COMP 8006 Assignment 3 Design

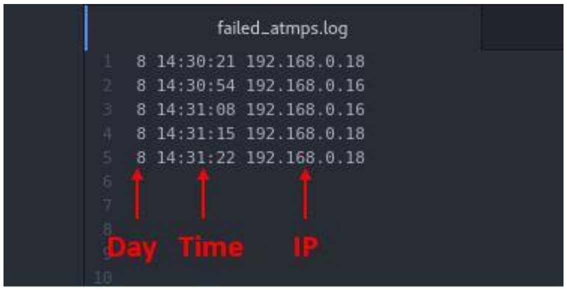
Alex Zielinski

03/08/2018

Contents

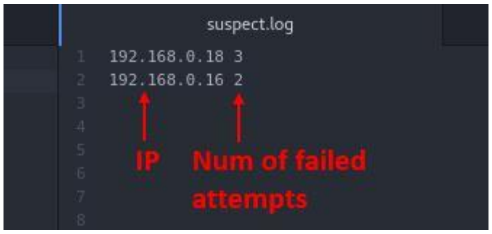
How it Works

When the intrusion prevention system (IPS) script is ran it immediately starts monitoring the ***/var/log/secure*** file for any signs of a failed SSH password event. The script makes use of ***grep*** to parse the secure log file for any instances of the string “failed password” and then uses ***awk*** to extract the day, time and IP of the failed attempt to a separate log file called ***failed_atmps.log***. Essentially the failed_atmps.log file holds a list of all the IP’s that have entered an incorrect password during an attempt to start an SSH session (as seen in the screenshot below).



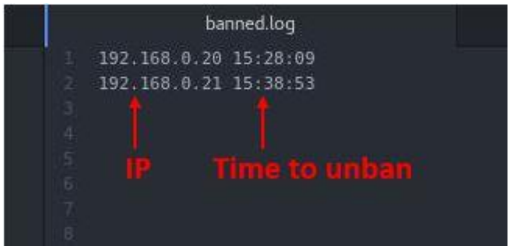
```
8 14:30:21 192.168.0.18
8 14:30:54 192.168.0.16
8 14:31:08 192.168.0.16
8 14:31:15 192.168.0.18
8 14:31:22 192.168.0.18
```

Next the script checks each entry in the failed_atmps.log file to see if it is a new failed password event or if it is an already documented event by comparing the day and time of the log entry to the current date. Any new log entries that have not yet been documented are then added to the ***suspect.log*** file. The suspect.log file contains a list of all the IP’s that have a failed SSH password attempt along with the number of failed attempts (as seen in the screenshot below).



```
192.168.0.18 3
192.168.0.16 2
```

Once the number of failed attempts made by an IP reaches the user specified *max number of attempts,* then that IP must be banned. As a result, a rule within iptables is set to drop any traffic from and to the banned IP, the IP’s log entry within the suspect.log file is removed, and a log entry is appended to the ***banned.log*** file containing the banned IP and the ban expiry time as set by the user (as seen in the screenshot below).



```
192.168.0.20 15:28:09
192.168.0.21 15:38:53
```

The script will continuously monitor the banned.log file to check if it is time to un-ban an IP. In such an event the rule that blocks the IP in iptables is removed and the log entry within the banned.log file is also removed.

User Guide

Running the Intrusion Prevention System

The IPS takes four command line arguments. The usage is as follows:

```
./ips.sh <NUM OF ATTEMPTS> <HOUR> <MIN> <SECONDS>
```

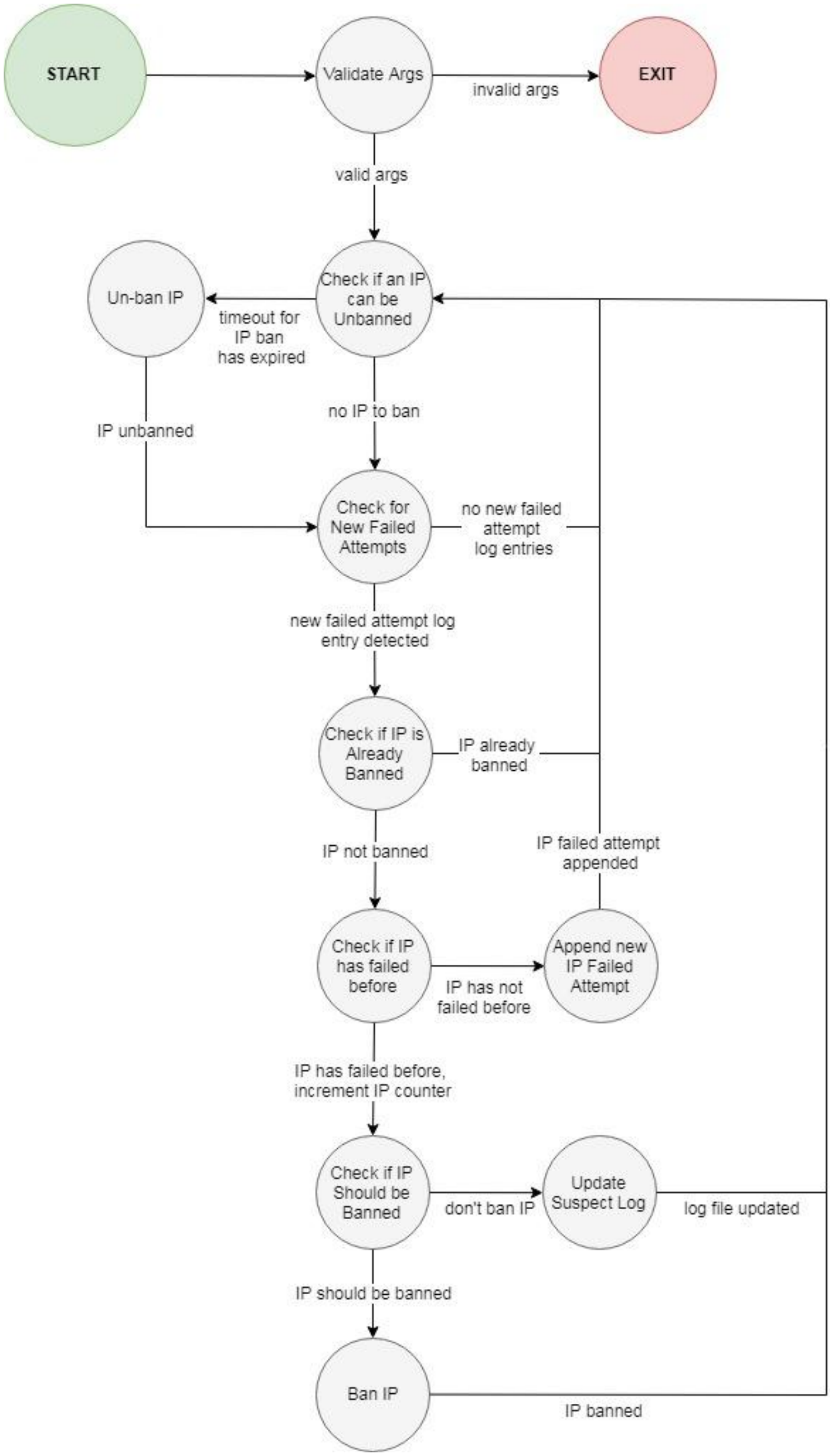
The first argument specifies how many times an IP can fail at an SSH password attempt before it becomes blocked. The last three arguments specify for how long an IP should be banned for. So, for example, if the user would like for an IP to be blocked after 5 failed attempts, and the user would like for that IP to be blocked for 2 hours, 5 mins and 15 seconds, then the command would look as follow:

```
./ips.sh 5 2 5 15
```

Project folder directory listings

```
/src
   |------ ips.sh
   |------ tmp
   |------ failed_atmps.log
   |------ suspect.log
   |------ banned.log
Design Doc
Testing Doc
```

Finite State Machines


START
Validate Args
invalid args
EXIT
valid args
Un-ban IP
Check if an IP can be Unbanned
timeout for IP ban has expired
IP unbanned
no IP to ban
Check for New Failed Attempts
no new failed attempt log entries
new failed attempt log entry detected
Check if IP is Already Banned
IP already banned
IP not banned
IP failed attempt appended
Check if IP has failed before
Append new IP Failed Attempt
IP has not failed before
IP has failed before, increment IP counter
Check if IP Should be Banned
Update Suspect Log
don't ban IP
log file updated
IP should be banned
Ban IP
IP banned

Pseudo Code

Validate CMD ARGs

```
Check if all four ARGs have been provided
If not, then EXIT and print usage statement, otherwise go to Check if IP can be Unbanned
```

Check if IP can be Unbanned

```
Parse the banned log file line by line
Extract the timeout expiry time for each banned IP
Get the current time
Check if the current time is equal to or greater than the timeout expiry time
If it is then go to Un-ban IP, otherwise go to Check for New Failed Attempts
```

Un-ban IP

```
Remove iptables rule that bans the IP
Remove the IP entry from the banned.log file
Go to Check for New Failed Attempts
```

Check for New Failed Attempts

```
Extract all failed SSH password attempts from /var/log/secure into a tmp file
Extract day, time and IP of the failed attempt log entry from tmp into failed_atmps.log
Parse the failed_atmps.log file line by line
Extract the day of the log entry
Check if the log entry is from today
If not, then go to Check if IP can be Unbanned, otherwise extract the time of the log entry
Check if the log entry is a new entry
If not, then go to Check if IP can be Unbanned, otherwise go to Check if IP is Already banned
```

Check if IP is Already Banned

```
Parse the banned log file line by line
Extract the IP from the files current line
Check if the IP from the file matches the IP from the new failed attempt log entry
If they are the same, then go to Check if IP can be Unbanned
otherwise go to Check if IP has Failed Before
```

Check if IP has Failed Before

```
Parse the suspect log file line by line
Extract the IP from the files current line
Check if the IP from the file matches the IP from the new failed attempt log entry
If they are the same, increment the IP fail counter and go to Check if IP should be banned
Otherwise go to Append New IP Failed Attempt
```

Append New IP Failed Attempt

```
Append to the end of the suspect.log file the IP and set the initial counter to 1
Go to Check if IP can be Unbanned
```

Check if IP Should be Banned

Extract the current IP’s fail counter
Check if it is equal to the user defined MAX NUMBER OF FAILS number
If they are equal, then go to **Ban IP**, otherwise go to **Update Suspect Log File**

Ban IP

Add iptables rule to ban current IP
Calculate the IP’s timeout expiry time (based on the user provided CMD ARGs)
Append the IP and its timeout expiry time to the banned.log file
Remove IP entry from suspect.log file
Go to **Check if IP can be Unbanned**

Update Suspect Log File

Increment the current IP’s failed counter within the suspect.log file
Go to **Check if IP can be Unbanned**