocols establish a connection to HTTPS. This attack involves an unauthorized party to intercept the communication between two people or systems. This unauthorized party is set out to impersonate or eavesdrop what is being said between the two systems in hopes of gathering sensitive information that can be decrypted/encrypted. Usually this information is gathered for financial or personal information that can be sold to a third party. The MITM attacker also has Main in the middle browser (MITMB) privileges and can serve a webpage to the victim that has dangerous content. Using this browser, the attacker can trigger an HTTPS POST request to the target webserver which has the correct FTP commands. The attacker can redirect the request to an FTP server that has a certificate that is compatible with the request, then the browser accepts the handshake with the FTP server and sends the HTTP request as application data which can be used against the organisation. This browser can find the correct JavaScript by searching throughout the content and run the JavaScript within the context of the original request, ultimately completing the attack.

How could you replicate the attack? (detailed step-by-step instructions for anyone who want to replicate the attack easily).

Due to the risk this program can have on your computer you should use a fresh, up-to-date Ubuntu virtual computer for this attack. Once you have this you should follow the first 4 instructions in the README section of the alpaca-code found at https://github.com/RUB-NDS/alpaca-code/tree/master/testlab. When you are to installing docker from https://docs.docker.com/engine/install/ubuntu/ follow the instructions on how to setup the repository and install the Docker engine, ignoring the section on installing a specific version of Docker as this is not needed.

Once you have done this, you must download git in your home directory using the command 'sudo apt install git' we were not in the root at the time so we called sudo to do this. Then you must clone the alpaca code within Github using this command 'git clone https://github.com/RUB-NDS/alpaca-code.git' once you have done this you can check it is there by listing the files in your current directory using 'ls'.

Then you must re-enter the root and navigate to /home/<USER>/alpaca-code/testlabs/ as this is where setup.sh is located (do this using cd and then the above file path). Once you are here you must continue following steps 5 and 6 of the README. Then open up Firefox and within the security section of settings you must view the certificates and import the ca.crt certificate.

From here you can continue on to follow the FTPS section of the README. However in our case everytime you open something in firefox such as target.com, attacker.com/download or attacker.com/upload, you should stop the program in the first terminal screen using control c and re-run it again using the 'python3 main.py --proto FTP –attacker_ip 127.0.0.2 127.0.0.1

21' command. This is because the webpages didn't seem to load unless the program had only just been started and hadn't really done anything yet.

How do you know the attacks (upload and download) are successful? What evidences (logs etc) have you got?

For the upload, to know that it had worked we made scripts/show_vsftp_log.sh an executable and ran it in the terminal, which returned the following log, with no error messages, showing it had worked:

Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Referer: https://attacker.com/
Upgrade-Insecure-Requests:1
Sec-Fetch-Dest: document
Sec-Fetch-Dest: navigate
Sec-Fetch-Dest: cross-site

For the download, to know it worked we used it and were navigated to target.com with a pop-up window saying https://target.com, this is what was meant to happen and so we knew it had worked.

Failure/success experience during the etude:

While completing the etude we ran into many problems, initially we had trouble with installing and building Docker images. We solved this by installing the newest version of Ubuntu. We then ran into issues trying to display the FTP log, however we found out that there was a certificate problem. After trying to make it work for a while, we found out that the certificate within the <USER> directory was no longer the same as the one being used by the program and had to be recopied, then moved out from sitting within the root directory and lastly imported to the approved certificates within Firefox's settings. Once we had done this everything went smoothly until it came to loading the upload option in the attacker.com website on Firefox. To get it to load, we ended the program, then re-ran it, re-armed it and tried again, which worked. But when we reopened attacker.com and tried the download option it again stopped working, however following the exact same process as before allowed us to carry out the attack.