HPY 413
Assignment 6

Submission deadline: 15/12/2024

In this assignment, you will get familiar with iptables/ip6tables rules. Iptables/ip6tables is used to set up, maintain, and inspect the tables of IP packet filter rules in the Linux kernel. Several different tables may be defined. Each table contains a number of built-in chains and may also contain user-defined chains. Each chain is a list of rules which can match a set of packets. Each rule specifies what to do with a packet that matches. Additional information on iptables can be found at https://en.wikipedia.org/wiki/Iptables.

The assignment assumes background knowledge in networks and familiarity with bash scripting. Specifically, you will develop a bash script (named "firewall.sh"), that uses the linux command iptables/ip6tables in order to create a simple blocking mechanism that rejects packets coming from specific network domains or IP addresses. Information about the iptables/ip6tables command can be found in the appropriate man pages (i.e., man iptables(8) and ip6tables(8)). A Bash shell script, at its simplest form, is just a list of shell commands separated by newlines and can be used for running multiple commands together, customizing administrative tasks, performing task automation, etc. You can find a Bash scripting cheatsheet at https://devhints.io/bash#functions.

The firewall.sh script is responsible for generating a set of firewall rules that block access for specific network domain names or IPv4 and IPv6 addresses. You may use different network administration command-line tools (e.g., dig, host, nslookup) for querying DNS nameservers for information about host addresses if you want. It is possible for one domain name to have multiple corresponding IP addresses, as such you can use your favorite command-line tool (e.g., grep, awk, sed, etc.) to parse the results of the DNS query. The iptables(8), ip6tables(8), dig(1), host(1), nslookup(1), grep(1), awk(1) and sed(1) are the man pages for the aforementioned command-line tools.

Your firewall.sh script is expected to do the following:

1. Configure rules based on the domain names and IPs (Ipv4 and IPv6) of "config.txt" file.
2. Save rules to "rulesV4" and "rulesV6" files (these files do not have a filename extension).
3. Load rules from "rulesV4" and "rulesV6" files .
4. Reset all rules to default settings (i.e. accept all).
5. List all current rules.

You are given the following files. You are **NOT** allowed to change the naming scheme of the given files.

Files:

- firewall.sh
  - The corpus of the firewall.sh script.
- config.txt
  - Contains network domain names and IP addresses .
- rulesV4 and rulesV6
  - Empty files used to load/save the iptables/ip6tables rules.

Tool Specification

The provided corpus already implements the options/arguments of your tool. Again you are **NOT** allowed to change the naming scheme of the options. Your script will receive the following arguments from the command line upon execution.

Options:

-config	Configure adblock rules based on the domain names and IPs of config.txt' file
-save	Save rules to rulesV4 and rulesV6 files.
-load	Load rules from rulesV4 and rulesV6 files.
-list	List current rules.
-reset	Reset rules to default settings (i.e. accept all).
-help	Display help and exit.

Hints

1. iptables/ip6tables commands require root privileges. Run your script with sudo.
2. You can make your script executable with chmod +x adblock.sh
3. Use the -j REJECT option to reject the connection.
4. DNS queries for many domains can take some time. You can force a command (or more) to run in the background (e.g., host google.com &) and then use wait to suspend the execution until the subprocesses have finished.

5. The config.txt file contains some popular advertising domains. Feel free to add more or remove some. Your script should be able to handle hundreds of domain names easily.

Question

1. After configuring the firewall rules, test your script by visiting your favorite websites without any other adblocking mechanism (e.g., adblock browser extensions). Can you see ads? Do they load? Some ads persist, why?

Notes

1. Your implementation must run in a linux based machine.
2. The naming scheme of the given files must remain as-is.
3. The options defined in the "Tool specification" section must remain as-is.
4. You MUST create a README that contains:

    The team number, names/surnames, AMs, a short description (1-2 lines) of your implementation and answer to the question.
5. You must submit the following files: firewall.sh, README.
6. You should place all these files in a folder named <teamX>_assign6 and then compress it as a .zip file. For example, if your team number is 4 is the folder should be named team4_assign6 you should commit team4_assign6.zip.
7. GPT generated code is forbidden. Submissions with GPT created code will be rejected.