



Installation **and Usage Guide**

Release **2.4 May 2016**

Table of contents

1	Overview	1
2	Getting Started	2
2.1	Installation	2
2.2	Appliance Import	3
2.2.1	Virtualbox.....	3
2.2.2	VMware.....	4
2.2.3	Logging in to the appliance.....	5
3	Configuration.....	6
3.1	Default Paswords.....	6
3.2	Static IP and hosts file setup	6
3.2.1	Host File Setup.....	6
3.3	Roles Explained	8
3.4	Scripting Role Assignment	8
4	Deployment	9
4.1	Minimum 2 Node Installation instructions.....	9
4.2	Corporate 4 Node Installation Instructions	10
5	Accessing Web Interfaces and Dashboards	12
5.1	Kibana.....	12
5.2	Incident Response	13
5.3	Graylog Streams & Alerts.....	14
5.4	Graylog TIME zone.....	14
5.5	Functionality Check.....	15
5.6	SSL Configuration	18
5.7	Cluster Health and Monitoring	19
6	Logstash Templates.....	21
7	Installing Agents	23
7.1	NXLOG SIEM Agents for Microsoft Hosts	23
7.1.1	Instructions for agent installation	23
7.2	Filebeat SIEM agents for Linux or Apache	26
7.3	OSSEC HIDS Agents for Windows Hosts.....	29
7.4	SYSLOG.....	32
8	Inputs.....	33
8.1	Logstash.....	33
8.2	Troubleshooting SYSLOGS	37
8.3	Troubleshooting Cisco ASA syslogs.....	39
8.4	Graylog	41
9	Alerting	43
9.1	Graylog	43
9.2	Elastalert	47
9.3	Logstash.....	47
10	OSINT.....	48
10.1	Critical-Stack-Intel.....	48
10.2	Malicious IP.....	50



11 HIDS.....	51
11.1 Rulesets	51
11.2 Management	51
12 Incident Response.....	52
12.1 Administration.....	52
12.2 Usage.....	52
12.3 Automated Ticketing	54
13 Frequently Asked Questions	55
13.1 Configuration / Installation	55
13.2 Backup/Scaling.....	55
13.3 Troubleshooting.....	55
14 Changing Passwords.....	57

TABLE OF FIGURES

Figure 1 - Siemonster Hash Checksum	2
Figure 2 - Windows MD5 & SHA Checksum Utility	2
Figure 3 - Virtual Box import.....	3
Figure 4 - Naming your image.....	3
Figure 5 - Importing your ova	3
Figure 6 - Vmware 12 Workstation Opening a VM	4
Figure 7 - VMware import	4
Figure 8 - Logging on to SIEMonster	5
Figure 9 - Default Passwords to be changes after build	6
Figure 10 - Default Server IP allocation	6
Figure 11 - Script install Proteus Single node (Database).....	9
Figure 12 - Proteus scripting complete - Reboot required.....	9
Figure 13 - Capricorn image role script.....	9
Figure 14 - Capricorn role complete, reboot for name change to take effect.....	9
Figure 15 - Kraken Installation First of the 4 nodes.....	10
Figure 16 - Tiamat Installation Database Node 2	10
Figure 17 - Proteus multi node installer	11
Figure 18 - Capricorn Installer Script.....	11
Figure 19 - Capricorn installed reboot for name change	11
Figure 20 - SIEMonster Default page in Kibana.....	12
Figure 21 - Incident Response normal user start-up screen.....	13
Figure 22 - Admin Access Incident Response.....	13
Figure 23 - Capricorn Graylog logon verification	14
Figure 24 - Graylog search criteria verification	15
Figure 25 - Example Histogram to ensure Graylog functioning.....	15
Figure 26 - Siemonster Kibana verification.....	16
Figure 27 - Verification Failed logon appears.....	16
Figure 28 - Siemonster OSSEC Alerts verification.....	16
Figure 29 - OSSEC Signature Alerts	17
Figure 30 - Raw alerts view	17
Figure 31 - Health Check view.....	19
Figure 32 - Cluster Health checking	19
Figure 33 - Curl output.....	20
Figure 34 - SSL Configuration	25
Figure 35 - Filebeat path modification	26
Figure 36 - Commented out Elasticsearch	27
Figure 37 - Proteus IP inserted.....	27
Figure 38 - SSL Transport follow picture for correct path	27
Figure 39 - Uncommented SSL ad path	27
Figure 40 - Working connection.....	28
Figure 41 - OSSEC HIDS Menu	29
Figure 42 - Setting up the OSSEC agent IP and name	29
Figure 43 - Capricorn IP and Key	30
Figure 44 - List the agent and finish.....	31
Figure 45 - Port change for Syslog.....	32
Figure 47 - Kibana Windows Events	34
Figure 48 - Logstash Index	35
Figure 49 - Visualization of the data.....	35
Figure 50 - Dashboard Configuration.....	36
Figure 51 - Open and save for Dashboards	36
Figure 52 - Syslog Checker	37

Figure 53 - Syslog received successfully	38
Figure 54 - Time zone mods.....	39
Figure 55 - Ruby troubleshooting.....	40
Figure 56 - Syslogs received command line access	40
Figure 57 - Graylog Input	41
Figure 58 - Gelf setup.....	41
Figure 59 - Gelf Capricorn and port details.....	42
Figure 60 - Stream Creation.....	43
Figure 61 - Windows event stream Edit Rules.....	43
Figure 62 - Input previously configured in Graylog.....	43
Figure 63 - Adding a Rule for alerting in the stream	44
Figure 64 - Confirmation of stream rule.....	44
Figure 65 - Managed alert in Graylog	44
Figure 66 - Setting up alert condition	45
Figure 67 - Slack Alarm callback.....	45
Figure 68 - Webhook URL in Graylog alerting	45
Figure 69 - Alarm call back result once configured	46
Figure 70 - Sensor Creation	48
Figure 71 - Demo Sensor.....	48
Figure 72 - Adding OSINT Feeds.....	48
Figure 73 - Adding Feeds to OSINT	49
Figure 74 - Sample Subscribe confirmation	49
Figure 75 - Selecting your previous demo-sensor	49
Figure 76 - Icon and Signature.....	49
Figure 77 - Critical Stack API download	50
Figure 78 - FIR New event creation.....	53
Figure 79 - FIR Events	53
Figure 80 - Health Check view.....	55
Figure 81 - Cluster Health checking	56
Figure 82 - Curl output.....	56

1 OVERVIEW

Welcome to the **SIEMonster** documentation.

SIEMonster is a collection of Open Source security event management tools in a single package. You can be up and running in 30 minutes with live data on a dashboard without any of the associated configuration headaches.

Be inspired to create dashboards, alert & get your event data into a SIEM the way you want it. Examples are provided for every step of the way to speed up the process and be in the position to monitor rather than develop. Video demonstrations are available on www.siemonster.com if you hate reading.

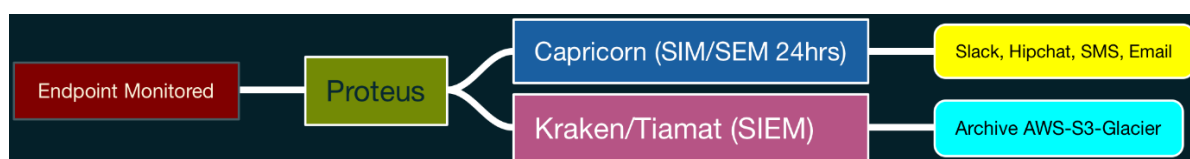
The single virtual image can be built into a 2 node instance for small companies or to try out or turn it into a 4 node cluster for an enterprise with scaling capabilities. The appliance is assigned a role via an assignment script which activates the required services and functions relevant to that role. If you want to try it out use the Proteus/Capricorn 2 node instance. If you want to install it into production use the Proteus/Capricorn and Kraken/Tiamat 4 node instance.

The default usernames and passwords for accessing **SIEMonster** are shown in the Configuration Section These passwords must be changed after installation as per Chapter 14.

SIEMonster has been built to run on VMware Workstation, Virtualbox & ESX server and an Amazon AWS cloud formation auto build is also available for Amazon customers.

SIEMonster minimum requirement for system memory is 4GB on each node as pre-set in the OVA, recommended allocation is 16GB for both Proteus and Capricorn, & 32GB for Kraken & Tiamat.

SIEMonster overview



2 GETTING STARTED

2.1 INSTALLATION

SIEMonster OVA Appliance Release 1.01 can be downloaded from

<http://releases.siemonster.com/siemonster-1.01.ova>

Hash	Checksum
MD5	02F3153A64CEB849D06A1245A798338B
SHA-1	74ABE482474FF10EA99054F1D47995AF5FD1F9E2
SHA-256	561F75C5DDE94C44AB89FF04B280FC15B556E03BE7293EA49BA893C1FEDA AFC8
SHA-512	2DEA480C34B6E4F4EBF1DB4F8FD6E018EB1178C926A2D6A8AB404A5FC6D 28C23917AC024D71A60DA3751F3B2453998939F248B73A0A31A8B6EC9CA95 5D9C97A3

Figure 1 - Siemonster Hash Checksum

You can verify the hash using a variety of hashing utilities or use the following

Windows - MD5 & SHA Checksum Utility

Linux - Sha256sum

Mac - Shasum

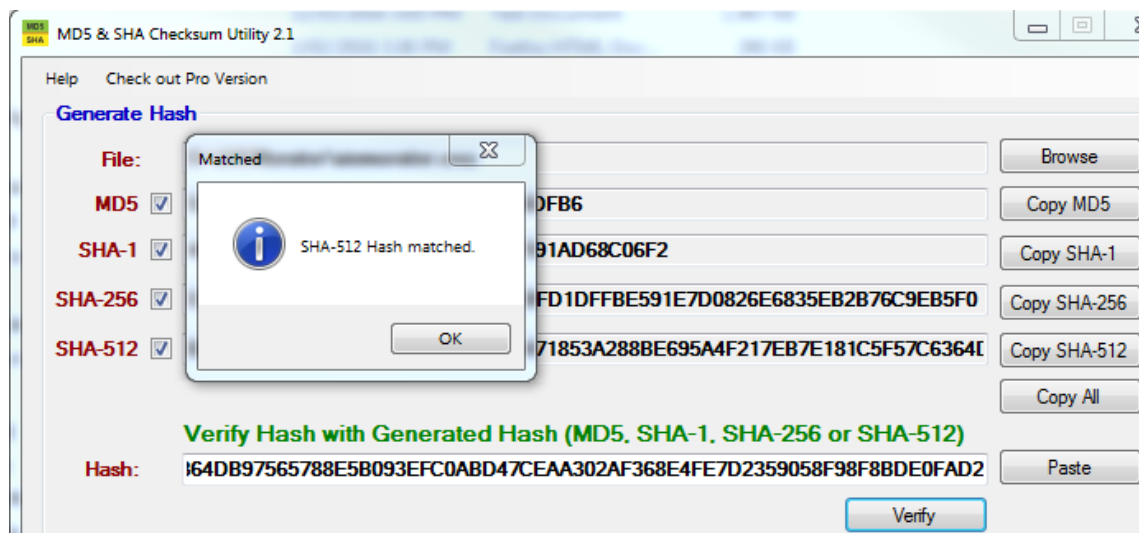


Figure 2 - Windows MD5 & SHA Checksum Utility

2.2 APPLIANCE IMPORT

2.2.1 Virtualbox

- Using Virtualbox select File – Import Appliance

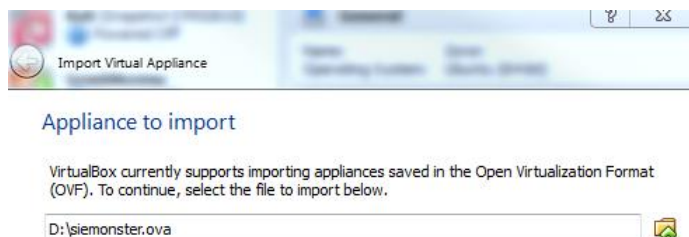


Figure 3 - Virtual Box import

- Change the Name to suit the role

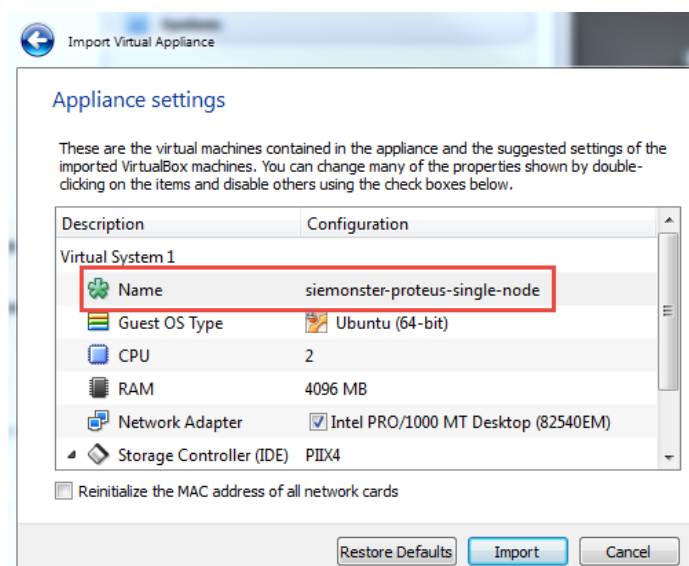


Figure 4 - Naming your image

- Import

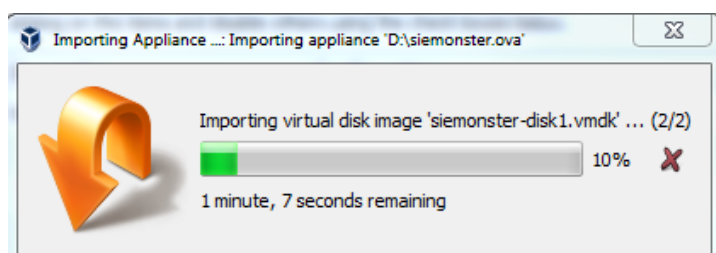


Figure 5 - Importing your ova

Once the appliance has started up, login as user siemonster with password siemonster.

2.2.2 VMware

- Using VMware Open a Virtual Machine, rename to the role and select import.



Figure 6 - VMware 12 Workstation Opening a VM

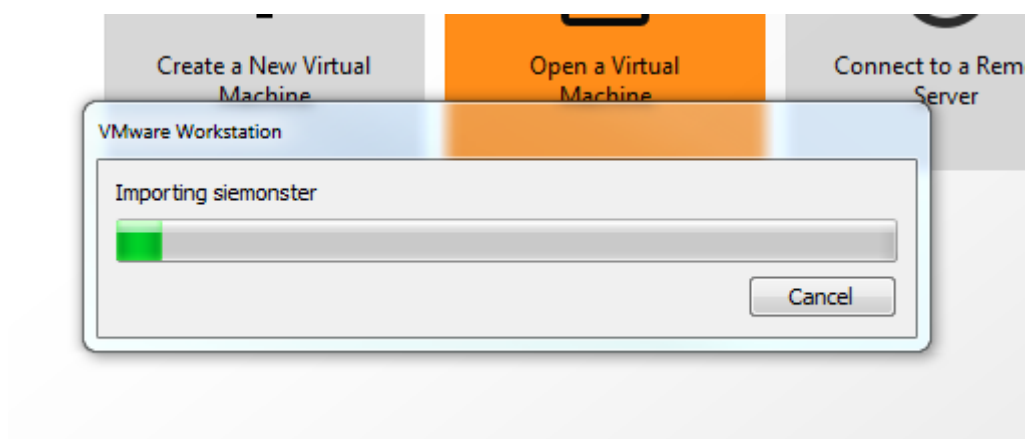


Figure 7 - VMware import

Once the image has been imported the appliance can be powered on. Adjustments to system memory are required, see the FAQ. Minimum requirement is 4GB per appliance.

Crucial: If the system memory is changed, then please edit the following file: `/etc/default/elasticsearch`
Find the line: `ES_HEAP_SIZE=2g` This applies to all images except Capricorn.

Change 2g to half of the system memory, e.g. if system memory is 8GB set to `ES_HEAP_SIZE=4g`

3 CONFIGURATION

3.1 DEFAULT PASSWORDS

Host	User	Password	Access
192.168.0.101-104	siemonster	siemonster	SSH/Local Access
Proteus Kibana	siemonster	siemonster	http:// 192.168.0.103:80
Proteus Incident Response	dev	dev	http://192.168.0.103:8443/admin
Proteus Incident Response Admin	admin	admin	http://192.168.0.103:8443
Proteus→Capricorn Graylog	admin	admin	http://192.168.0.103:8080
192.168.0.101-104 MySQL root	siemonster	siemonster	Local Access

Figure 9 – Default Passwords to be changes after build

3.2 STATIC IP AND HOSTS FILE SETUP

In order that each server can be resolved by name, a suitable hosts file must be configured and a static IP address set. Examples have been provided in the templates/network folder on the image

Plan out the IP range that will be used and adjust the values accordingly. The template files presume an IP range from 192.168.0.101 to 192.168.0.104 and can be adjusted to suit your environment.

Server Name	IP Address	Role
Kraken	192.168.0.101	Database Cluster Node 1
Tiamat	192.168.0.102	Database Cluster Node 2
Proteus	192.168.0.103	Front End (Database in 2 node instance)
Capricorn	192.168.0.104	Front End

Figure 10 - Default Server IP allocation

3.2.1 Host File Setup

Hosts file examples are located on the image in the following location templates/network/

kraken.hosts example

```
127.0.0.1    localhost
127.0.1.1    kraken
192.168.0.102 tiamat
192.168.0.103 proteus
192.168.0.104 capricorn
```

Static IP Example templates/network/kraken.interfaces:

```
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.0.101
gateway 192.168.0.1
netmask 255.255.255.0
dns-nameservers 192.168.0.1
```

If these examples are suitable they can be simply copied over, replace Kraken with the host name your building, ie Kraken, Proteus, Capricorn or Tiamat

If you need to modify the templates use any Linux editing tool, for beginners use pico

- pico templates/network/kraken.hosts
- pico templates/network/kraken.interfaces

Use the arrow keys to move around and change the IP subnets, ip or gateways of all entries. Control O to write out and press enter to save for both files.

Replace Kraken with the host name your building, ie Kraken, Proteus, Capricorn or Tiamt and copy your modified files across into real configs.

- sudo cp templates/network/kraken.hosts /etc/hosts
- sudo cp templates/network/kraken.interfaces /etc/network/interfaces

Verify you copied over the files over before rebooting by showing that all your IP entries are set using cat

- cat /etc/hosts
- cat /etc/network/interfaces

If everything is set perfectly reboot

- sudo reboot



3.3 ROLES EXPLAINED

2 Server Instance: Proteus Single Node & Capricorn

4 Server Instance: Proteus Multi Node, Capricorn, Kraken and Tiamat.

Proteus Single Node: The core is run as a single Elasticsearch node without redundancy or scalability perfect for small companies or a trial of **SIEMonster**. By running this node and Capricorn you are running the baby **SIEMonster**. Proteus will house the Elasticsearch database. Data is forked to Capricorn for instant data stream analysis, virus outbreaks, admin password attempts etc.

Proteus Multi Node: The core appliance controls the cluster (Kraken/Tiamat) acting in a non-data eligible master node, Proteus can be scaled out to affect load balancing of requests as required. Data is forked to Kraken/Tiamat for long term analysis and Capricorn for instant data stream analysis, virus outbreaks, admin password attempts etc.

Capricorn: The role of Capricorn is to affect a volatile indexing system for Streams and Alerting. The indices are limited by size or count and are subsequently overwritten. Should additional visualization or storage of streaming/alert data be a requirement then outputs can be easily configured to afford this.

Kraken & Tiamat: The Elasticsearch data nodes Kraken & Tiamat are assigned as the cluster workhorses. These nodes can be scaled out to increase performance as and when required. These nodes provide the 6-12month+ index storage required for forensic analysis of security events.

3.4 SCRIPTING ROLE ASSIGNMENT

The following scripts have been provided in the siemonster home directory:

```
kraken-installer.sh
tiamat-installer.sh
proteus-multinode-installer.sh
proteus-singlenode-installer.sh
capricorn-installer.sh
```

The order for installation is covered in the following section.

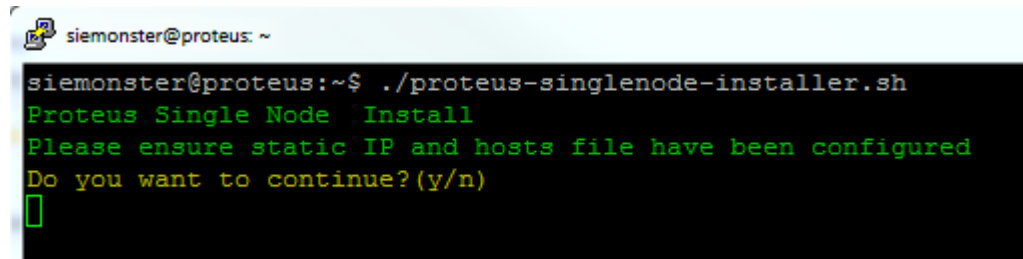
4 DEPLOYMENT

4.1 MINIMUM 2 NODE INSTALLATION INSTRUCTIONS

Proteus Installation: Import the siemonster.ova file and assign the name proteus-single-node. Ensure that the static IP and hosts files have been set as per section 3.2 and don't forget to sudo reboot.

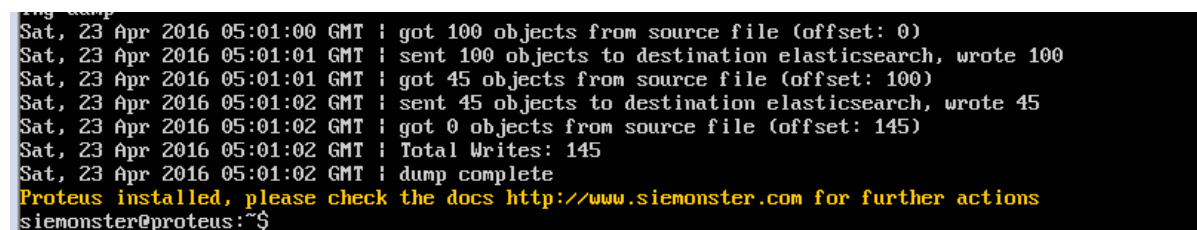
- Run the following script to initiate role assignment, enter 'y' to continue:

./proteus-singlenode-installer.sh



```
siemonster@proteus: ~$ ./proteus-singlenode-installer.sh
Proteus Single Node Install
Please ensure static IP and hosts file have been configured
Do you want to continue?(y/n)
█
```

Figure 11 - Script install Proteus Single node (Database)



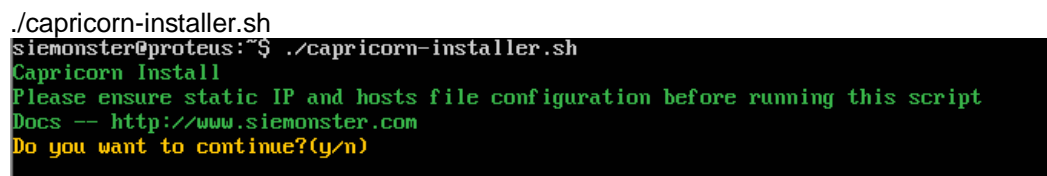
```
Sat, 23 Apr 2016 05:01:00 GMT : got 100 objects from source file (offset: 0)
Sat, 23 Apr 2016 05:01:01 GMT : sent 100 objects to destination elasticsearch, wrote 100
Sat, 23 Apr 2016 05:01:01 GMT : got 45 objects from source file (offset: 100)
Sat, 23 Apr 2016 05:01:02 GMT : sent 45 objects to destination elasticsearch, wrote 45
Sat, 23 Apr 2016 05:01:02 GMT : got 0 objects from source file (offset: 145)
Sat, 23 Apr 2016 05:01:02 GMT : Total Writes: 145
Sat, 23 Apr 2016 05:01:02 GMT : dump complete
Proteus installed, please check the docs http://www.siemonster.com for further actions
siemonster@proteus:~$
```

Figure 12 - Proteus scripting complete - Reboot required

- The script may take a minute or two to run. Reboot the appliance on completion. Don't be alarmed the prompt still says siemonster@proteus it changes after script is run and reboot

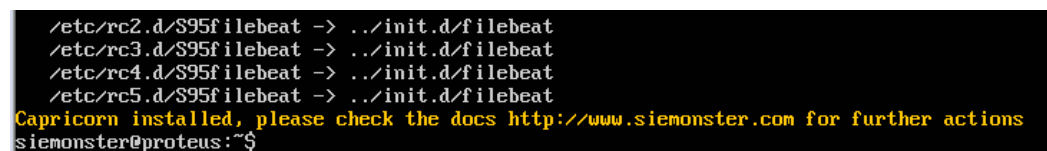
Capricorn Installation: Import the siemonster.ova file once more and assign the name capricorn. Ensure that the static IP and hosts files have been set as per section 3.2 and don't forget to sudo reboot.

- Run the following script to initiate role assignment, enter 'y' to continue:



```
siemonster@proteus:~$ ./capricorn-installer.sh
Capricorn Install
Please ensure static IP and hosts file configuration before running this script
Docs -- http://www.siemonster.com
Do you want to continue?(y/n)
```

Figure 13 - Capricorn image role script



```
/etc/rc2.d/S95filebeat -> ../init.d/filebeat
/etc/rc3.d/S95filebeat -> ../init.d/filebeat
/etc/rc4.d/S95filebeat -> ../init.d/filebeat
/etc/rc5.d/S95filebeat -> ../init.d/filebeat
Capricorn installed, please check the docs http://www.siemonster.com for further actions
siemonster@proteus:~$
```

Figure 14 - Capricorn role complete, reboot for name change to take effect

- Please allow up to 3 minutes for install, reboot on completion.

4.2 CORPORATE 4 NODE INSTALLATION INSTRUCTIONS

The following instructions must be done in this order. These tasks normally are quite complex for a running cluster and health monitor. These have been scripted to reduce installation headaches. A beginner user can have it up in less than 20 minutes. This can normally take a day without scripts.

Note: Ensure that the static IP and hosts files have been set as per section 3.2

Order of install per import:

1. kraken-installer.sh
2. tiamat-installer.sh
3. proteus-multinode-installer.sh
4. capricorn-installer.sh

Kraken Installation: Import the siemonster.ova file and assign the name 'kraken'. Ensure that the static IP and hosts files have been set as per section 3.2 and don't forget to sudo reboot.

- Run the following script to initiate role assignment, enter 'y' to continue

sudo ./kraken-installer.sh

```
siemonster@proteus:~$ ./kraken-installer.sh
Kraken Install
Please ensure static IP and hosts file configuration before running this script
Docs -- http://www.siemonster.com
Do you want to continue?(y/n)
```

Figure 15 - Kraken Installation First of the 4 nodes

- Reboot on completion.

Tiamat Installation: Import the siemonster.ova file and assign the name 'tiamat'. Ensure that the static IP and hosts files have been set as per section 3.2 and don't forget to sudo reboot.

- Run the following script to initiate role assignment, enter 'y' to continue:

sudo ./tiamat-installer.sh

```
siemonster@proteus:~$ ./tiamat-installer.sh
Tiamat Install
Please ensure static IP and hosts file configuration before running this script
Docs -- http://www.siemonster.com
Do you want to continue?(y/n)
```

Figure 16 - Tiamat Installation Database Node 2

- Reboot on completion

Proteus Installation: Import the siemonster.ova file and assign the name 'proteus'. Ensure that the static IP and hosts files have been set as per section 3.2 and don't forget to reboot.

- Run the following script to initiate role assignment, enter 'y' to continue:

```
sudo ./proteus-multinode-installer.sh
```

```
siemonster@proteus:~$ ./proteus-multinode-installer.sh
Proteus Multi Node Install
Please ensure static IP and hosts file configured
Data nodes Kraken & Tiamat must live before proceeding
Do you want to continue?(y/n)
```

Figure 17 - Proteus multi node installer

- Reboot on completion

Capricorn Installation: Import the siemonster.ova file and assign the name 'capricorn'. Ensure that the static IP and hosts files have been set as per section 3.2 and don't forget to reboot.

- Run the following script to initiate role assignment, enter 'y' to continue:

```
sudo ./capricorn-installer
```

```
siemonster@proteus:~$ ./capricorn-installer.sh
Capricorn Install
Please ensure static IP and hosts file configuration before running this script
Docs -- http://www.siemonster.com
Do you want to continue?(y/n)
```

Figure 18 - Capricorn Installer Script

```
/etc/rc2.d/S95filebeat -> ../init.d/filebeat
/etc/rc3.d/S95filebeat -> ../init.d/filebeat
/etc/rc4.d/S95filebeat -> ../init.d/filebeat
/etc/rc5.d/S95filebeat -> ../init.d/filebeat
Capricorn installed, please check the docs http://www.siemonster.com for further actions
siemonster@proteus:~$
```

Figure 19 - Capricorn installed reboot for name change

- Reboot on completion

5 ACCESSING WEB INTERFACES AND DASHBOARDS

5.1 KIBANA

The Kibana web interface may be accessed at the IP address assigned to the Proteus appliance. In the example shown above the URL would be:

<http://192.168.0.103>

The username and password is 'siemonster'.

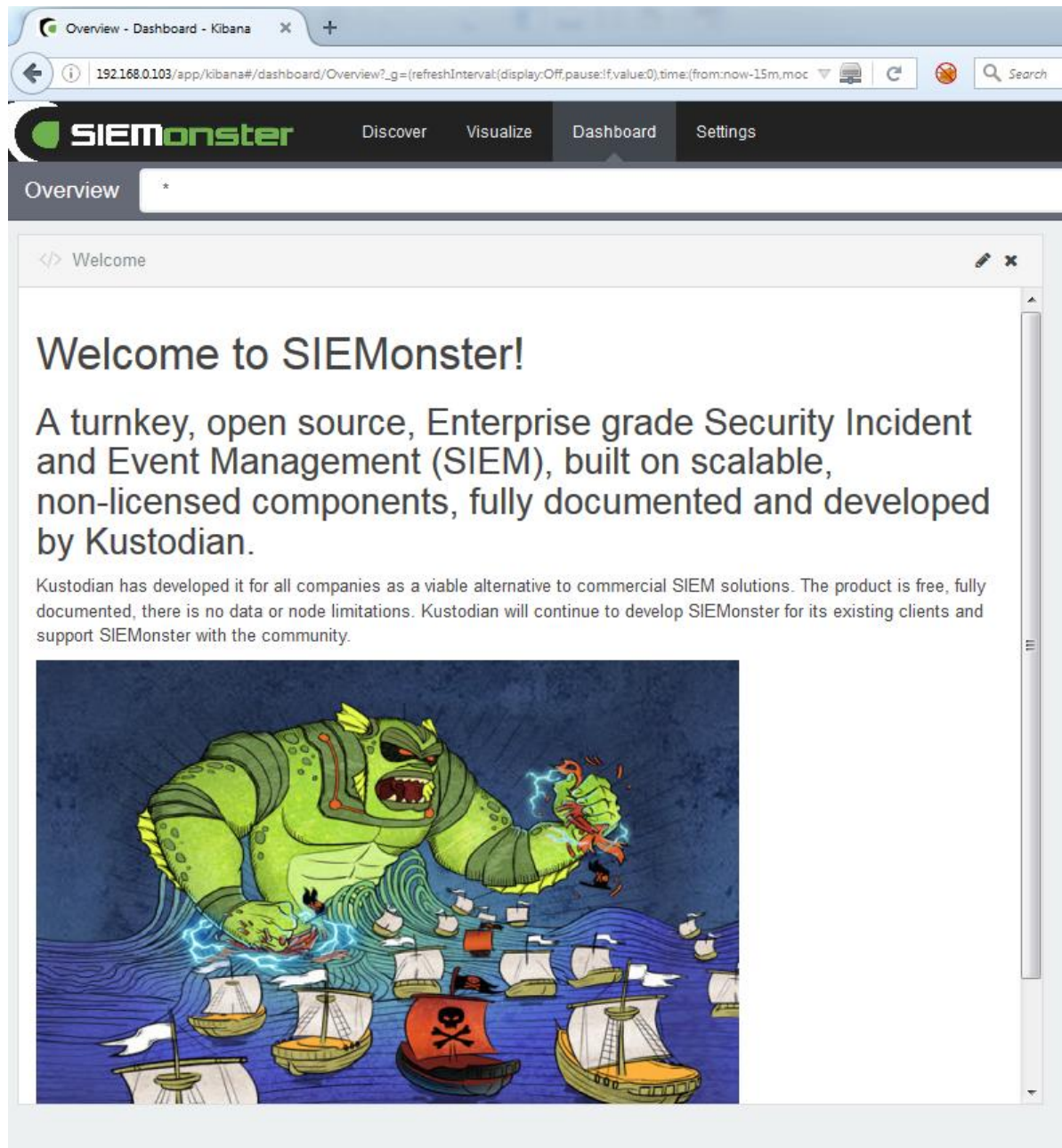


Figure 20 - SIEMonster Default page in Kibana

5.2 INCIDENT RESPONSE

The Incident Response web interface may be accessed at the IP assigned to the Proteus appliance on port 8443. In the example shown above the URL would be:

<http://192.168.0.103:8443>

The username and password is 'dev'.



Figure 21 - Incident Response normal user start-up screen

The URL for administration is: <http://192.168.0.103:8443/admin> - username and password 'admin'.

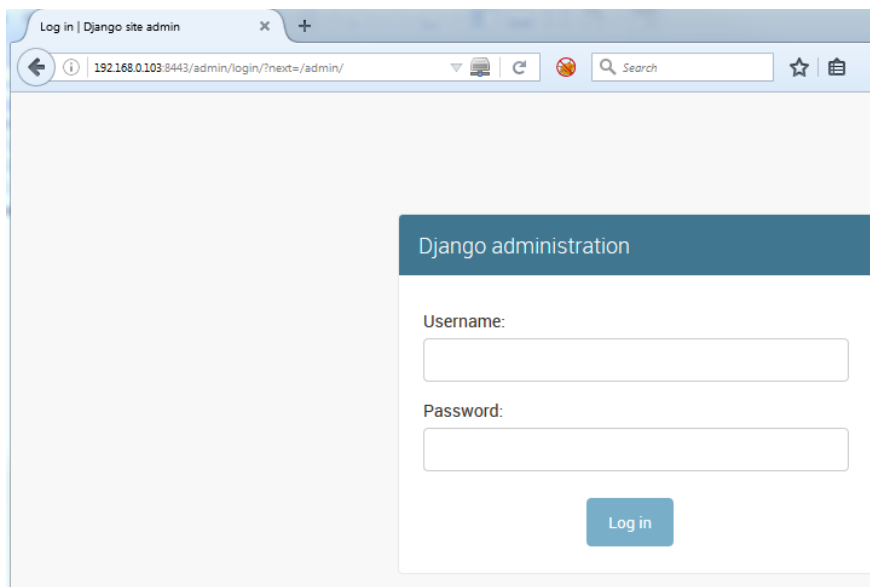


Figure 22 - Admin Access Incident Response

5.3 GRAYLOG STREAMS & ALERTS

The Graylog web interface may be accessed at the IP address assigned to the Proteus appliance. In the example shown above the URL would be:

<http://192.168.0.103:8080>

The username and password is 'admin'.

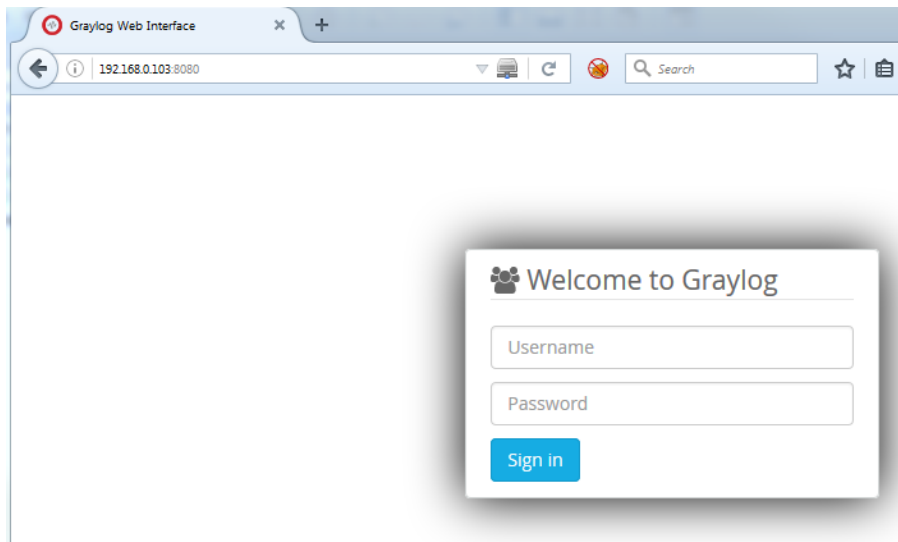


Figure 23 - Capricorn Graylog logon verification

5.4 GRAYLOG TIME ZONE

To get Graylog (Capricorn) to use your current timezone go to the Capricorn shell prompt and type

- `sudo graylog-ctl set-timezone newtimezone`

example: `sudo graylog-ctl set-timezone Australia/Melbourne` TimeZone formats can be found here <http://joda-time.sourceforge.net/timezones.html>

- `sudo graylog-ctl reconfigure`

In Graylog you can check your changes under System/Overview there is a section called Time Configuration where you can check if you did it correctly.

5.5 FUNCTIONALITY CHECK

To check functionality of the system the following steps can be taken:

- Login to the Graylog web interface and click on the Search menu item. Change the criteria to Search in the last 30 minutes:

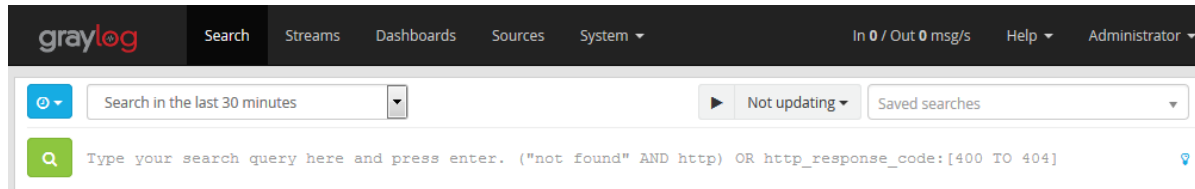


Figure 24 - Graylog search criteria verification

- Check that Histogram data is displayed:

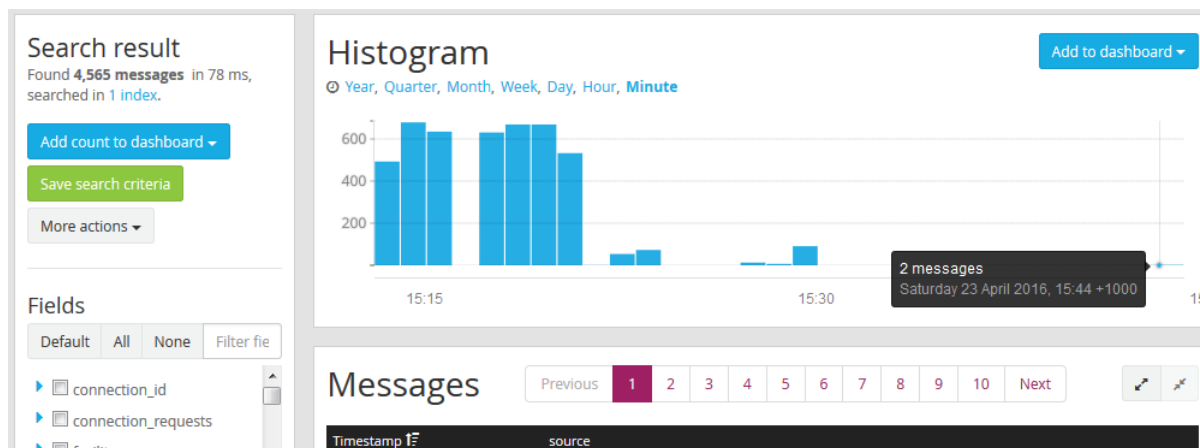


Figure 25 - Example Histogram to ensure Graylog functioning

- Using an SSH client, login to the Capricorn appliance (in this example 192.168.0.104) Login using a non-existent user, e.g. 'baduser' with any password
- Open the Kibana web interface (in this example <http://192.168.0.103>)
- Open the Discover menu item
- Click on the clock top right, set Auto-refresh for 5 seconds. Check for activity as shown below:

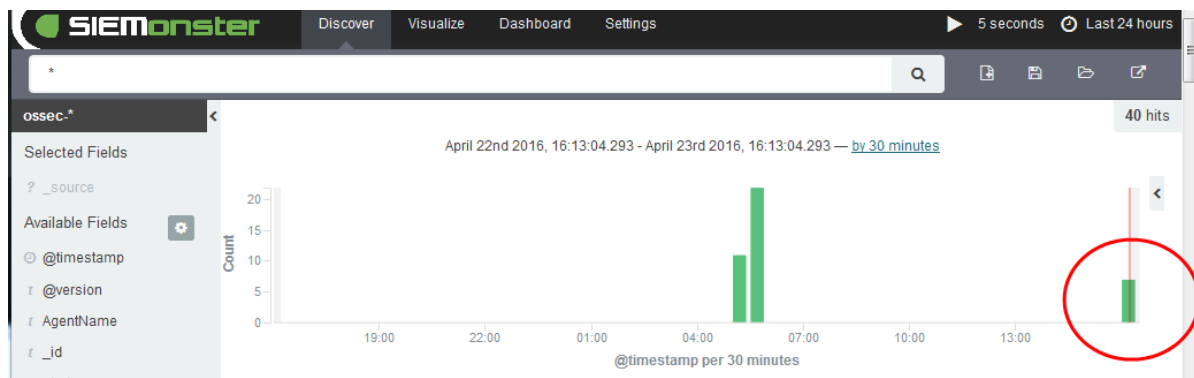


Figure 26 - Siemonster Kibana verification

- Open the top event and verify failed login:

t rule.description	Attempt to login using a non-existent user
# rule.firedtimes	3
t rule.groups	syslog, sshd, invalid_login, authentication_failed
# rule.sidid	5,710
t source	/var/ossec/logs/alerts/alerts.json
t srcip	192.168.0.16
t srcuser	baduser
t tags	beats_input_codec_json_applied
t type	log

Figure 27 - Verification Failed logon appears

- Click on the Dashboard menu item at the top of the Page next to Discover, Visualize
- Open the OSSEC Alerts Dashboard and check for data Little Folder on top right

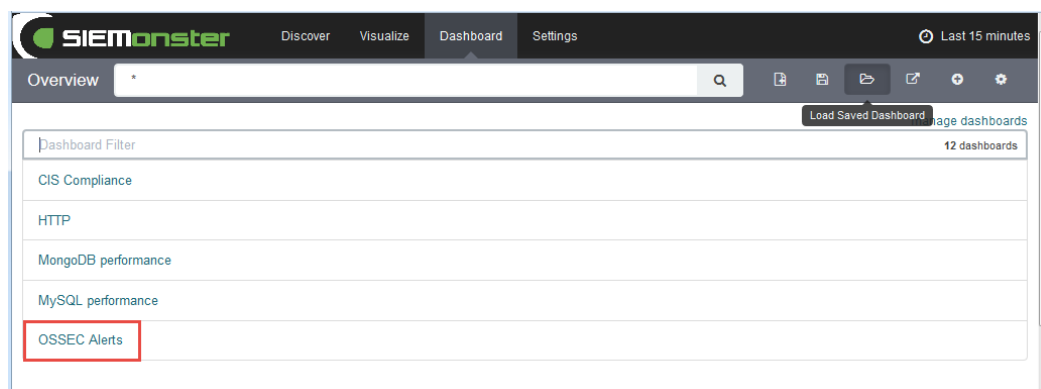


Figure 28 - Siemonster OSSEC Alerts verification

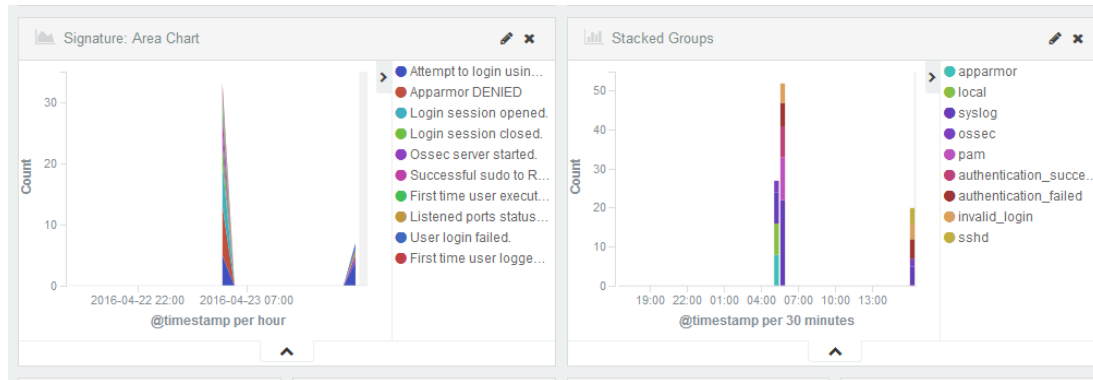


Figure 29 - OSSEC Signature Alerts

Last alerts						
Time	AgentName	AgentIP	rule.sidid	rule.AlertLevel	rule.description	full_log
▶ April 23rd 2016, 16:12:40.000	capricorn	-	5,710	5	Attempt to login using a non-existent user	Apr 23 16:12:38 capricorn sshd[1970]: Failed password for invalid user baduser from 192.168.0.1 port 64192 ssh2
▶ April 23rd 2016, 16:12:40.000	capricorn	-	5,710	5	Attempt to login using a non-existent user	Apr 23 16:12:39 capricorn sshd[1970]: Failed password for invalid user baduser from 192.168.0.1 port 64192 ssh2
▶ April 23rd 2016, 16:12:38.000	capricorn	-	5,710	5	Attempt to login using a non-existent user	Apr 23 16:12:37 capricorn sshd[1970]: Failed password for invalid user baduser from 192.168.0.1 port 64192 ssh2
▶ April 23rd 2016, 16:12:36.000	capricorn	-	5,503	5	User login failed.	Apr 23 16:12:34 capricorn sshd[1970]:

Figure 30 - Raw alerts view



5.6 SSL CONFIGURATION

To setup SSL access to Kibana to ensure encryption between the browser and the SIEM, and FIR, the following steps can be taken. This was not done as part of the SIEM build to ensure IP configurations, server names or your own certificates could be used.

With regard to Graylog SSL access, this has been found to be unstable and displays different results in different browsers. Also there are issues with self-signed certificates. We will wait on the upcoming Graylog 2.0 release test, and roll out an update when it is stable.

Self-signed certificates were created during the install and are located at `/etc/nginx/ssl/nginx.crt` of your **SIEMonster** instances. (ie Proteus). Note: If you have your own certificates then copy them to the same location and modify as appropriate. You can edit `/etc/ssl/openssl.cnf` to personalize it.

Scripts have been created in `scripts/ssl` to simplify the process.

To enable SSL access to Kibana & FIR run the command on the Proteus appliance:

- `sudo ./scripts/ssl/ssl_web_enable.sh`

Check access to the web interfaces, use the standard links as you will automatically be redirected to the SSL encrypted session. If there are any issues with connection, then run the following script to return to non SSL access and troubleshoot:

- `sudo ./scripts/ssl/ssl_web_disable.sh`

5.7 CLUSTER HEALTH AND MONITORING

To monitor your cluster and stack health and detailed statistics we have included a health monitor. These can be accessed on your Proteus URL. Change the IP to suit your Proteus IP

On the Proteus appliance the URLs are as follows:

http://192.168.0.103:9200/_plugin/kopf

http://192.168.0.103:9200/_plugin/hq

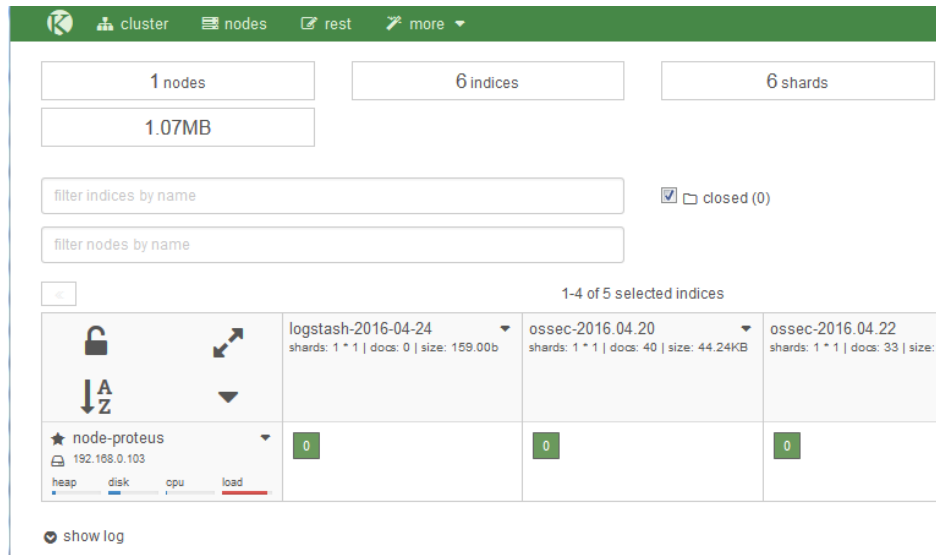


Figure 31 - Health Check view

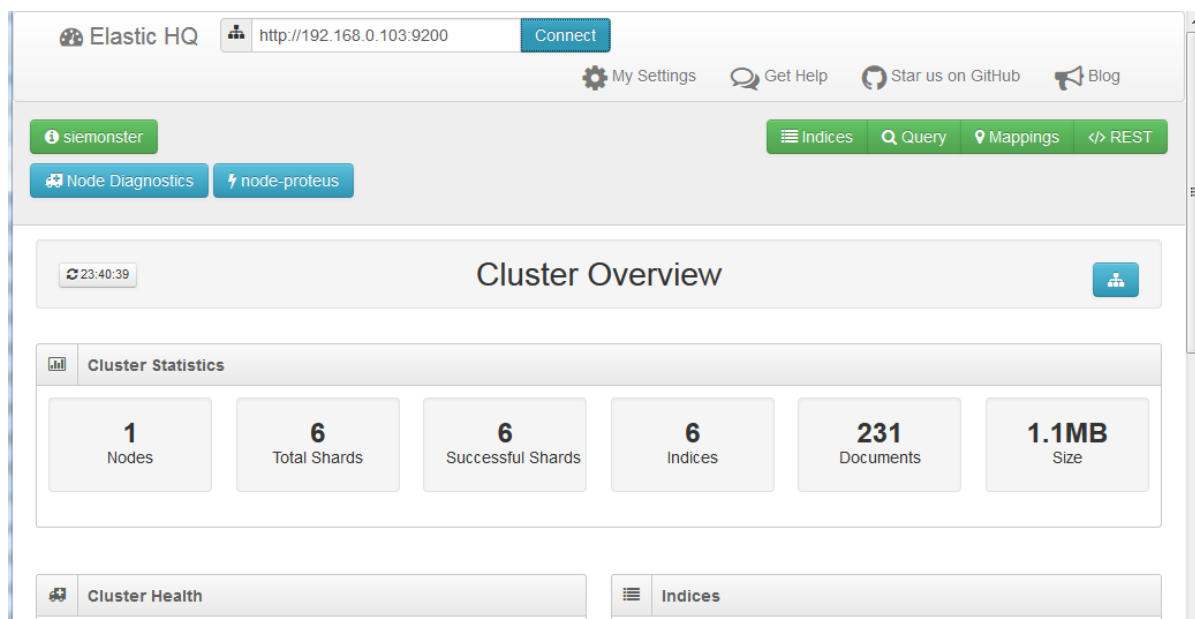


Figure 32 - Cluster Health checking

On the command line at the Proteus appliance cluster health may be found using the command:

- `curl http://localhost:9200/_cluster/health?pretty`

```
siemonster@proteus:~$ curl http://localhost:9200/_cluster/health?pretty
{
  "cluster_name" : "siemonster",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 1,
  "number_of_data_nodes" : 1,
  "active_primary_shards" : 6,
  "active_shards" : 6,
  "relocating_shards" : 0,
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
}
```

Figure 33 - Curl output

6 LOGSTASH TEMPLATES

In order to better analyse and visualise incoming logs into Logstash from Windows, Linux, Syslogs and alike, we have developed a better architecture that will allow for quicker device parsers to be written. The premise being, to re-organising the way Logstash processes the config files for better flow and understanding and fast new device take-up. For example, a new Cisco ASA firewall hits the market, currently no configuration file has a parser for it. A listening catch-all will ensure it is picked up in a generic syslog listener. Once identified the fields will then be moved across into the ASA config file.

The layout of the files in `/etc/logstash/conf.d` will now be

- 00-inputs - Receives the logs and tags them
- 99-outputs - Send the logs to Elasticsearch based on their tags.

The filter files per device now fall in between 00-Input and 99-output, between process the files based on their tags, i.e. 10-Windows, 30-Apache. So, rather than have config files with inputs, filters and outputs we have separation of function. To do this we will delete all the files in the `conf.d` directory, write a new template and copy some new files down ready to run on the new template

On Proteus, grab the SIEMonster syslog template to your home directory:

- `wget https://raw.githubusercontent.com/siemonster/elasticsearch/master/syslog.template`

Apply the template:

- `curl -XPUT 'http://localhost:9200/_template/syslog' -d@syslog.template`

Remove existing config files

- `sudo rm ~/templates/logstash/*.conf`
- `sudo rm /etc/logstash/conf.d/*.conf`

Grab the new party pack:

- `wget -O ~/templates/logstash/confpack.tar.gz https://github.com/siemonster/logstash/raw/master/confpack.tar.gz`
 - `tar xvzf ~/templates/logstash/confpack.tar.gz -C ~/templates/logstash/`
 - `rm ~/templates/logstash/confpack.tar.gz`
 - `sudo cp ~/templates/logstash/*.conf /etc/logstash/conf.d/`

To ensure syslog's times are accurate you must edit the

/etc/logstash/conf.d/03-multisyslog-filter.conf file and replace the time zone fields with your own time zone. <http://joda-time.sourceforge.net/timezones.html>

Otherwise events may appear in the future and will not be visible in Kibana. The default value is "Australia/Melbourne"

e.g. sudo nano /etc/logstash/conf.d/03-multisyslog-filter.conf

```
syslog_pri { }
date {
  match => [ "syslog_timestamp", "MMM d HH:mm:ss",
  timezone => "Australia/Melbourne"
}
```

Figure 34 - Time zone mods

Restart logstash:

- sudo service logstash restart

WARNING: All the conf files in our repository have been tested and verified. There are multiple versions of Logstash out with varying syntax. V1.5 and V2.2 each have different syntax for logstash conf files. If you download from other repositories Logstash service may not start due to custom input/output filter tagging. Logstash is version specific ensure ssl_values match your version number ie Logstash 2.2 uses ssl_certificate where logstash 1.5 uses ssl_cert. These simple syntax changes will stop logstash from working. If in doubt just email us at info@siemonster.com and we will validate it.

7 INSTALLING AGENTS

Now that the **SIEMonster** is up and running. It's time to install some agents to get some data into the SIEM. You will need to install an agent on the boxes that support agents like Windows and Linux. For boxes that don't support agents you will need to forward syslogs to the SIEM. The basic premise for endpoint installation is the following.

- Install the agent on the endpoint
- Copy the configuration file (contains IP/Port details) to the endpoint to point to connect back to **SIEMonster**.
- Copy the certificate to the endpoint to encrypt the traffic back to the SIEM
- Copy the correct template on the SIEM to catch the endpoint events.

7.1 NXLOG SIEM AGENTS FOR MICROSOFT HOSTS

Endpoint Agent Software & Configuration: NXLOG is a universal log collector and forwarder. The software NXLOG can be found directly from the vendor or on agents.siemonster.com website.

Certificate for endpoint: SSL certificates for transport were created during the installation process to ensure encryption between the client and the SIEM to protect your data travelling to the SIEM. The certificate needed for NXlog can be found on your Proteus server at

`/etc/pki/tls/certs/logstash-forwarder.crt`

Note: If you have your own certificates then the private key is located at `/etc/pki/tls/private/logstash-forwarder.key`

Configuration file for SIEM: Within the `templates/nxlog` folder on your Proteus server there are example configuration files for Windows Domain, Exchange 2007, Exchange 2010-2013 and File Servers. These are required to catch your endpoint traffic. Eg `nxlog.conf.domain.controller`

7.1.1 Instructions for agent installation

To install the agent on the endpoint such as a Windows Domain Controller, you will need 3 files. Once you have these files it is suggested you put them on a file share for quick agent installation for multiple hosts or copy them onto a USB stick for small environments or labs.

The NXLOG.msi file. This is the installer onto your endpoint server

The certificate on your Proteus Appliance to encrypt your traffic to the SIEM

The `nxlog.conf` file also on your Proteus appliance to send the traffic to the SIEM on the right port.

NXLOG Download

The latest NXLOG agent can be downloaded from:

<http://agents.siemonster.com/nxlog-ce-2.9.1504.msi>

SHA256 Checksum:

D49B7BF1A631361DC2B701848BD370668BB08D11BD869C2C48C7A31E21B3C154

Certificate Retrieval

- Access the Proteus appliance using SCP or mounted USB to retrieve the SSL certificate from /etc/pki/tls/certs/logstash-forwarder.crt & the required nxlog.conf file from templates/nxlog

Configuration Retrieval

- You will need to modify this nxlog.conf file to connect back to **SIEMonster**. Within the output section of the file, update the IP address/port to match the Proteus IP address as follows:

<Output out>

```
Module    om_tcp
# Module  om_ssl
Host      192.168.0.103
Port      3522
# CAFile  %CERTDIR%\logstash-forwarder.crt
```

</Output>

- For initial deployment it is recommended to comment out the line beginning 'CAFile' as above and run without SSL to first check for a successful connection.
- Also comment out the line 'Module om_ssl' and add the line 'Module om_tcp' as above.

Installation on Endpoint with the 3 components

- Install the NXLOG msi on your Microsoft endpoint with Admin credentials.
- Copy over the modified nxlog.conf file to "C:\Program Files (x86)\nxlog\conf" on your endpoint. Ensure the name of the copied file is nxlog.conf not nxlog.conf.domain.controller. May require view file extensions in Windows.
- Copy the certificate to "C:\Program Files (x86)\nxlog\cert" on your endpoint.
- Start the NXlog service

Proteus SIEM steps

On Proteus, if SSL transport for logs is required, edit the `/etc/logstash/conf.d/00inputs.conf` file.

Uncomment the lines containing SSL configuration:

```
GNU nano 2.2.6      File: 00-inputs.conf
#   codec => json
#   }
tcp {
  type => "apache"
  port => 3521
#   ssl_enable => true
#   ssl_cert => "/etc/pki/tls/certs/logstash-forwarder.crt"
#   ssl_key => "/etc/pki/tls/private/logstash-forwarder.key"
#   ssl_verify => false
}
tcp {
  type => "events"
  port => 3522
#   ssl_enable => true
#   ssl_cert => "/etc/pki/tls/certs/logstash-forwarder.crt"
#   ssl_key => "/etc/pki/tls/private/logstash-forwarder.key"
#   ssl_verify => false
#   codec => json_lines
  tags => ["windows", "eventlog"]
}
tcp {
  type => syslog
  port => 3523
  tags => ["main_syslog"]
}
}
```

Figure 34 – SSL Configuration

Restart Logstash on the Proteus appliance if any changes are made:

- `sudo service logstash restart`

Check that the NXLOG agent has connected:

- `netstat -ant | grep 3522` (or other configured port)

The connection may take up to 2-3 minutes to be established. See the troubleshooting section should no connection be established. Once established, repeat the installation on other hosts.

7.2 FILEBEAT SIEM AGENTS FOR LINUX OR APACHE

Filebeat is a lightweight, open source shipper for log file data. As the next-generation Logstash Forwarder, Filebeat tails logs and quickly sends this information to Logstash for further parsing or to Elasticsearch for centralized storage and analysis. **SIEMonster** uses this agent for collecting logs from Unix hosts, typically Apache web logs. The latest Filebeat agent Debian 64 package may be downloaded from:

http://agents.siemonster.com/filebeat_1.2.1_amd64.deb

SHA256 Checksum:

015336320A85D2DE4F3FF2A657861BBD840381EAC623B103D808436FAE84559F

Transfer this file via SCP to the target server and install using the following command:

- `sudo dpkg -i filebeat_1.2.1_amd64.deb`

Once installed the filebeat service will be inactive and the configuration file can be found at `/etc/filebeat/filebeat.yml`. This configuration file must be modified to suit the logs being monitored and the IP address of the Proteus server.

Retrieve the SSL certificate from the Proteus appliance located at `/etc/pki/tls/certs/logstash-forwarder.crt` via SCP and transfer to the target server. On the target server, copy the certificate to `/etc/pki/tls/certs/logstash-forwarder.crt`. If this location does not exist, create using the command:

- `sudo mkdir -p /etc/pki/tls/certs`

Secure the certificate as follows:

- `sudo chown root:root /etc/pki/tls/certs/logstash-forwarder.crt`
- `sudo chmod 644 /etc/pki/tls/certs/logstash-forwarder.crt`

Edit the Filebeat configuration file `/etc/filebeat/filebeat.yml` as follows: The first element to change will be the 'paths' directive in the prospectors section which is on Capricorn OSSEC path in Figure Below.

For example, to modify this for Apache logs this path may be altered to:

`/var/log/apache2/access.log`

```
##### Filebeat Configuration Example #####

##### Filebeat #####
filebeat:
  # List of prospectors to fetch data.
  prospectors:
    # Each - is a prospector. Below are the prospector specific configurations
    -
      # Paths that should be crawled and fetched. Glob based paths.
      # To fetch all ".log" files from a specific level of subdirectories
      # /var/log/*/*.log can be used.
      # For each file found under this path, a harvester is started.
      # Make sure not file is defined twice as this can lead to unexpected behaviour.
      paths:
        - /var/ossec/logs/alerts/alerts.json
        # - C:\programdata\elasticsearch\logs\
```

Figure 35 - Filebeat path modification which is the path settings for Capricorn where OSSEC is installed.

- Next ensure the 'elasticsearch' section is commented out:

```
### Elasticsearch as output
#elasticsearch:
# Array of hosts to connect to.
# Scheme and port can be left out and will be set to the default (http and 9200)
# In case you specify an additional path, the scheme is required: http://localhost:9200/path
# IPv6 addresses should always be defined as: https://[2001:db8::1]:9200
# hosts: ["localhost:9200"]
```

Figure 36 - Commented out Elasticsearch

- Next locate the Logstash output section and enter the IP address of the Proteus server.

```
### Logstash as output
logstash:
# The Logstash hosts
hosts: ["192.168.0.103:3520"]

# Number of workers per Logstash host.
#worker: 1
```

Figure 37 - Proteus IP inserted

- For SSL transport locate the TLS section and make the following changes:

```
# Optional TLS. By default is off.
tls:
# List of root certificates for HTTPS server verifications
certificate_authorities: ["/etc/pki/tls/certs/logstash-forwarder.crt"]
```

Figure 38 - SSL Transport follow picture for correct path

For SSL transport ensure the SSL section is uncommented:

```
input {
  beats {
    port => 3520
    type => "apache"
    ssl => true
    ssl_certificate => "/etc/pki/tls/certs/logstash-forwarder.crt"
    ssl_key => "/etc/pki/tls/private/logstash-forwarder.key"
    codec => json
  }
}
```

Figure 39 - Uncommented SSL ad path

Restart Logstash using the command:

- `sudo service logstash restart`

From the Capricorn appliance test the SSL connection as follows:

- `sudo service filebeat stop`
- `curl -v --cacert /etc/pki/tls/certs/logstash-forwarder.crt https://192.168.0.103:3520` (Replace IP with yours)

A successful response should contain 'curl: (52) Empty reply from server'

An unsuccessful response will contain 'curl: (51) SSL: certificate verification failed (result: 5)'

See the troubleshooting section for diagnosing SSL problems.

OSSEC agents for Linux servers are installed via the OSSEC binary:

<http://ossec.github.io/downloads.html> (Server/Agent Unix)

Documentation for agent install on Unix: [http://ossec-](http://ossec-docs.readthedocs.io/en/latest/manual/installation/install-source.html)

[docs.readthedocs.io/en/latest/manual/installation/install-source.html](http://ossec-docs.readthedocs.io/en/latest/manual/installation/install-source.html)

Restart Filebeat with the command:

- `sudo service filebeat restart`

Test for connection status on the Proteus appliance:

- `sudo netstat -ant |grep 3520` (or other configured port)

```
root@proteus:~/scripts# netstat -ant |grep 3520
tcp6      0      0 :::3520          :::*              LISTEN
tcp6      0      0 192.168.0.103:3520 192.168.0.104:54708 ESTABLISHED
```

Figure 40 - Working connection

7.3 OSSEC HIDS AGENTS FOR WINDOWS HOSTS

OSSEC HIDS agents may be installed as follows to report to the OSSEC/Wazuh manager on the Capricorn appliance. This is great edition to the SIEM. For detailed information on OSSEC have a look at the OSSEC reference manual <http://ossec.github.io/docs/manual/index.html>

On the Capricorn appliance, run the command:

- `sudo /var/ossec/bin/manage_agents`

Note: Using a puTTY session from Windows to Capricorn will allow easy copy and paste for generated keys than mucking around with vmware tools and copy/pasting.

The following options will be available:

```
*****
* OSSEC HIDS v2.8 Agent manager.      *
* The following options are available: *
*****
(A)dd an agent (A) .
(E)xtract key for an agent (E) .
(L)ist already added agents (L) .
(R)emove an agent (R) .
(Q)uit.
Choose your action: A,E,L,R or Q: q
```

Figure 41 - OSSEC HIDS Menu

- Choose 'A'

Add a name for the agent and an IP address that should match the one the agent will be connecting from i.e CALFORNIADC01 192.168.0.100

- Press 'Y'

```
- Adding a new agent (use '\q' to return to the main menu) .
Please provide the following:
* A name for the new agent: myagent
* The IP Address of the new agent: 192.168.0.100
* An ID for the new agent[001]:
Agent information:
ID:001
Name:myagent
IP Address:192.168.0.100
Confirm adding it?(y/n): [Y]
```

Figure 42 - Setting up the OSSEC agent IP and name

Retrieve the agent key information by entering 'E' for extract and the ID for the agent. Copy this key as it will be required for the remote agent install.

Example:

Agent key information for '001' is:

MDAxIFRlc3RBZ2V0biAxMTEuMTEuLjExMS4xMTEgY2MxZjA1Y2UxNWQyNzEyNjdIMmE3MTRIO
DIOMTA1YTgxNTM5ZDIiN2U2ZDQ5MWYxYzBkOTU4MjRmNjU3ZmI2Zg==

Restart the OSSEC/Wazuh manager:

- `sudo /var/ossec/bin/ossec-control restart`

To install the remote agent on a windows machine, first download the agent install file
<http://agents.siemonster.com/ossec-agent-win32-2.8.3.exe>

SHA256Checksum:

FEB135286ED19382CC479B7F035BE5296360291900FAF01338ACCAD59F910E4A

Note. The agent must be installed with administrator privileges.

Launch the agent and enter the IP address of the Capricorn appliance along with the key previously presented.

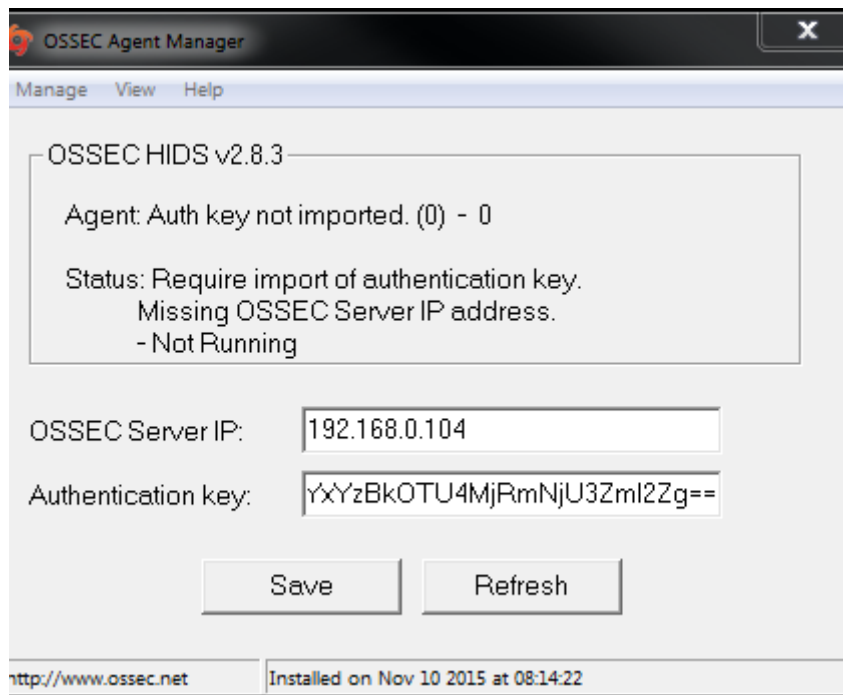


Figure 43 - Capricorn IP and Key

Back on the Capricorn appliance check that the agent has connected correctly.

- `sudo /var/ossec/bin/manage_agents`

Choose 'L' for list. Agent should be listed as follows:

```
*****
* OSSEC HIDS v2.8 Agent manager.          *
* The following options are available: *
*****
(A)dd an agent (A) .
(E)xtract key for an agent (E) .
(L)ist already added agents (L) .
(R)emove an agent (R) .
(Q)uit.
Choose your action: A,E,L,R or Q: L

Available agents:
  ID: 001, Name: SIEMonster, IP: 192.168.0.16

** Press ENTER to return to the main menu.
█
```

Figure 44 - List the agent and finish

7.4 SYSLOG

Network devices with remote syslog settings should be set to the Proteus appliance IP address. Syslogs are accepted on the ports

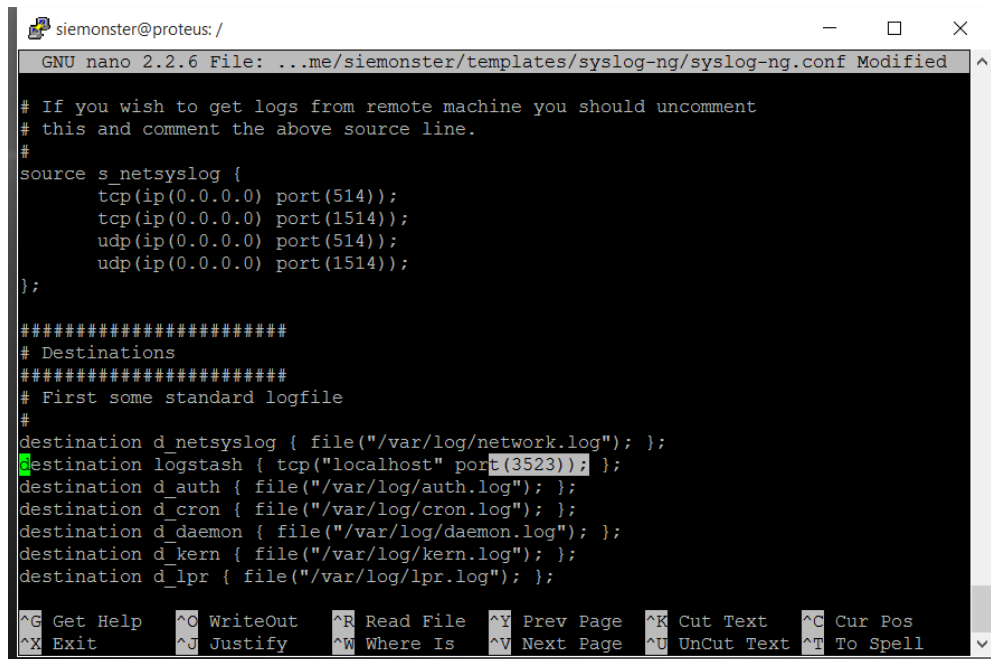
514 TCP	1514 TCP
514 UDP	1514 UDP

Parsing is handled by Logstash before forwarding to the ES cluster and forked to Graylog for Stream & Alerting sections. Syslog-NG is used to receive syslogs.

On the Proteus appliance edit the configuration file `/home/siemonster/templates/syslog-ng/syslog-ng.conf` and locate the line:

```
#destination logstash { tcp("localhost" port(3521)); };
```

- `pico /home/siemonster/templates/syslog-ng/syslog-ng.conf`
- Change the port for to **3523**:



```
siemonster@proteus: /
GNU nano 2.2.6 File: ...me/siemonster/templates/syslog-ng/syslog-ng.conf Modified
# If you wish to get logs from remote machine you should uncomment
# this and comment the above source line.
#
source s_netsyslog {
    tcp(ip(0.0.0.0) port(514));
    tcp(ip(0.0.0.0) port(1514));
    udp(ip(0.0.0.0) port(514));
    udp(ip(0.0.0.0) port(1514));
};

#####
# Destinations
#####
# First some standard logfile
#
destination d_netsyslog { file("/var/log/network.log"); };
destination logstash { tcp("localhost" port(3523)); };
destination d_auth { file("/var/log/auth.log"); };
destination d_cron { file("/var/log/cron.log"); };
destination d_daemon { file("/var/log/daemon.log"); };
destination d_kern { file("/var/log/kern.log"); };
destination d_lpr { file("/var/log/lpr.log"); };

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

Figure 45 - Port change for Syslog

Copy this file over with the command:

- `sudo cp /home/siemonster/templates/syslog-ng/syslog-ng.conf /etc/syslog-ng/syslog-ng.conf`

Restart the syslog-ng service as follows:

- `sudo service syslog-ng restart`

Check that UDP ports are in a listening state:

- `netstat -anu |grep 514`

Further configuration is covered in the following section under Logstash.

8 INPUTS

8.1 LOGSTASH

All event data is initially received and processed by Logstash. Incoming data is normalized and formatted for suitability for indexing into Elasticsearch.

Configuration files are located within the folder `/etc/logstash/config.d` on the Proteus appliance. Check out the SIEMonster logstash repo for the latest files - <https://github.com/siemonster>.

A working example file has been presented for processing OSSEC alerts from the Capricorn appliance: `01-ossec.conf` as a proof of concept. This file combines the input, filter & output for demonstration purposes.

The layout of these files is set out in sections as follows: Input {} Filter {} Output{}

Multiple configuration files are processed in an alphanumeric sequence. For this reason, the most efficient method of deployment is to separate the input, filter and output into separate files. Examples of this are provided in the `~/templates/logstash` folder. Each input is given a tag and/or type for identification. This tag/type is used through the pipeline 'input-filter-output' to ensure that each data source is processed correctly.

For the sake of simplicity and demonstration the deployed file `01-ossec.conf` contains the complete input, filter, output sections.

The following example illustrates how to configure Logstash to process Windows event logs:

On the Proteus appliance edit the example file `~/templates/logstash/00-inputs.conf` to match the port setup to receive these logs from NXLOG.

Deploy initially with the SSL lines commented to test functionality. Match with NXlog configuration shown in 7.1.1

```
Input {
  tcp {
    type => "events"
    port => 3522
    # ssl_enable => true
    # ssl_cert => "/etc/pki/tls/certs/logstash-forwarder.crt"
    # ssl_key => "/etc/pki/tls/private/logstash-forwarder.key"
    # ssl_verify => false
    codec => json_lines
    tags => ["windows", "eventlog"]
  }
}
```

Logstash configuration files may be tested as follows: First shutdown the Logstash service:

- `sudo service logstash stop`

Test the Windows events file:

- `sudo /opt/logstash/bin/logstash --configtest -f /etc/logstash/conf.d/10-windows-events-filter.conf`

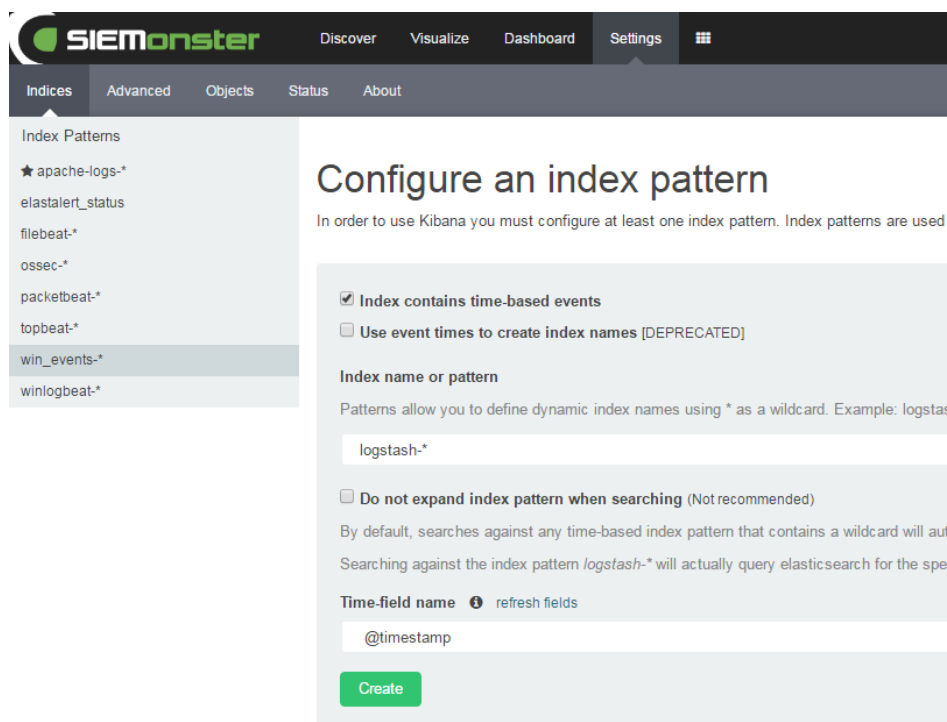
Note the double dash preceding configure. Successful test will result in 'Configuration OK'.

On failure a message will be displayed indicating the error. Once the test is successful restart the logstash:

- `sudo service logstash start`

Check the connection from NXLOG: `netstat -ant |grep 3522` (or configured port)

The next step is to register the index in Kibana. Assuming the index is named logstash-DATE as above:



The screenshot shows the Kibana 'Configure an index pattern' page. The left sidebar lists index patterns, with 'win_events-*' selected. The main content area has the title 'Configure an index pattern' and a subtitle 'In order to use Kibana you must configure at least one index pattern. Index patterns are used'. The configuration options include:

- ☒ Index contains time-based events
- ☐ Use event times to create index names [DEPRECATED]
- Index name or pattern:
- ☐ Do not expand index pattern when searching (Not recommended)
- Time-field name: refresh fields
-

Figure 46 - Kibana Windows Events

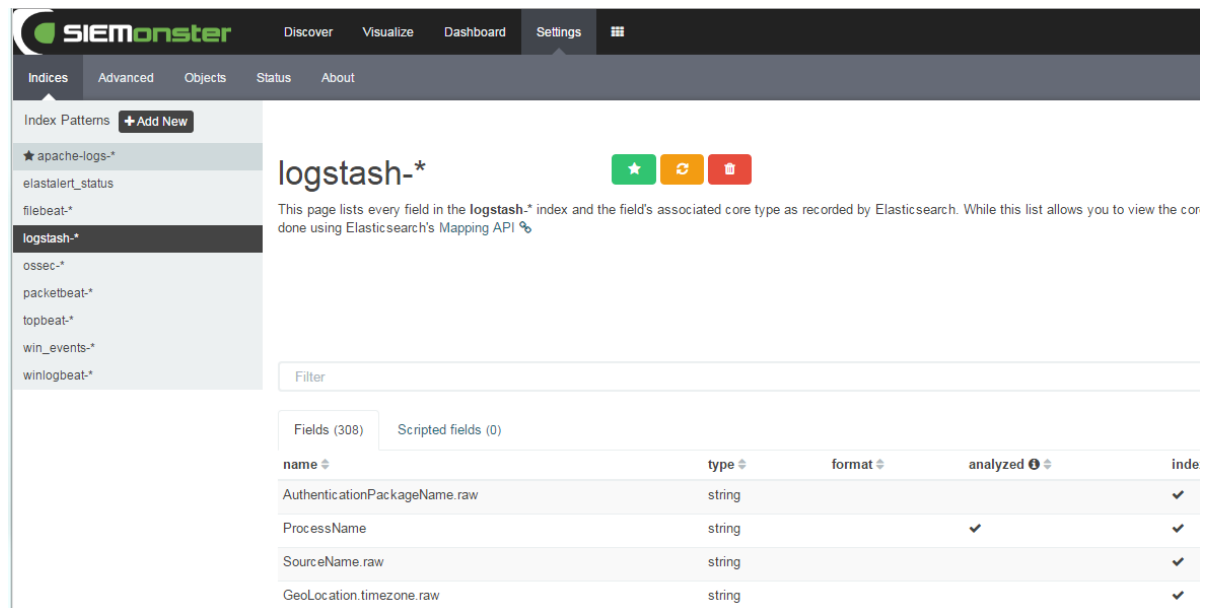


Figure 47 - Logstash Index

- Visit the Discovery menu and select the logstash-* index



Figure 48 - Visualization of the data

From here some sample saved searches, visualizations and dashboards in json format may be imported from a local folder. Some examples are available on <https://github.com/siemonster>.

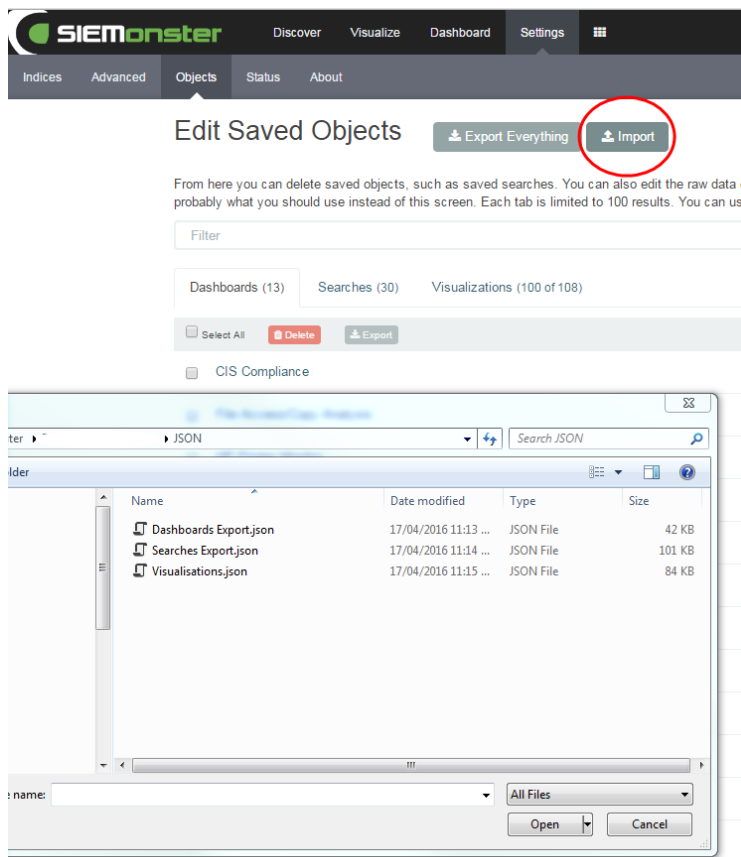


Figure 49 - Dashboard Configuration

Dashboards will now be available for Windows events under the load saved dashboards option:



Figure 50 - Open and save for Dashboards

8.2 TROUBLESHOOTING SYSLOGS

If you're not sure whether there is a firewall in your way, or that the syslog conf is working there are a few things you can do.

First telnet from the router/firewall subnet to Proteus on the port number i.e. telnet Proteus on the TCP port if it can see Proteus you're off to a good start for example

- telnet 192.168.0.103 514

To test syslogs, a test can be performed as follows, provided that the Syslog-ng changes were made as shown in section 7.4. This involves downloading a free tool that can be run on any Microsoft Windows box. The tool generates messages that get sent to Proteus and we can see if they are getting processed. This test file is called 15-syslog.conf.test

Download a syslog message generator on a windows machine.

<http://downloads.solarwinds.com/solarwinds/Release/FreeTool/Kiwi-SyslogGen-v2.zip>

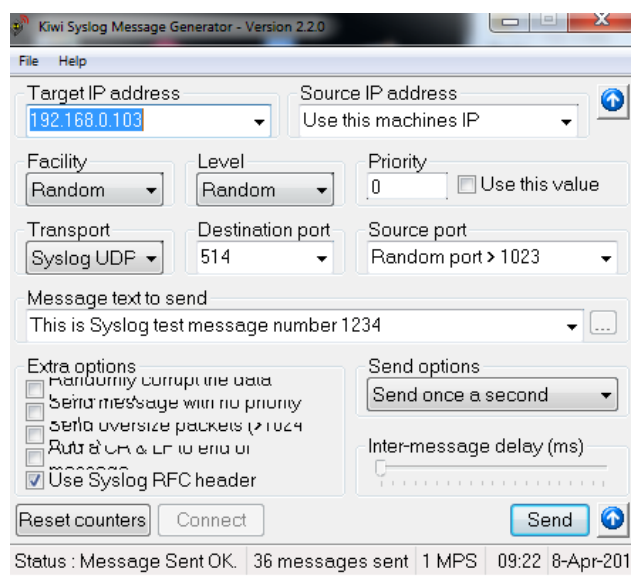


Figure 51 - Syslog Checker

Enter the Target IP address of the Proteus appliance and click Send.

Stop the logstash service:

- `sudo service logstash stop`

Run the following command:

- `sudo /opt/logstash/bin/logstash -f ~/templates/logstash/15-syslog.test`

Wait for messages to appear:

```
{
  "message" => "<138>Apr 28 09:13:33 192.168.137.1 SyslogGen This is Syslog test message number 1234",
  "@version" => "1",
  "@timestamp" => "2016-04-27T23:13:23.299Z",
  "host" => "127.0.0.1",
  "port" => 36828,
  "type" => "syslog",
  "tags" => [
    [0] "_grokparsefailure"
  ]
}
```

Figure 52 - Syslog received successfully

Press CTL-C to end the test.

Don't forget to start logstash when you are finished

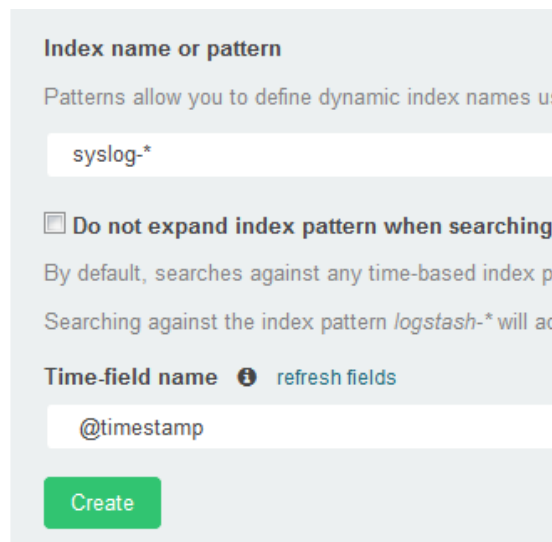
- `sudo service logstash start`

8.3 TROUBLESHOOTING CISCO ASA SYSLOGS

You have completed all the steps above, and you can see traffic hitting Proteus, but you're not sure how to check whether its appearing in Logstash. Use the steps below.

Send some syslogs to Proteus on UDP port 514 or 1514 from your ASA or other device. For example, log onto the device, or fail to log onto the device.

- Create a syslog index if you haven't already in Kibana - Settings:



The screenshot shows the Kibana 'Index name or pattern' settings. The 'Index name or pattern' field contains 'syslog-*'. Below it, there is a checkbox labeled 'Do not expand index pattern when searching' which is currently unchecked. A description below the checkbox states: 'By default, searches against any time-based index pattern will expand to match all time-based indices. Searching against the index pattern *logstash-** will automatically search all logstash indices.' The 'Time-field name' field contains '@timestamp' and has a 'refresh fields' link next to it. At the bottom is a green 'Create' button.

Notes:

To avoid a time travel paradox, edit the 03-multisyslog-filter.conf file and replace the timezone fields with your own timezone. <http://joda-time.sourceforge.net/timezones.html>

Otherwise events may appear in the future and will not be visible in Kibana.

e.g. `sudo pico /etc/logstash/conf.d/03-multisyslog-filter.conf`

```

syslog_pri { }
date {
  match => [ "syslog_timestamp", "MMM d HH:mm:ss",
    timezone => "Australia/Melbourne"
  }
}

```

Figure 53 - Time zone mods

Debugging:

To check that logs are reaching Proteus, edit the file 99-outputs.conf:

- `sudo pico /etc/logstash/conf.d/99-outputs.conf`

Add the following line:

- `stdout { codec => rubydebug }`

```
output {
  if "ossec" in [tags] {
    elasticsearch {
      hosts => ["localhost:9200"]
      index => "ossec-%{+YYYY.MM.dd}"
      document_type => "ossec"
      template => "/etc/logstash/elastic-ossec-template.json"
      template_name => "ossec"
      template_overwrite => true
    }
  }
  if "main_syslog" in [tags] {
    elasticsearch {
      hosts => ["localhost:9200"]
      index => "syslog-%{+YYYY.MM.dd}"
    }
    stdout { codec => rubydebug }
  }
}
```

Figure 54 - Ruby troubleshooting

Run logstash from the command line:

- `sudo service logstash stop`
- `sudo /opt/logstash/bin/logstash -f /etc/logstash/conf.d/`

Send some syslogs to Proteus: (failed or normal logon to Cisco ASA)

```
root@proteus:/etc/logstash/conf.d# /opt/logstash/bin/logstash -f /etc/logstash/conf.d/
Settings: Default pipeline workers: 2
Logstash startup completed
{
  "@version" => "1",
  "@timestamp" => "2016-05-02T05:40:02.000Z",
  "host" => "127.0.0.1",
  "port" => 33892,
  "type" => "syslog",
  "tags" => [
    [0] "main_syslog",
    [1] "Firewall",
    [2] "ASA"
  ],
  "syslog_timestamp" => "May  2 15:40:02",
  "syslog_host" => "192.168.137.1",
  "received_at" => "May  2 15:40:02",
  "syslog_severity_code" => 5,
  "syslog_facility_code" => 1,
  "syslog_facility" => "user-level",
  "syslog_severity" => "notice",
  "action" => "Denied",
  "protocol" => "ICMP",
  "icmp_type" => "3",
  "icmp_code" => "3",
  "src_ip" => "104.70.220.229",
  "SourceGeo" => {
    "ip" => "104.70.220.229",
    "country_code2" => "US",
    "country_code3" => "USA",
    "country_name" => "United States",
    "continent_code" => "NA",
    "region_name" => "MA",
    "city_name" => "Cambridge",
    "postal_code" => "02142",
    "latitude" => 42.3625999999999986,
    "longitude" => -71.0843,
    "dma_code" => 506,
    "area_code" => 617,
    "timezone" => "America/New_York",
    "real_region_name" => "Massachusetts",
    "location" => [
      [0] -71.0843,
      [1] 42.3625999999999986
    ]
  }
}
```

Figure 55 - Syslogs received command line access

CTRL +C to end.

If all is well:

- `sudo service logstash start`

8.4 GRAYLOG

Graylog is a highly efficient log management system that is used within **SIEMonster** to forward log data into Streams and subsequently the Alerting mechanism. Graylog also provides many other features, some of which will be incorporated into **SIEMonster** in the coming months. The reasoning behind Graylog as an included module can be summarised as follows:

1. User and role based access with LDAP integration
2. User friendly Stream and Alert Modules with outputs for Slack, Email, Hipchat etc.
3. Extensibility for further log data analysis

The following example illustrates the configuration of an input for Windows events output from Logstash on the Proteus appliance.

- Choose Inputs from the System/Inputs menu

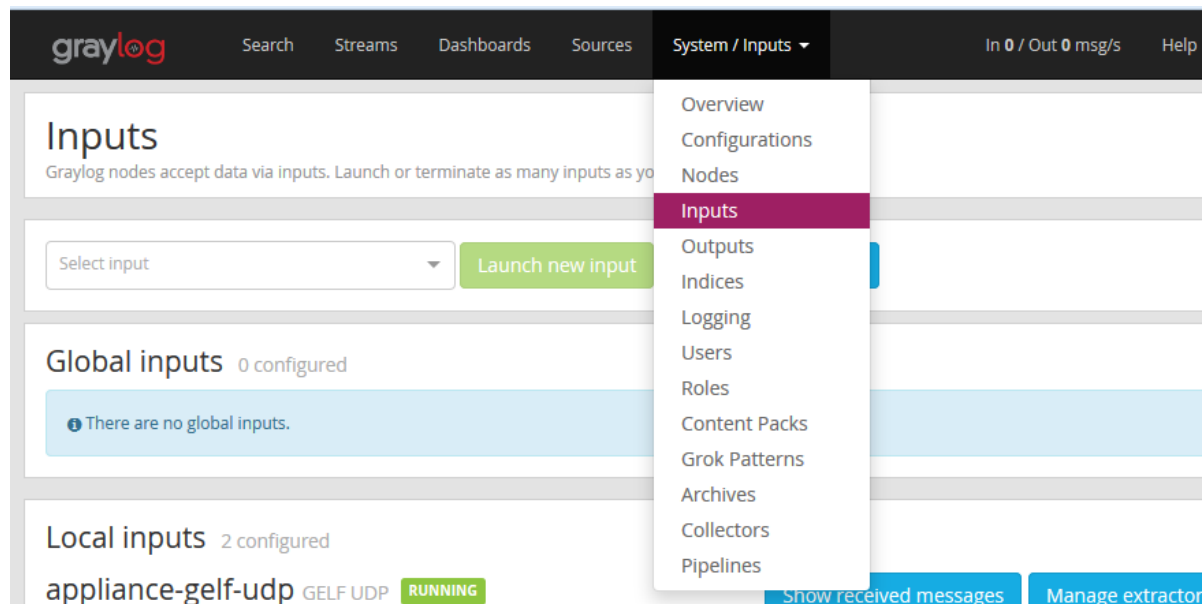


Figure 56 - Graylog Input

Select Gelf UDP and click 'Launch new input'



Figure 57 – Gelf setup

- Choose the Capricorn node with port 12202 and save.

Launch new *GELF* UDP input ×

Title

Select a name of your new input that describes it.

☐ Global

Should this input start on all nodes

Node

On which node should this input start

Bind address

Address to listen on. For example 0.0.0.0 or 127.0.0.1.

Port

Port to listen on.

Receive Buffer Size (optional)

The size in bytes of the `recvBufferSize` for network connections to this input.

Override source (optional)

The source is a hostname derived from the received packet by default. Set this if you want to override it with a custom string.

Figure 58 - Gelf Capricorn and port details

Input should be displayed as follows:

Local inputs 3 configured

Windows Events GELF UDP RUNNING

On node ● c4d55199 / capricorn

```

override_source:
recv_buffer_size: 1048576
bind_address: 0.0.0.0
port: 12202

```

Throughput / Metrics

1 minute average rate: 0 msg/s

Network I/O: ~0B ~0B (total ~0B ~0B)

Empty messages discarded: 0

Show received messages Manage extractors Stop input More actions ▾

9 ALERTING

9.1 GRAYLOG

Following on from the previous example setting up the Graylog input for Windows events. Before you begin goto slack and create a free slack account.

This input is first piped into a Stream. Streams – Create Stream:

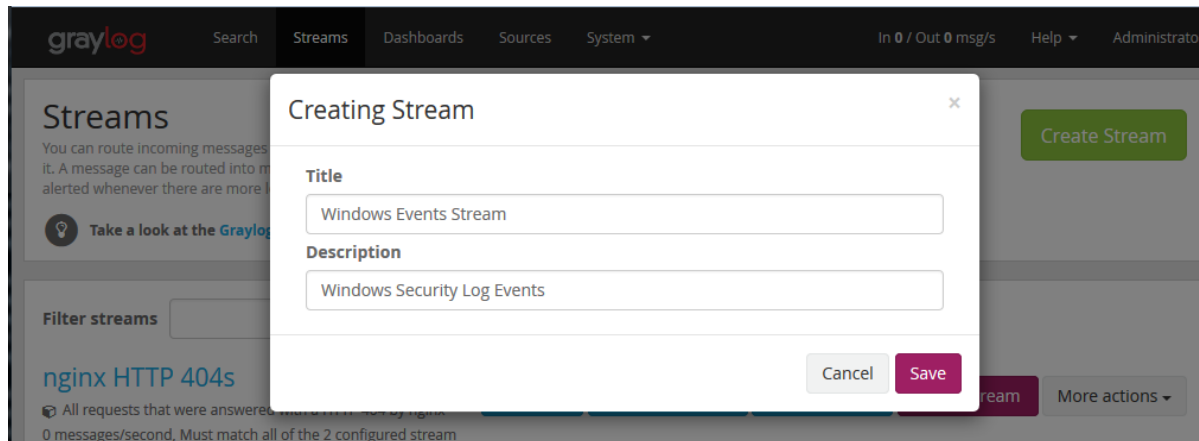


Figure 59 - Stream Creation

Enter a suitable title and description for the Stream.

Edit the rules for the Stream

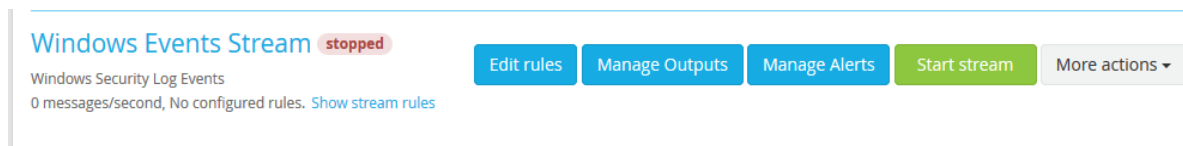
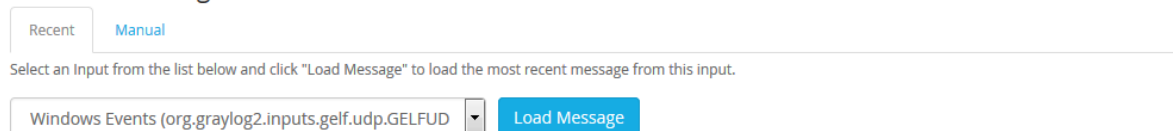


Figure 60 - Windows event stream Edit Rules

Choose the input previously configured

1. Load a message to test rules



2. Manage stream rules

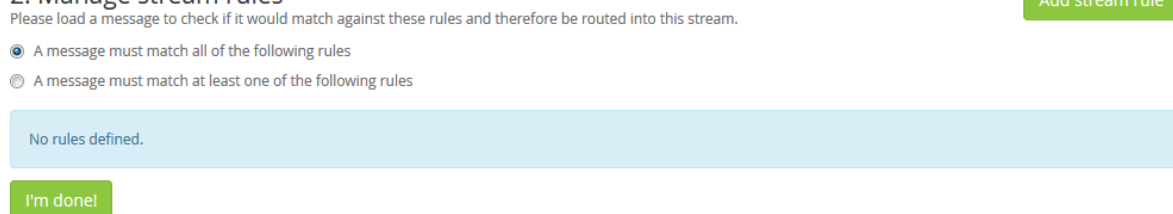


Figure 61 - Input previously configured in Graylog

- Add Stream rule for EventID equal to 4771

New Stream Rule

Field

EventID

Type

match exactly

Value

4771

☐ Inverted

Result: Field EventID must match exactly 4771

The server will try to convert to strings or numbers based on the matcher type as good as it can.

[Take a look at the matcher code on GitHub](#)

Regular expressions use Java syntax. [?](#)

Cancel



Save

Figure 62 - Adding a Rule for alerting in the stream

2. Manage stream rules

Please load a message to check if it would match against these rules and therefore be routed into this stream.

- ☒ A message must match all of the following rules
- ☐ A message must match at least one of the following rules

  EventID must match exactly 4771

I'm done!

Figure 63 - Confirmation of stream rule

- Click I'm done.
- Go back to Streams and Select Manage Alerts

Windows Events Stream stopped

Edit rules

Manage Outputs

Manage Alerts

Start stream

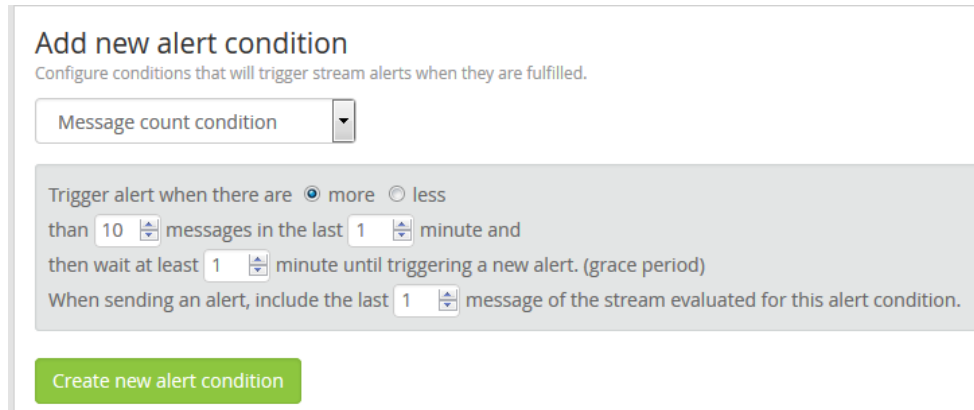
More actions ▾

Windows Security Log Events

0 messages/second, Must match all of the 1 configured stream rule(s). [Show stream rules](#)

Figure 64 - Managed alert in Graylog

- Select Message count condition. For this example, alert when more than 10 logon failures/minute



Add new alert condition
Configure conditions that will trigger stream alerts when they are fulfilled.

Message count condition

Trigger alert when there are ☒ more ☐ less
than messages in the last minute and
then wait at least minute until triggering a new alert. (grace period)
When sending an alert, include the last message of the stream evaluated for this alert condition.

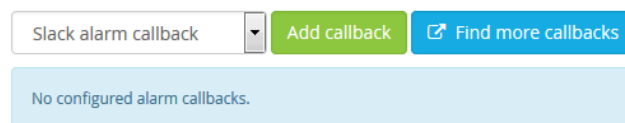
Create new alert condition

Figure 65 - Setting up alert condition

- Click Create new alert condition
- Select Callbacks – Slack alarm callback

Callbacks

The following callbacks will be performed when this stream triggers an alert.



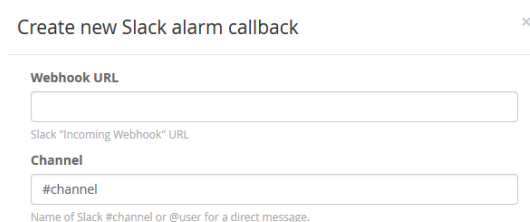
Slack alarm callback

Add callback Find more callbacks

No configured alarm callbacks.

Figure 66 - Slack Alarm callback

- Add callback – Only the Webhook URL and Channel is required initially



Create new Slack alarm callback

Webhook URL

Slack "Incoming Webhook" URL

Channel


#channel

Name of Slack #channel or @user for a direct message.

Figure 67 - Webhook URL in Graylog alerting

To find or create the webhook URL for your #Slack account refer to the App Directory area:

Browse apps > Custom Integrations > Incoming WebHooks > Configurations on SIEMonster > New configuration



Incoming WebHooks

Send data into Slack in real-time.

Post to Channel

Start by choosing a channel where your Incoming Webhook will post messages to.

or [create a new channel](#)

Add Incoming WebHooks integration

By creating an incoming webhook, you agree to the [Slack API Terms of Service](#).

This will provide a URL of the form <https://hooks.slack.com/services/XXXX/XXXX/XXX>

- Use this URL and the required channel in the previous form.

Result:

Slack alarm callback
Executed once per triggered alert condition.

add_attachment:	true
channel:	#siemonster_alerts
color:	#FF0000
graylog2_url:	<empty>
icon_emoji:	<empty>
icon_url:	<empty>
link_names:	true
notify_channel:	false
short_mode:	false

Figure 68 - Alarm call back result once configured

Additionally, an email address can be added as a secondary alert but it is not mandatory.

This alert is now ready to go. Click on Send test alert to check all is well in Slack you can use a pc, or mobile device for slack.

9.2 ELASTALERT

A secondary alerting system has been provided offering alerts to Email, Slack, Hipchat, Victor Ops, Pager Duty etc. This system is used for anomalies, spikes, or other patterns of interest from data in Elasticsearch.

To get started with Elastalert there are some sample rules provided on the Proteus appliance in the folder `/opt/elastalert/example_rules/`.

A good rule to start with is the frequency rule.

1. Ensure the `es_host` is set to `localhost` & `es_port` 9200.
2. Check the index setting, `logstash-*` is default
3. Check the timeframe
4. Adjust the filter, e.g. term: `EventID: "4771"`
5. Enter the required email address.

On completion test the rule with the following command:

- `sudo elastalert-test-rule /opt/elastalert/example_rules/ example_frequency.yaml`

If the test is successful, the rule file can be moved to `/opt/elastalert/rules` and will be automatically processed by Elastalert.

9.3 LOGSTASH

The 3rd alerting option is the Logstash system which can be useful for monitoring specific events. As an example, adding the following to the output section of a the Logstash Windows event configuration file will alert on a new security enabled group being created:

```
if [EventID] == 4727 {  
  email {  
    from => "siem@siemonster.com"  
    subject => "%{EventDesc}"  
    to => "alerts@siemonster.com"  
    cc => "tickets@siemonster.com"  
    via => "sendmail"  
    body => "Alert - %{SubjectUserName} has created a new security enabled global group  
    %{SamAccountName} %{message}"  
    options => { "location" => "/sbin/sendmail"  
  }  
}
```

10 OSINT

10.1 CRITICAL-STACK-INTEL

Providing vectors for translation tables in the form of known malicious domains used for Phishing, C&C hosts, TOR endpoints and known compromised hosts. This open source intelligence is then used to identify/detect such hosts contained within incoming security log data.

The critical-stack-intel client application has been pre-installed and just requires an api to retrieve the relevant data.

The process begins with registering an account with Critical Stack:

https://intel.criticalstack.com/user/sign_up

- Once logged on go to the Collections tab and Create New Sensor

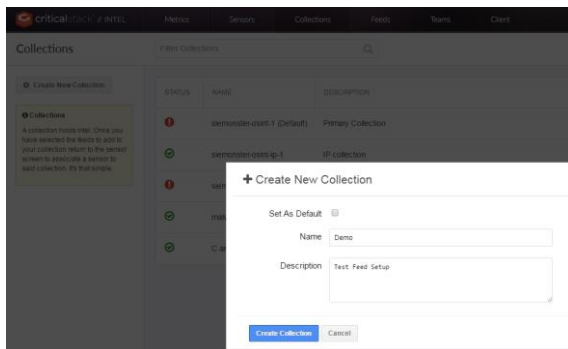


Figure 69 - Sensor Creation



Figure 70 - Demo Sensor

- Click on the new collection and the Add More Feeds

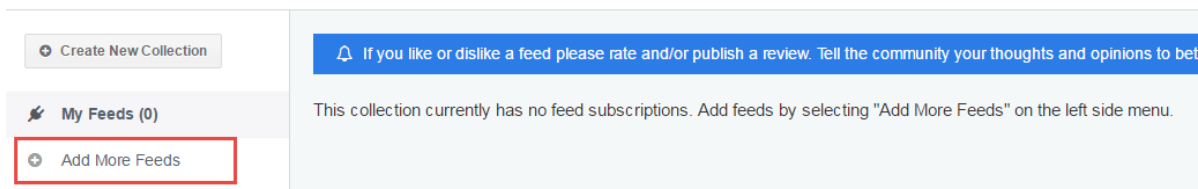


Figure 71 - Adding OSINT Feeds

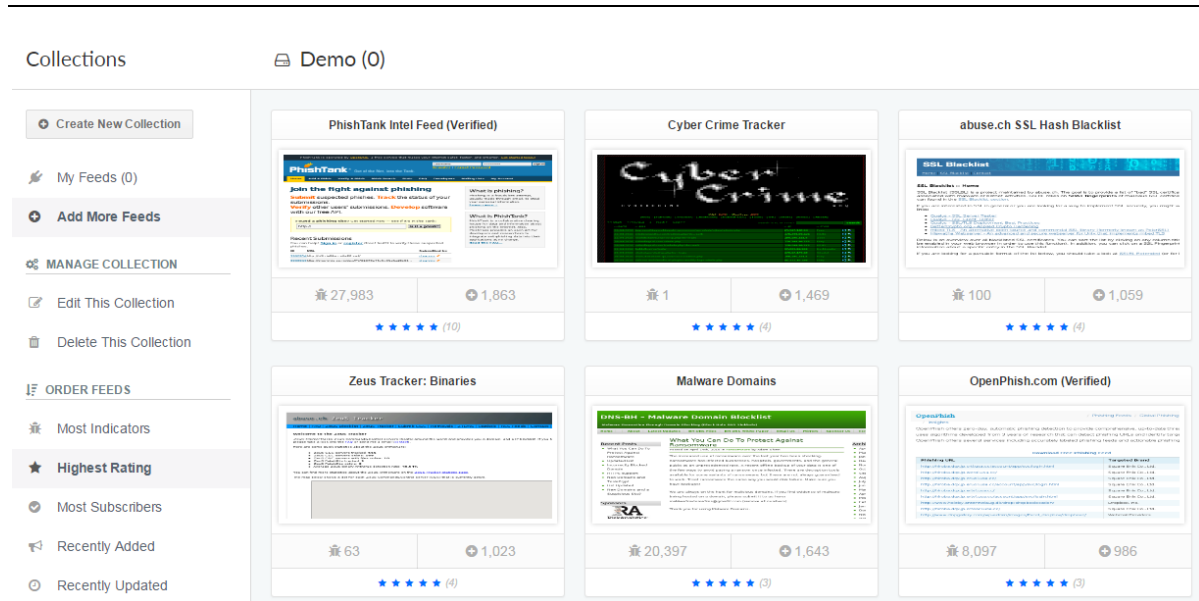


Figure 72 - Adding Feeds to OSINT

- Choose your required feeds containing domains and/or phishing URLs and subscribe.

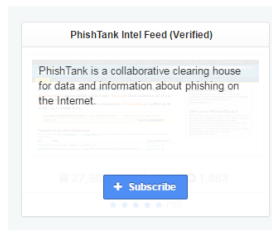


Figure 73 - Sample Subscribe confirmation

- Over on the Sensors tab, click Create New Sensor selecting the Collection previously chosen

+ Create New Sensor

Collection

Name

Figure 74 - Selecting your previous demo-sensor

A new line will be displayed showing the new Collection.

- Click on the copy icon to the left of the api key and paste into a text editor for the next stage




	Test	Demo	 6d36743b-f82d-4d64-444c-8... N/A	1	0	
<div>Click to copy me.</div>						

Figure 75 - Icon and Signature

On the Proteus appliance use the following command to initiate the Critical Stack client:

- `sudo critical-stack-intel api xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx (api key)`

The chosen feeds will then be downloaded.

```
siemonster@proteus:~$ sudo critical-stack-intel api 6d36743b-f82d-4d
2016/04/24 18:46:52 API key created successfully.
critical-stack 18:46:52 [INFO] Pulling feed list from the Intel Marketplace.
critical-stack 18:46:54 [INFO] Downloading feed information. Run with the '--debug' flag for more information.
1 / 1 [=====]
critical-stack 18:46:57 [INFO] Creating master file: master-public.bro.dat. Please wait.
critical-stack 18:46:57 [INFO] Master file created successfully.
critical-stack 18:46:58 [INFO] Intel files located at: /opt/critical-stack/frameworks/intel
critical-stack 18:46:58 [INFO] API Requests Remaining: 998 of 1000/minute
```

Figure 76 - Critical Stack API download

The downloaded feed can be found at `/opt/critical-stack/frameworks/intel/master-public.bro.dat`

Intended to be used with the Bro IDS system, this file is parsed by Logstash within **SIEMonster** for matching and visualization of incoming event log data.

Bro IDS installation is also supported and relevant Logstash configuration files can be found in the `templates/logstash` folder on the Proteus appliance.

10.2 MALICIOUS IP

A python script, `maliciousIP.py`, has been provided on the Proteus appliance as a secondary collection point for the latest malicious IP addresses. This script can be run as a daily cron job and it is located in the folder `/etc/logstash` on the Proteus appliance. When run, a Logstash translation table is output to the same folder, `maliciousIP.yaml`, which can be used in conjunction with Logstash to match geoip fields listed in the file.

11 HIDS

11.1 RULESETS

OSSEC rulesets can be updated manually, but running on a weekly schedule via a cronjob is recommended.

To perform a manual update, issue the following command on the Capricorn appliance:

- `sudo /var/ossec/update/ruleset/ossec_ruleset.py -s`

To add as a cron job first run

- `sudo crontab -e`

Add the following line at the end of the file:

- `@weekly root cd /var/ossec/update/ruleset && ./ossec_ruleset.py -s`

11.2 MANAGEMENT

OSSEC management is handled by the ossec-control feature: `/var/ossec/bin/ossec-control`

Usage: `/var/ossec/bin/ossec-control {start|stop|restart|status|enable|disable}`

More features will be used in coming weeks for Kibana integration and agent deployment/monitoring.

12 INCIDENT RESPONSE

FIR (Fast Incident Response) is a cybersecurity incident management platform designed with agility and speed in mind. It allows for easy creation, tracking, and reporting of cybersecurity incidents. For those companies that do not have Incident Response Software or ticketing system, this is an ideal system. For those that do this feature can be turned off. However, consider using this tool for its automated ticketing of alerts and forwarders into your existing ticketing system.

12.1 ADMINISTRATION

Using a web browser to <http://192.68.0.104/admin/> (or configured Capricorn IP address) and log in with the credentials admin/admin.

Once you're logged in

- Click on the Add button in the Users row. Fill-in the fields and click on save. On the next screen, go to the Groups section, click on "incident handlers", and on the arrow to add it to the column "Chosen groups". Click on Save at the bottom of the screen.

A standard user 'dev' has already been created with a password of 'dev'.

Next, you need to add a profile to the user. Still logged in as the super-user,

- Click on "Add" in the "Profiles" row of the admin panel. Select the created user and chose the number of incidents they will see in their view. Click "Save", and log out.

Creating labels: (Sample labels have already been created).

Labels are used to populate choices in some incident fields:

Detection source

Actions taken

Actor

Plan

FIR uses these "label groups" to know how where to map the labels.

The four mandatory label groups are detection, action, actor, and plan. You can add these through the admin interface in the "Label groups" section.

You should then specify options for each of the labels. Remember that an incident has a mandatory detection field, and comments have a mandatory action field; You'll need to populate at least those two.

12.2 USAGE

New Event:

Login as a standard user, you can start off with user 'dev', password 'dev'.

Click on the New event button to go to the event creation form:

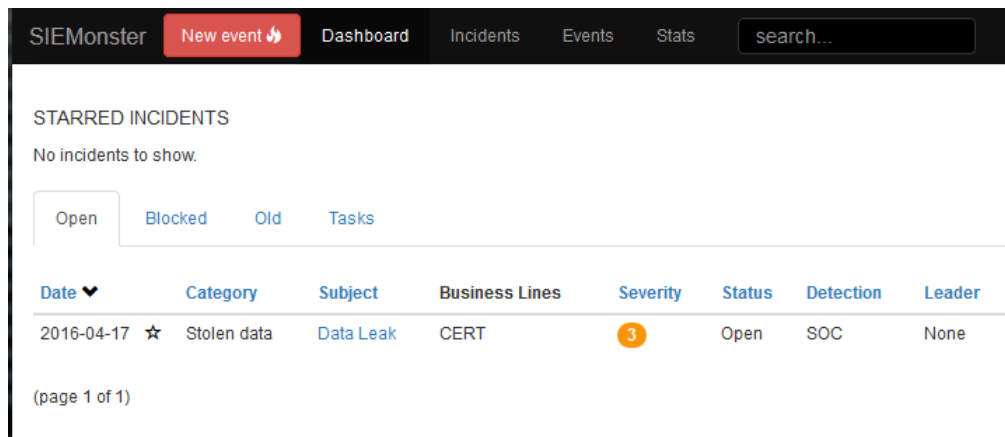


Figure 77 - FIR New event creation

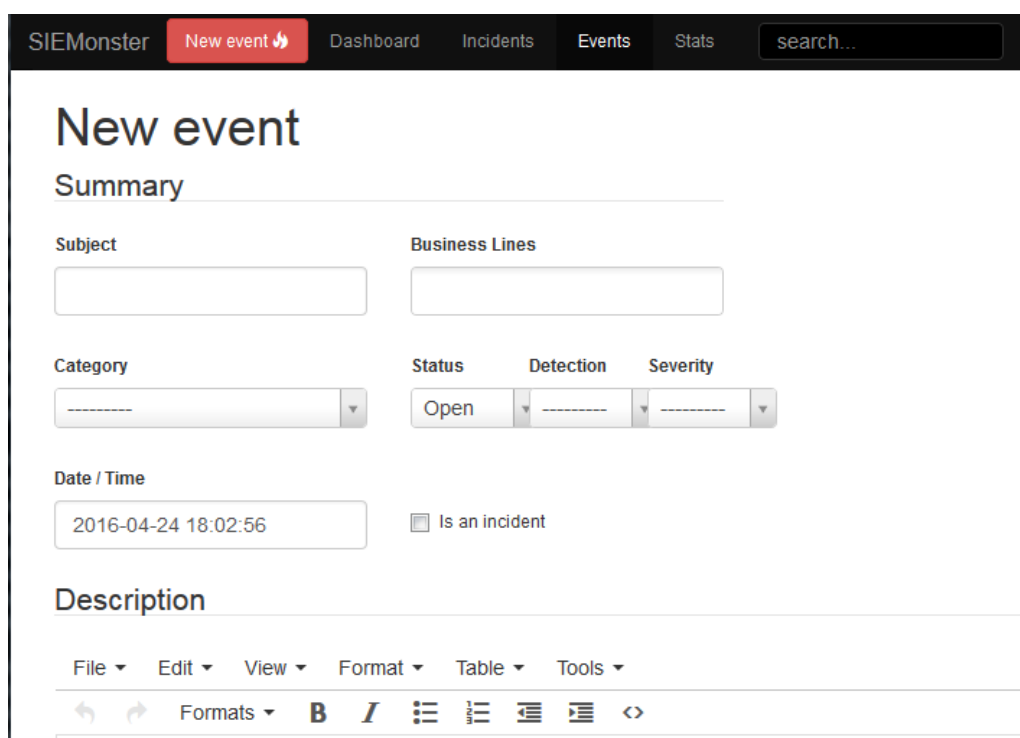


Figure 78 - FIR Events

Here is the description of the available fields:

Subject: short description of your incident. The one that will appear on event tables.

Business Lines: entities concerned by this incident. You choose what you make of business lines: internal department, customers, etc.

Category: category of the incident (ex: phishing, malware). Categories are also customizable in the admin panel.

Status: can take three values: Open, Closed and Blocked. These are all labels defined in the admin panel

Detection: how the incident was detected. Default values: CERT, External. These values can be changed in the admin panel in the labels section

Severity: from 1 to 4.

Date / Time: date and time of the incident

Is an incident: differentiates between an event and an incident

Description: free-form text describing the event

When you are dealing with an incident, the following additional fields are available. These fields are only used for display and statistics:

Actor: who is the leader on this incident management? Default values are CERT and Entity

Plan: what is the named remediation plan used?

Confidentiality: from C0 to C3

Click on Save, and you will be redirected to the incident details page.

12.3 AUTOMATED TICKETING

Automated ticketing is based on a provided python script which is located within the scripts/fir directory on the Proteus appliance, fir_email.py with a template email.txt This feature is being further developed due to the new pipeline features offered by the Graylog event management console and new options will be available in coming weeks.

13 FREQUENTLY ASKED QUESTIONS

13.1 CONFIGURATION / INSTALLATION

Is there a license requirement for SIEMonster?

There is no license needed, you can have as many nodes and ingest as much data as you want.

SIEMonster is a collection of tools licensed under the [GNU General Public License](https://www.gnu.org/licenses/gpl-3.0.html).

Where is the latest FAQ's

<https://siemonster.zendesk.com/hc/en-us>

13.2 BACKUP/SCALING

Backup to Amazon S3 storage is available. The Elasticsearch Cloud AWS plugin is preinstalled. An example script can be found in templates/S3 on the Proteus appliance.

Edit this script to suit region etc.

Scaling out the ELK stack can be achieved by simply performing another import of the original VM OVA file and modifying the Kraken/Tiamat scripts to suit. A scaling guide and install instructions will be included in the next revision of this guide.

13.3 TROUBLESHOOTING

How do I know if my ELK stack is up and running?

2 Elasticsearch plugins have been pre-installed to monitor stack health and detailed statistics.

On the Proteus appliance the URLs are as follows:

http://192.168.0.103:9200/_plugin/kopf

http://192.168.0.103:9200/_plugin/hq

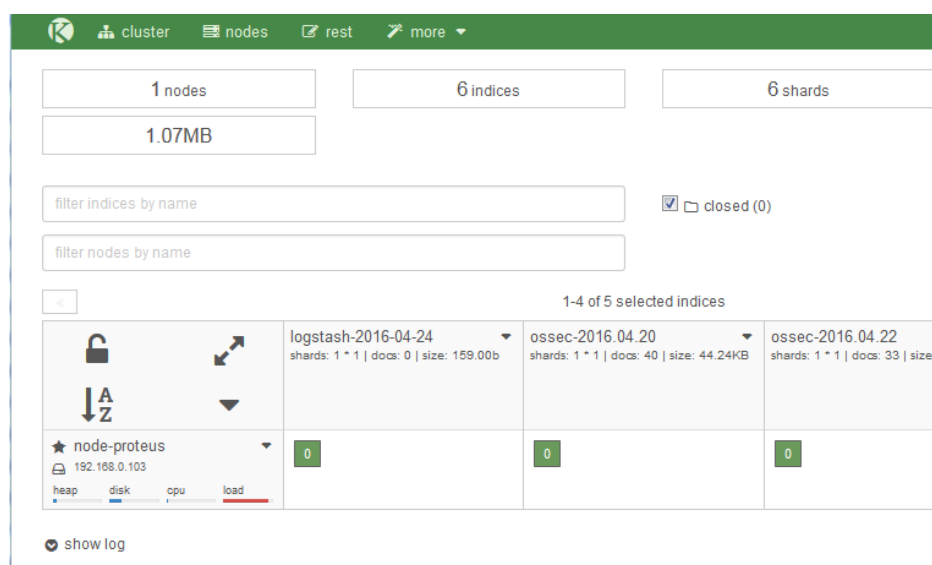


Figure 79 - Health Check view

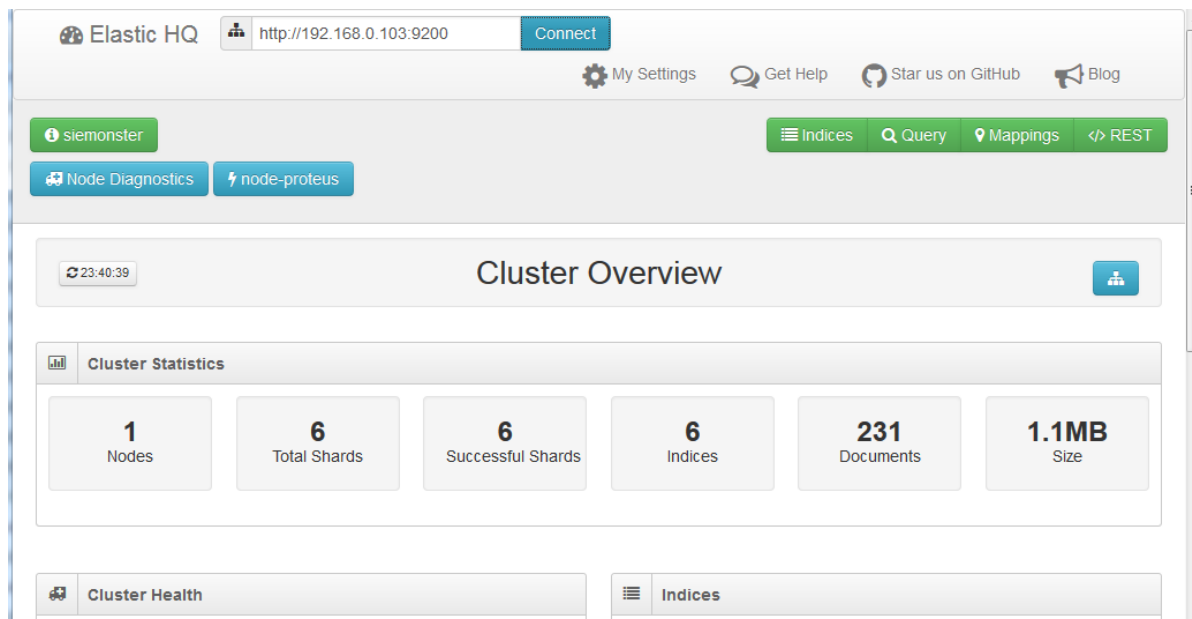


Figure 80 - Cluster Health checking

On the command line at the Proteus appliance cluster health may be found using the command:

- `curl http://localhost:9200/_cluster/health?pretty`

```
siemonster@proteus:~$ curl http://localhost:9200/_cluster/health?pretty
{
  "cluster_name" : "siemonster",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 1,
  "number_of_data_nodes" : 1,
  "active_primary_shards" : 6,
  "active_shards" : 6,
  "relocating_shards" : 0,
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
}
```

Figure 81 - Curl output

I cannot see any incoming event logs/messages.

This issue usually arises when the message timestamp is incorrect or the remote device has the wrong time set. If you adjust the range back to 7 days, the messages might be visible.



14 CHANGING PASSWORDS

Once you are happy with your SIEM and its in production it's time to lockdown the system. This includes changing all the default passwords. Below is a simple guide on changing the passwords on all the systems. Place these passwords in a safe place.

Linux passwords for root & siemonster on all servers:

- `sudo passwd root – sudo passwd siemonster`

Kibana password:

- `sudo htpasswd -c /etc/nginx/htpasswd.users siemonster`

FIR password: Login as admin to the FIR web interface and change within the user section

Graylog: On the Capricorn appliance run the following command:

- `sudo graylog-ctl set-admin-password <password> - change admin password`
- `sudo graylog-ctl set-admin-username <username> - change admin username`

MySQL: On the Proteus appliance run the following command:

`mysqladmin -u root -p'siemonster' password newpassword`

In this case 'siemonster' is the old (default password) and 'newpassword' is the new password.