Design

This script has been designed with extensibility in mind. It is clearly commented with sections for TCP, UDP code under each chain.

At the top we define the input rules for TCP and UDP individually and then the output rules. We label the chains:

* TCP\_INPUT\_RULES
* UDP\_INPUT\_RULES
* TCP\_OUTPUT\_RULES
* UDP\_OUTPUT\_RULES

The accounting is set up by implementing:

* SSH\_TRAFFIC
* WWW\_TRAFFIC
* OTHER\_TRAFFIC

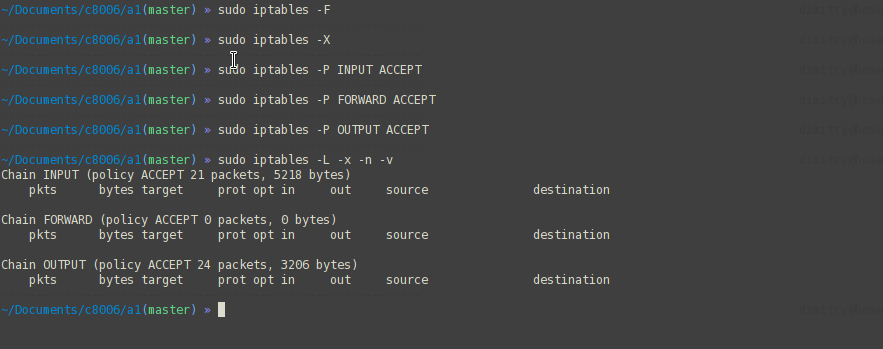
Chains which are called from our previously defined rules.

Testing

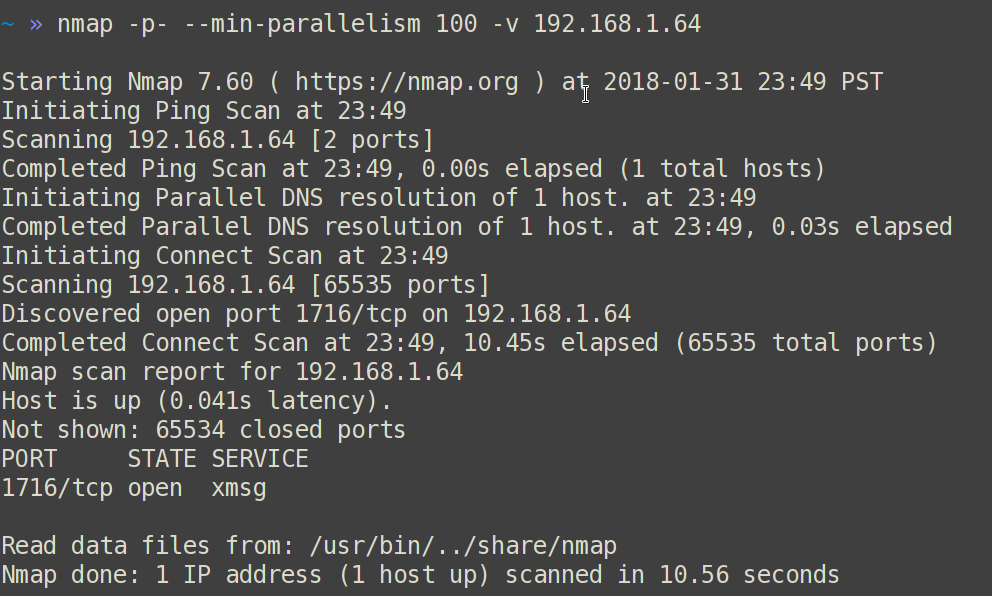
The testing procedure is simple. I will disable iptables on the firewall machine and scan it with nmap. Then I will enable iptables and run the scan again. This should show which ports are available and if my configuration is correct.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Description | Tool | Expectation | Result |
| 1 | Scan a machine with no firewall enabled. | Nmap, iptables | The machine will have either no ports open or many ports open depending on configuration. | 1716/tcp was open on the machine. Result is in line with expectations. |
| 2 | Scan machine with iptables script enabled. | Nmap, iptables | The machine will have web ports open, SSH port open as well as DHCP and DNS ports open. | Web ports were open. SSH ports detected once but never again. Error logs displayed sshd process crashing. It was assumed that the testing caused the crash. |
| 3 | Wireshark testing of web traffic. | Wireshark, iptables, web browser. | The firewall will block all web traffic on ports other than 80 and 443. Some sites may not load correctly. | In the included pcap file we can see the firewall block packet 1211 because it does not match the outbound rules. |

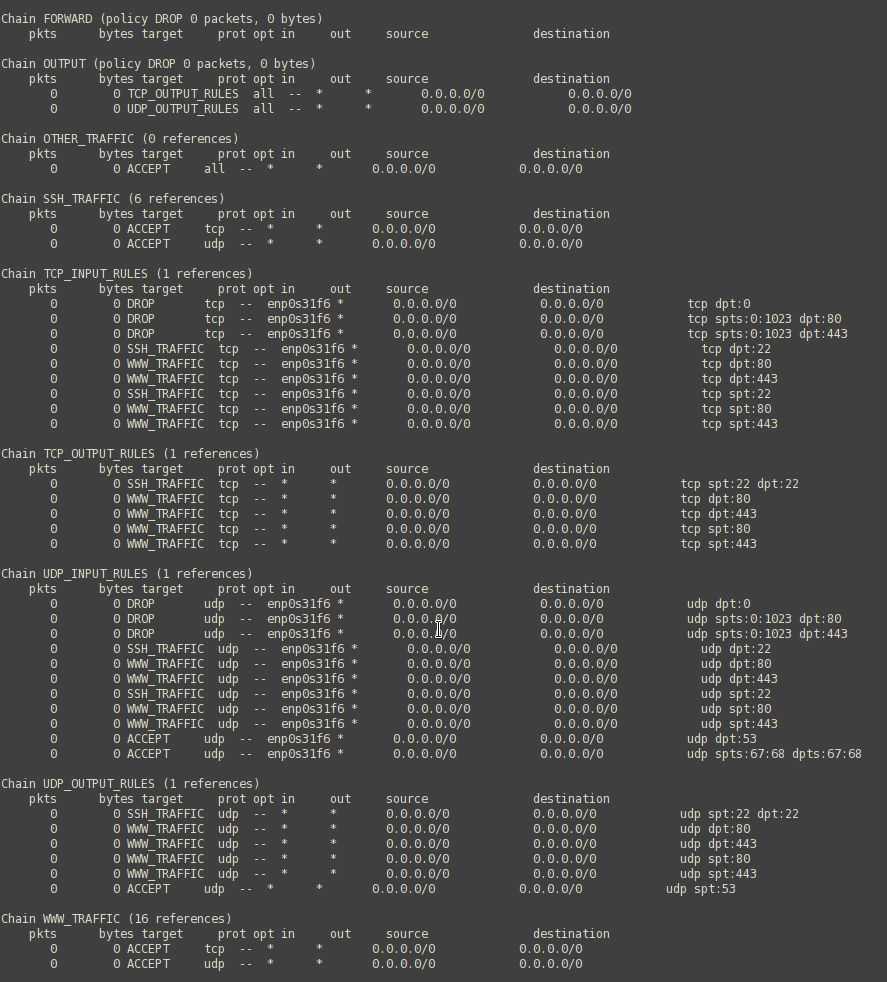
Disabling firewall:



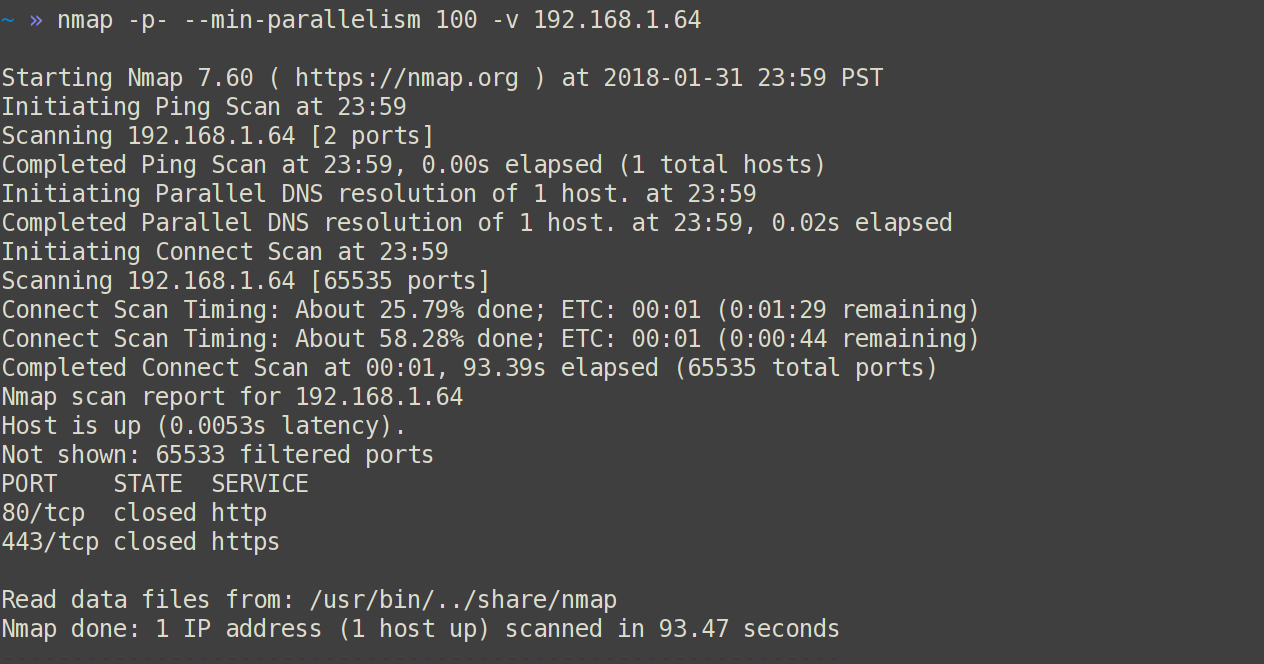
First scan:



Running firewall / port forwarding rules via “*sudo bash ipt.sh*”:



Here is the scan after running the script ssh is disallowed on my system:

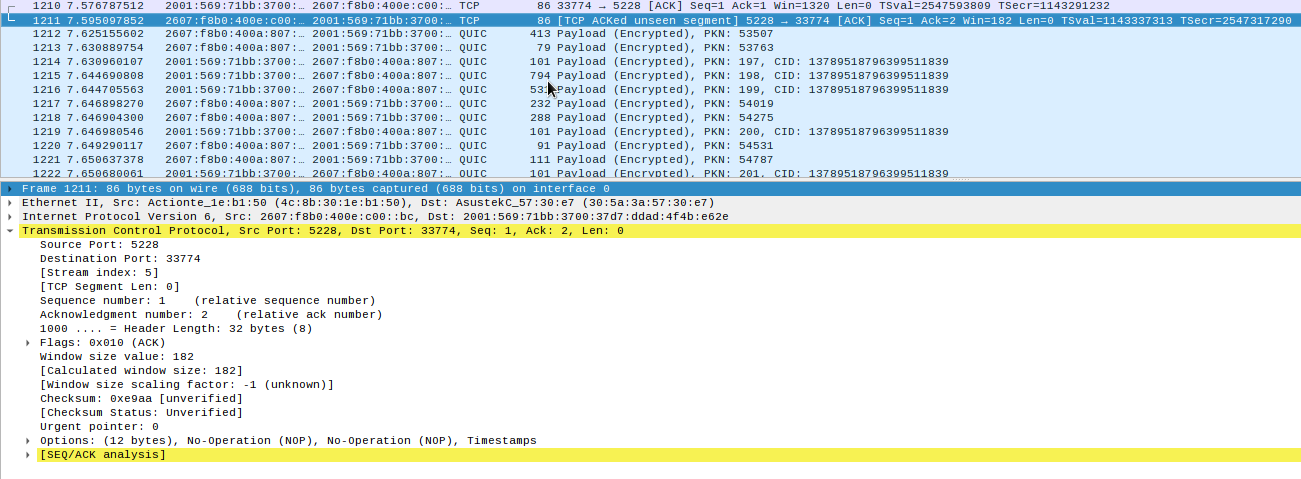


This shows that the config file allows WWW traffic through the machine. I was unable to test on a machine that has ssh correctly configured but the rules are configured inside the script.

Wireshark

The final test is done via Wireshark. Here I browse the web and see if there are any unexpected ports accessed.

In this capture we can see that a TCP packet #1211 from my machine to a server was dropped because it was not on port 443 or 80.



The capture file is included in the submitted package.