

Viewing TTY Logs with Lighttpd

This section provides the steps to convert the ttylogs captured by the DShield sensor into an HTML formatted document that can be accessed from Kibana interface.

Enable tty logging in each DShield sensors:

```
guy@picollector: sudo vi /srv/cowrie/cowrie.cfg
```

Search for `ttylog = false` change to: `ttylog = true`

Default ttylog → `/srv/cowrie/var/lib/cowrie/tty`

```
$ sudo systemctl restart isc-agent
```

```
$ sudo /srv/dshield/status.sh
```

Install txt2html to convert the ttylogs to an html format to have to ability to access them from ELK

```
$ sudo apt-get install txt2html
```

Install lighttpd in the ELK server

```
$ sudo apt-get install lighttpd
```

```
$ sudo systemctl enable lighttpd
```

```
$ sudo systemctl start lighttpd
```

Log location: `/var/www/html`

```
$ ls -la /var/www/html
```

```
$ sudo chown -R guy:guy /var/www/html
```

```
$ ls -l /var/www
```

Confirm the webserver is accessible, that your account is now the owner of the html directory and running before moving the default index out of the default directory

Using browser access: `http://elk`

```
$ mv /var/www/html/index.lighttpd.html .
```

Configure TTYLog in Kibana

Management -> Kibana -> Data Views -> cowrie*

Edit TTYLog and change the IP 192.168.25.231 to your ELK server IP or Name

URL template

```
http://192.168.25.231/{{rawValue}}.html
```

In ELK server, create SSH Shared Keys and don't put a password:

```
$ ssh-keygen
```

Copy id_rsa.pub over to each sensor(s). Likely the easiest way to copy the public key over might be to scp from DShield sensor.

```
$ scp guy@192.168.25.231:/home/guy/.ssh/id_rsa.pub .
```

```
$ cat id_rsa.pub >> .ssh/authorized_keys
```

```
$ rm id_rsa.pub
```

Test SSH between the ELK server to DShield sensor

```
$ ssh -l guy 192.168.25.165 -p 12222
```

Confirm the authentication to allow shared keys to work:

```
The authenticity of host '[192.168.25.165]:12222 ([192.168.25.165]:12222)' can't be established.
```

Next time you ssh it will log the client automatically in the DShield sensor. This is important for the ELK server to transfer the logs hourly.

Setup DShield Sensor for TTYLogs to HTML Format

There are 2 scripts provided for the sensor

- replayttylog.sh → Used when setting up TTYLogs parsing for the first time
- ttylog.sh → Runs on a hourly cronjob

If the sensor has been running for a while, un replayttylog.sh until it is completed. This could take a while depending on the number of logs is currently stored. The converted hashes will be stored in the home directory in ~/ttylog.

```
$ mkdir scripts
```

```
$ cd scripts
```

```
$ curl https://raw.githubusercontent.com/bruneaug/DShield-SIEM/main/sensor_scripts/replayttylog.sh -o replayttylog.sh
```

```
$ curl https://raw.githubusercontent.com/bruneaug/DSHield-SIEM/main/sensor_scripts/ttylog.sh -o ttylog.sh
```

```
$ chmod 755 *.sh
```

```
$ ./replayttylog.sh
```

When the script finish to run, it is time to manually transfer the logs to the Elastic server.

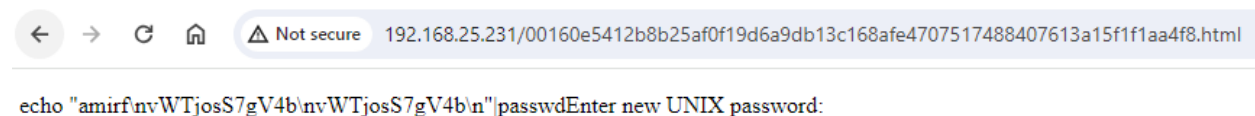
Setup Elastic Server for TTYLogs Transfer

The script has completed the log conversion to HTML, the Elastic server should be ready to receive the logs (we assume the webserver is setup as per above) and manually transfer them via scp to ELK:

```
$ scp -P 12222 guy@192.168.25.165:~/ttylog/* /var/www/html
```

```
guy@ubuntu:~$ scp -P 12222 guy@192.168.25.165:~/ttylog/* /var/www/html
00160e5412b8b25af0f19d6a9db13c168afe4707517488407613a15f1flaa4f8.html 100% 348 298.8KB/s 00:00
024b1c3004a1c415b7610f00fe3f49f6dd7c24b3439485f802aacd7634142317.html 100% 349 422.4KB/s 00:00
0367fab77720eb8e9a4792fe3e638f204609c447d23f9f7a78bdf67ad58084ad.html 100% 312 377.8KB/s 00:00
```

You can also confirm that you can view the html logs by taking one of the html files:



← → ↻ 🏠 ⚠ Not secure 192.168.25.231/00160e5412b8b25af0f19d6a9db13c168afe4707517488407613a15f1flaa4f8.html

echo "amirf\n\nWTjosS7gV4b\n\nWTjosS7gV4b\n"|passwdEnter new UNIX password:

The last step is to enable the hourly cronjob on the server to convert and transfer the TTYLogs over to the web server. First is to download the ELK server script to the elastic server:

```
$ cd scripts
```

```
$ mv ~/DSHield-SIEM/parsing_tty.sh .
```

```
$ chmod 755 parsing_tty.sh
```

Edit the parsing_tty.sh script and change the IP address 192.168.25.105 to the IP address of the DShield sensor and save the change:

```
$ vi parsing_tty.sh
```

```
$ crontab -e
```

```
1 * * * * ~/scripts/parsing_tty.sh > /dev/null 2>1&
```

This complete converting to HTML the ttylogs and having them available from the ELK server.

Accessing TTYLogs from Kibana

In Kibana, lookf for the Top 10 TTY Logs summary and just select TTYLog to go to the webserver page containing the log.

Top 10 TTY Logs	TTYLog	Total
2399d04d930c2f8fa71	TTYLog	10
e3b0c44298fc1c149af1	TTYLog	9
01ba4719c80b6fe911k	TTYLog	6
6fa4c8ac58e7a1d947d	TTYLog	3
28ba533b0f3c4df63d6	TTYLog	2

This is an example of the output:

```

<  →  ↻  🏠  🔒  https://elastic/2399d04d930c2f8fa713ec76aac747c19befe  📄  ☆  📧  ⬇️  📖  🌐  📌  ☰
The programs included with the Debian GNU/Linux system are free software; the exact distribution terms for each program are described
in the individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.
[4hadmin@inquisitors:~$ sh
shell
enable
system
ping:sh
kill %%%1

sh
[4l[4hadmin@inquisitors:~$ shell
-bash: shell: command not found
admin@inquisitors:~$ enable
[4lenable .
enable :
enable [
enable alias

```