Comp 8006 Assignment 3

Mat Siwoski
Shane Spoor

Notes:

Running the application:

To run the application, navigate to the folder that the script was installed to and type:

sudo python3 a3_secure_log.py

In the event that the file does not run:

chmod +x a3_secure_log.py

chmod +x a3_secure_log.py

to ensure that the file can be executed.

Requirements:

The application has to follow the following requirements:

- Design, implement and test an application that will monitor the /var/log/secure file and detect password guessing attempts and then use iptables to block that IP.
- Your application will get user specified parameters (see constraints) and then continuously monitor the log file specified.
- As soon as the monitor detects that the number of attempts from a particular IP has gone over a user-specified threshold, it will generate a rule to block that IP.
- If the user has specified a time limit for a block, your application will flush the rule from Firewall rule set upon expiration of the block time limit.
- Design a test procedure that will test your application under a variety of conditions. For example, how will you handle a situation when an attacker sends a slow scan of your system, meaning several password guessing attempts, but spaced far enough apart in time so that your application will miss the attack.

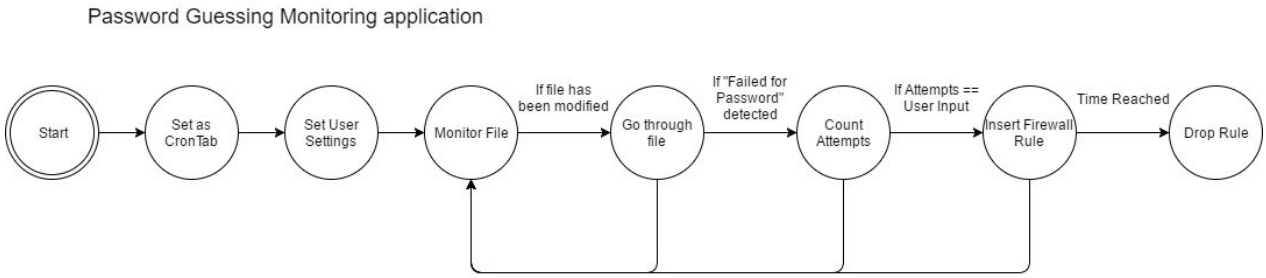
Constraints:

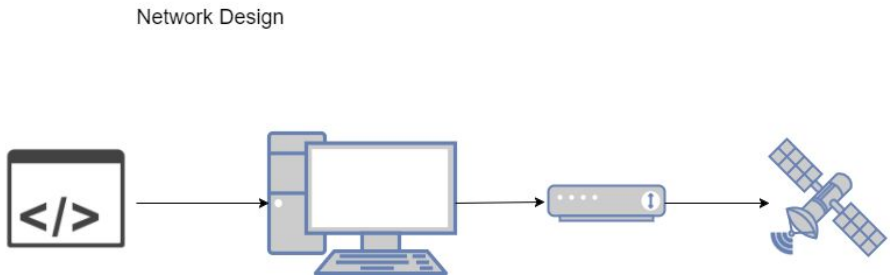
The application has to follow the following constraints:

- The application will be implemented using any scripting or programming language of your choice.
- The Firewall rules will be implemented using Netfilter.
- Your application will be activated through the crontab.
- Your application will obtain user input for the following parameters: The number of attempts before blocking the IP
    - The time limit for blocking the IP (Optional – bonus). The default setting will be block indefinitely.
    - Monitor a log file of user's choice (Optional - bonus).Keep in mind that different log files have different formats.

Design:

Design Diagrams





Pseudocode:

```
check_for_failed_password
        Go through each line
                If “Failed password for” is found
                        Send to record_operations
                Else
                        Continue

Scan_var_log
        If mod_time is equal
                Continue
        Else
                If Open File
                        Read Lines
                        Send to check_for_failed_password
                Else
                        Send Error

Main
        scan_var_log
        exit()
```

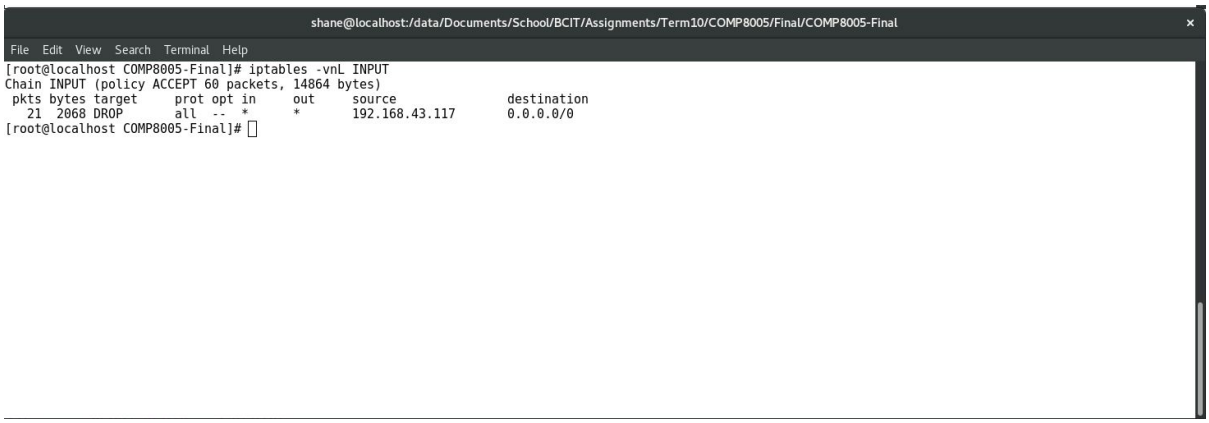
record_operations
    Split data for date
    Split data for IP, Port, User & Connection
    Check to see if user has reached threshold,
        if true, add iptables rules and block the user

if(__main__)
    If ubuntu
        Use /var/log/auth.log
    Else
        Use /var/log/secure

Test Results:

1.) Application in Crontab

2.) Firewall rule inserted



3.) Application detected failed password in file

4.) Blocked access to User