University of Liège
Faculty of Applied Science

Securing network with firewalls and NATs

Step 3: iptables rules

Maxime Meurisse
Valentin Vermeylen

Master in Civil Engineering
Academic year 2020-2021

1 Implementation of firewall rules

For this last part of the project, we based ourselves on the high level rules given in the solution of the previous part of the project.

We implemented the rules listed in this solution via `iptables`. For this, we encoded each rule in their respective firewall (FW1, FW1, FW3 and FW4) via a `config_FWX.sh` file (with `X` $\in [1, 4]$).

To make our life easier, each IP address is first stored in a variable. Then, each rule is implemented via `iptables` with a command of the form `iptables -A FORWARD [-s <ip> | -d <ip>] -p <proto> --dport <port> -j ACCEPT`.

All the rules have been implemented in the `FORWARD` chain because no requests exit or go directly to a firewall. We have also changed the default policy of each firewall to `DROP` and use the following command to put each firewall in a stateful mode: `iptables -A FORWARD -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT`. For the latter, we found that it is not always secured[1] to use this rule but for the sake of simplicity, we decided to let it as it is.

Finally, we implemented the NAT rules in firewall 1 (FW1), also via `iptables`. We have implemented the (static) rules given in the solution of part 2 in the `PREROUTING` chain. For outgoing traffic we used the `MASQUERADE` target in the `POSTROUTING` chain.

2 Implementation testing

In order to be sure that our firewall implementation is correct, we carried out tests via Netkit.

- We tried to reach LWEB from U1 and U2. It works with HTTP requests but not with HTTPS, as expected. We also tried to reach LWEB from U3 but it did not work, as expected.
- We tried to reach PWEB from U1, U2 and U3 with HTTP and HTTPS requests. It works for U1 and U2 but not for U3, as expected.
- We tried SSH connections: from U1 to U3, from Internet (DT) to U2 and from U3 to Internet. All these tests work, as expected. We also tried U2 to U3, but it did not work, as expected.
- We tried FTP connections from U1 and U2 to LWEB. It works for U2 but not for U1, as expected.
- We tried to access NFS server from U3. It works (and only for U3), as expected.
- We tried to use RSYNC with U1 (Joe's account) and U3 (Donald's account). It works, as expected.
- We test DNS servers by making HTTP(S) requests to domain names (instead of IP addresses) and it works, as expected. We also tried connections to external DNS servers by making HTTP(S) requests to domain names like `https://www.google.be/`. It works, as expected.

[1] see `https://gist.github.com/azlux/6a70bd38bb7c525ab26efe7e3a7ea8ac`

- Finally, we try mail systems by sending mails (from JOE to DONALD and vice-verso). It works, as expected.

It is difficult to envisage all possible cases, but this series of tests makes us confident about the validity of our implementation.

We have also tested cases that are not supposed to work: connecting via SSH to U2 from U1, connecting to a DNS server via U1 or U2, etc. Of course, we have not tested all possible cases (there would be too many). However, as we have set the default policy to DROP for all firewalls, apart from the cases we have allowed (and tested), we shouldn't have any unpleasant surprises (allowed cases that should not be allowed).

3 Conclusion

This project taught us how to configure firewalls and NATs for a large network.

We first started by drawing the network topology. This part was, in our opinion, the most difficult of the three. We had to pay attention to the position of each element in the network. It gave us a better understanding of the notion of zone and their importance (in order to classify them).

The second part consisted in listing the rules of each firewall and NAT. This part seemed easier to us than the previous one. However, it is necessary to check that all possible cases have been considered and listed so as not to leave possible flaws in the system.

Finally, we ended up implementing these rules via `iptables` and testing them via Netkit. The implementation was quite fast since it was sufficient to translate the handwritten rules into `iptables` commands. We were confronted with a few slight problems (notably concerning the stateful mode of the firewall), but we were able to correct them quickly and carry out tests correctly.

Accompanied by the expert G.I. JOE, this work allowed us to work for a large company and extend our knowledge in the computer networking field. Although we realised that the task is not as easy as it might seem, we are satisfied with our result. However, despite this, we are not sure that DONALD's company will last long; perhaps we should consider looking for new jobs...