

Log processing

Drilling down by “whittling away”

Key Concept

The messages that occur the least are probably more important than those that occur all the time.

The first step is to identify the messages that occur most often. To do this, you would look through the log file(s) manually, and make note of the messages that you do not consider important. How you would do this depends of course on the application.

For example, in the grep command below we’ve filtered out the following strings:

```
[xxx examples from the log file]
```

```
pam_unix      - information about login/logout
publickey     - successful login with an SSH public key
Invalid       - an invalid login attempt (more on this below)
disconnect    - someone has disconnected
sudo          - use of the sudo command
BREAK-IN ATTEMPT - in fact, simply a warning about a mismatch between forward
                  and reverse DNS
Timeout       - a connection terminated due to timeout
Bad protocol  - warnings about protocol errors
identification - ...
```

If we were to filter all the above out of our search, we would run:

```
$ egrep -v '(pam_unix|publickey|Invalid|disconnect|sudo|BREAK-IN ATTEMPT|Timeout|Bad protocol|identification)
```

... this would also strip away the first 16 characters (the timestamp) so it would be easier to sort/compare messages based on content similarity.

(Explain drilling down to “rare” messages, and then deciding to add those to the “interesting” list, which we will add those to swatch/tenshi)

2. Identify targeted attacks/scanning attempts on your system

In the step above, we excluded the ‘Invalid user’ message as it occurred very often, and was not immediately of interest to us. But let’s look closer at a typical message now:

```
Aug 12 09:48:14 nsrsc sshd[20111]: Invalid user oracle from 106.186.28.45
--- --  -----  ---  -----  -----  ---  -----  ---  -----
1   2   3       4       5       6       7       8       9       10
```

Which IPs are most actively scanning?

```
$ grep 'Invalid' auth.log | awk '{ print $10 }' | sort | uniq -c | sort -n -k 1
```

You may want to consider tools like fail2ban:

Which username?

```
$ grep 'Invalid' auth.log | awk '{ print $8 }' | sort | uniq -c | sort -n -k 1
```

If you see, in the list of usernames, a username that is known/exists on your system (a real user, not a generic account like root/admin/...), then you should start to worry, or at least pay attention! You may also want to send email to the abuse contact listed for the containing netblock for this IP.