SolidFire Stats Collection with Grafana and the Netapp Docker Volume Plugin

*All Your Graphs Are Belong to Us*

High level steps

1. (optional) Install and configure Trident for persistent storage
2. Clone the jedimt/sfcollector repository
3. Setup and run the collector

Overview

The SolidFire collector is a fully packaged metrics collection and graphing solution for Element OS 9+ and is based on four Docker containers.

- SFCollector-> runs a python script to scrape results from SolidFire clusters
- vsphere-graphite -> vSphere stats collector, written in Go
- Graphite database -> keeps all time series data from the SFCollector
- Grafana -> Graphing engine

The collector stores metrics in graphite and presents those metrics through a set of pre-configured Grafana dashboards. Optionally, Trident/NDVP can be used for persistent storage of metrics on a NetApp system.
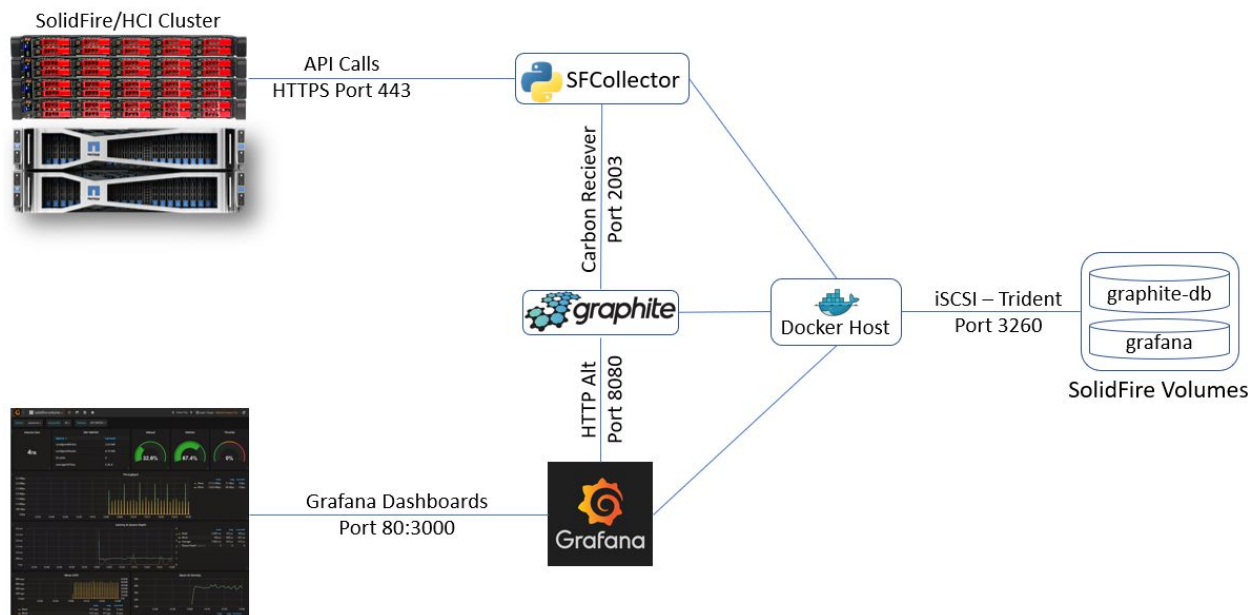
Trident Installation and Configuration

Prerequisites

Operating System	- Debian 8 and above - Ubuntu 14.04 and above, 15.10 and above if using iSCSI multipathing - CentOS 7.0 and above - RHEL 7.0 and above - CoreOS 1353.8.0 and above
Element OS Version	9.0+
Docker Version	17.03+

Note: Ubuntu 16.04.1 LTS is used in a multipath configuration for this example.

1. Install packages

```
sudo apt-get install -y open-iscsi lsscsi sg3-utils multipath-tools scsitools
```

2. Enable multipathing

```
sudo tee /etc/multipath.conf <<-'EOF'
defaults {
    user_friendly_names yes
    find_multipaths yes
}
EOF

sudo service multipath-tools restart
```

3. Start services

```
sudo service open-iscsi start
sudo service multipath-tools start
```

4. Create a location to store the NDVP configuration files

```
sudo mkdir -p /etc/netappdvp
```

5. Create the configuration file for SolidFire

```
cat << EOF > /etc/trident/config.json
{
    "version": 1,
    "storageDriverName": "solidfire-san",
    "Endpoint": "https://admin:solidfire@10.193.136.240/json-rpc/9.0",
    "SVIP": "10.193.137.240:3260",
    "TenantName": "docker",
```

```
        "InitiatorIFace": "default",
        "Types": [
            {
                "Type": "docker-default",
                "Qos": {
                    "minIOPS": 1000,
                    "maxIOPS": 2000,
                    "burstIOPS": 4000
                }
            },
            {
                "Type": "docker-app",
                "Qos": {
                    "minIOPS": 4000,
                    "maxIOPS": 6000,
                    "burstIOPS": 8000
                }
            },
            {
                "Type": "docker-db",
                "Qos": {
                    "minIOPS": 6000,
                    "maxIOPS": 8000,
                    "burstIOPS": 10000
                }
            }
        ]
}
EOF
```

6. Install the NDVP plugin

```
docker plugin install --grant-all-permissions --alias netapp
netapp/trident-plugin:18.01 config=/etc/netappdvp/config.json
```

7. Verify the plugin installed and is enabled

```
docker plugin list
ID                      NAME                    DESCRIPTION
ENABLED
047ac2d0663f            netapp:latest           Trident - NetApp Docker
Volume Plugin    true
```

8. Create docker volumes to be used for Graphite and Grafana persistent storage.

```
# Create Graphite docker volume
docker volume create -d netapp --name graphite-db -o type=docker-db
-o size=50G

#Create Grafana docker volume
docker volume create -d netapp --name grafana -o type=docker-app -o
size=10G
```

```
#show volume
docker volume list --filter 'driver=netapp:latest'
DRIVER            VOLUME NAME
netapp:latest     grafana
netapp:latest     graphite-db
```

Graphite and Collector Setup

Bringing Up the Containers

1. Install docker-compose if not installed

```
apt install docker-compose
```

2. Clone the jedimt/sfcollector repo from git

```
mkdir -p /opt/github/sfcollector
git clone https://github.com/jedimt/sfcollector
/opt/github/sfcollector
```

3. Run the bootstrap.sh file to create the password for basic_auth. This will be used to secure the nginx web component

```
/opt/github/sfcollector/bootstrap.sh
```

4. Edit the docker-compose.yml file and specify the persistent data volumes to use for graphite and grafana as well as the password to secure the Grafana web interface

```
version: "2"
services:
  graphite:
    build: ./graphiteconfig
    restart: always
    ports:
        - "8080:80"
        - "8125:8125/udp"
        - "8126:8126"
        - "2003:2003"
        - "2004:2004"
    volumes: #Trident or local volumes for persistent storage
        - graphite-db:/opt/graphite/storage/whisper
    networks:
        - net_sfcollector

  grafana:
    image: grafana/grafana
    restart: always
    ports:
        - "80:3000"
    volumes: #Trident or local volumes for persistent storage
        - grafana:/var/lib/grafana
    networks:
        - net_sfcollector
    environment: #Set password for Grafana web interface
        - GF_SECURITY_ADMIN_PASSWORD=<your password>
```

```
  sfcollector-alpine:
    build: ./collector-alpine
    restart: always
    networks:
        - net_sfcollector

  vsphere-collector:
    build: ./vsphere-graphite
    restart: always
    networks:
        - net_sfcollector
    depends_on:
        - graphite

networks:
  net_sfcollector:
    driver: bridge

volumes:
  graphite-db:
    external: true
  grafana:
    external: true
```

5. Modify the `/opt/github/sfcollector/collector-alpine/wrapper.sh` script with the appropriate SolidFire cluster MVIP, username and password. If the graphite container name was changed, also specify the new hostname

```
#!/usr/bin/env bash
while true
do
/usr/bin/python /solidfire_graphite_collector_v3.py -s 10.193.136.240
-u admin -p solidfire -g graphite &

#To monitor additional clusters add another line
# /usr/bin/python /solidfire_graphite_collector_v3.py -s
172.27.40.205 -u admin -p solidfire -g graphite &
sleep 60
done
```

6. Edit the `/opt/github/sfcollector/graphiteconfig/storage-schemas.conf` file to adjust the retention period for the NetApp stats if desired. By default the following retention is set which keeps 1 minute stats for 7 days, 5 minute stats for 28 days and 10 minute stats for a year. If you are also collecting stats from vCenter, add the [vsphere] section as well.

```
[netapp]
pattern = ^netapp\.*
```

```
retentions = 1m:7d,5m:28d,10m:1y

[vsphere]
pattern = ^vsphere\.*
retentions = 1m:7d,5m:28d,10m:1y
```

7. Edit the /opt/github/sfcollector/vsphere-graphite/vsphere-graphite.json file and add your
    vCenter details.

```
  "Domain": ".<your domain FQDN>",
  "Interval": 60,
  "FlushSize": 100,
  "VCenters": [
    { "Username": "administrator@sflab.local", "Password": "<your password>",
"Hostname": "<vcenter host name>.<your domain FQDN>" }
  ],
  "Backend": {
    "Type": "graphite",
    "Hostname": "graphite",
    "Port": 2003
  },
  "Metrics": [
    { … <removed for brevity>
```

8. Bring up the containers using docker-compose. This will take several moments to
    complete.
    TIP: The containers can be brought up the first time with the (-d) flag ommited so the
    logs can be viewed.

```
docker-compose -f /opt/github/sfcollector/docker-compose.yml up -d
```

Grafana Configuration

1. When the compose process finishes, launch a web browser to http://<VM IP Addr> The
    Grafana web interface should appear. Log in with the admin account and password
    configured in the docker-compose.yml file
2. From the home dashboard, add the Graphite database as a datastore for Grafana

    A. Click the **Add data source** button
    B. In the Name field, create a name for the data source
    C. In the Type field, select Graphite
    D. In the URL field, enter "http://<VM IP>:8080"
    E. In the Access field. select proxy
    F. Check the Basic Auth checkbox
    G. In the User field, enter 'graphite'
    H. In the Password field, enter the configured password from the basic_auth file

Edit data source

| Config | Dashboards |

| Name | graphitedb | ⓘ | Default | ☑ |

| Type | Graphite ▾ |

Http settings

| Url | http://172.27.96.14:8080 | ⓘ |
| Access | proxy ▾ | ⓘ |

Http Auth

| Basic Auth | ☑ | With Credentials | ⓘ | ☐ |
| TLS Client Auth | ☐ | With CA Cert | ⓘ | ☐ |

Basic Auth Details

| User | graphite |
| Password | ········· |

Success
Data source is working

[Save & Test] [Delete] Cancel

3. Import the included Grafana example dashboards. There are a number of pre-built dashboards available on grafana.com for the sfcollector.

| Dashboard Name	ID	Link
NetApp HCI - vSphere VM Stats	4541	https://grafana.com/dashboards/4541
NetApp HCI - vSphere ESXi Stats	4544	https://grafana.com/dashboards/4544
NetApp HCI - SolidFire Cluster	4547	https://grafana.com/dashboards/4547
NetApp HCI - SolidFire Node	4550	https://grafana.com/dashboards/4550
NetApp HCI - SolidFire Volume	4553	https://grafana.com/dashboards/4553
NetApp HCI - SolidFire Volume TopN	4559	https://grafana.com/dashboards/4559

A. To import these preconfigured dashboards, log into Grafana
B. From the Home screen, select Dashboards -> Import
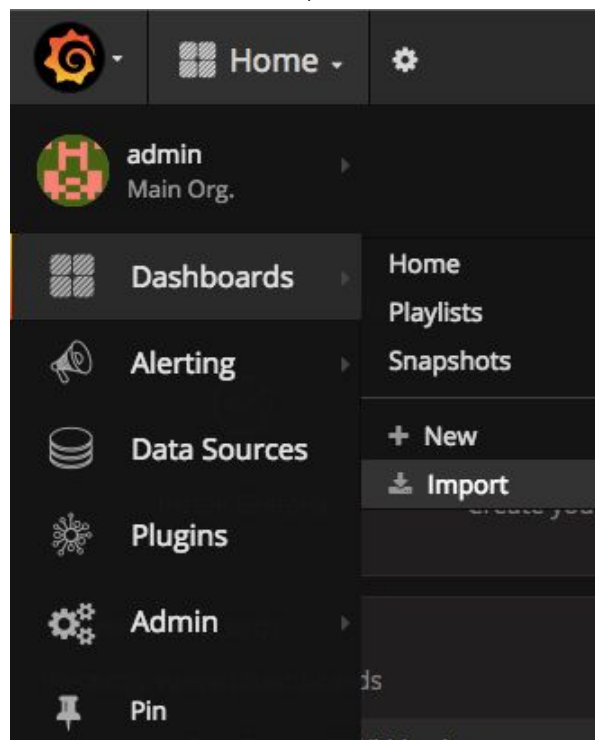
C. From the table above, specify the ID or URL of the dashboard to import in the "Grafana.com Dashboard" box
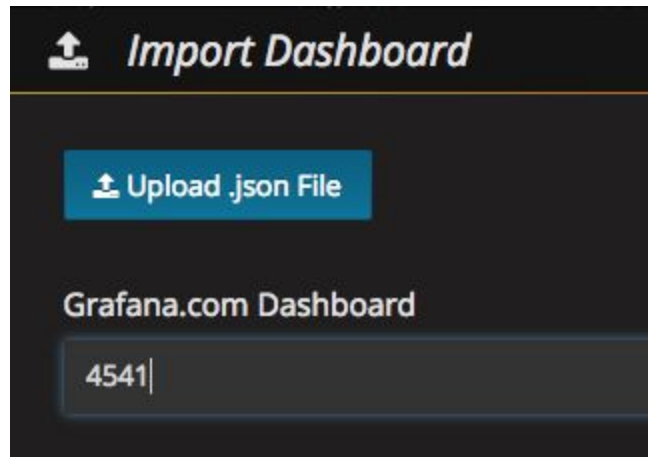
D. Hit tab and the dashboard should load. Select the datasource and click the Import button to finish importing the dashboard. Repeat for the other included example dashboards
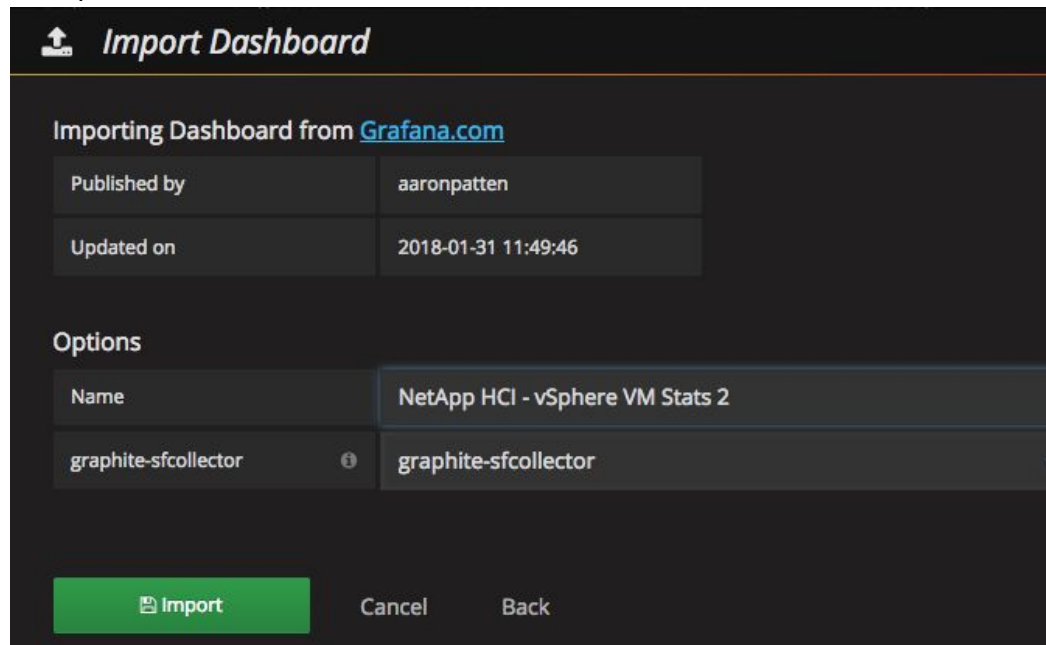
Graph Conventions Used

The following conventions are used in the system graphs.

- Null values are shown as null for most graphs. This allows spotting objects that fail to report stats. It is augmented by keepLasValue.
- keepLastValue(5) - Continues the line with the last received value when gaps ('None' values) appear in your data, rather than breaking your line. If there are more than (5) consecutive missed reporting periods, a break will show in the graph. Removing this option will show a graph with breaks for any object that has no stats for the reporting period. Five minutes was chosen as that is the break point for evicting a node from the cluster. Note the difference in the following screen shots.
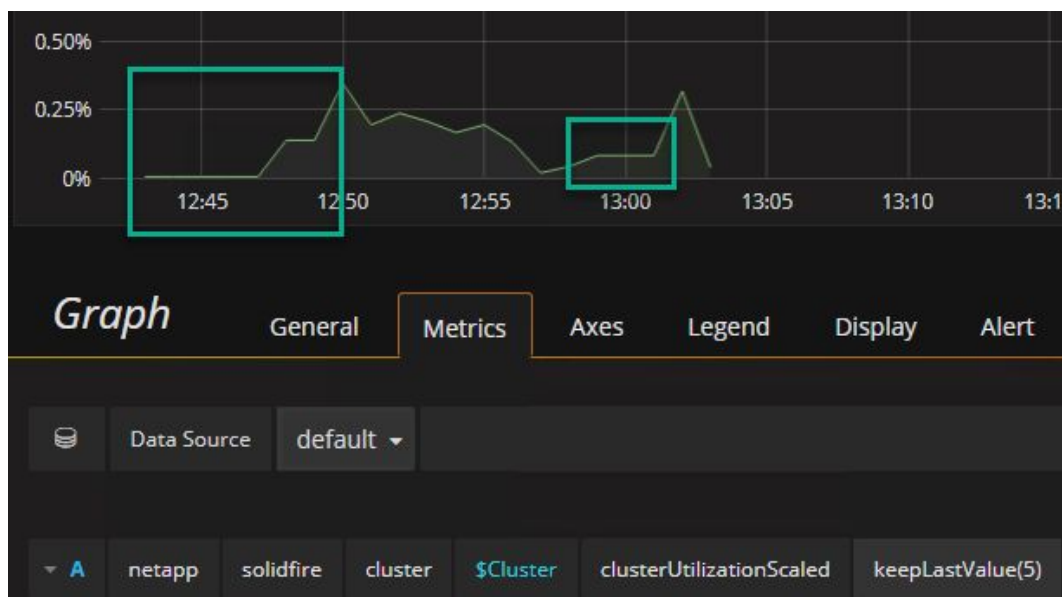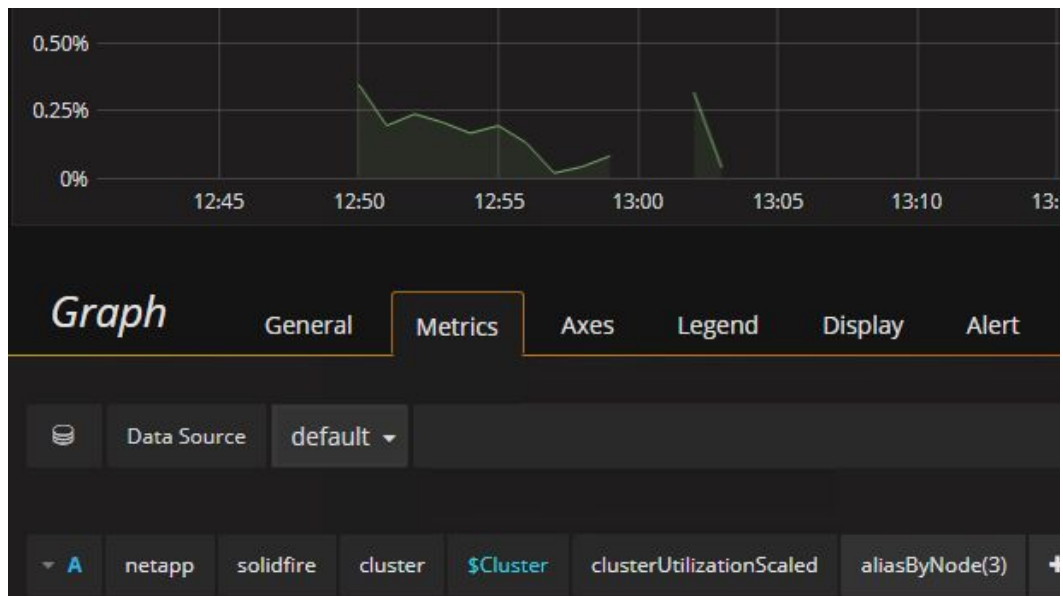
Troubleshooting

This blog has some exellent troubleshooting steps for a graphite+Grafana configuration and well worth the time to read through.

http://dieter.plaetinck.be/post/25-graphite-grafana-statsd-gotchas/

Graphing a datapoint from the Graphite database to validate metrics

The Graphite API used in this project does not include the graphical front end for Graphite so the render API for Graphite can be used to verify that metrics are being pushed into the graphite database.
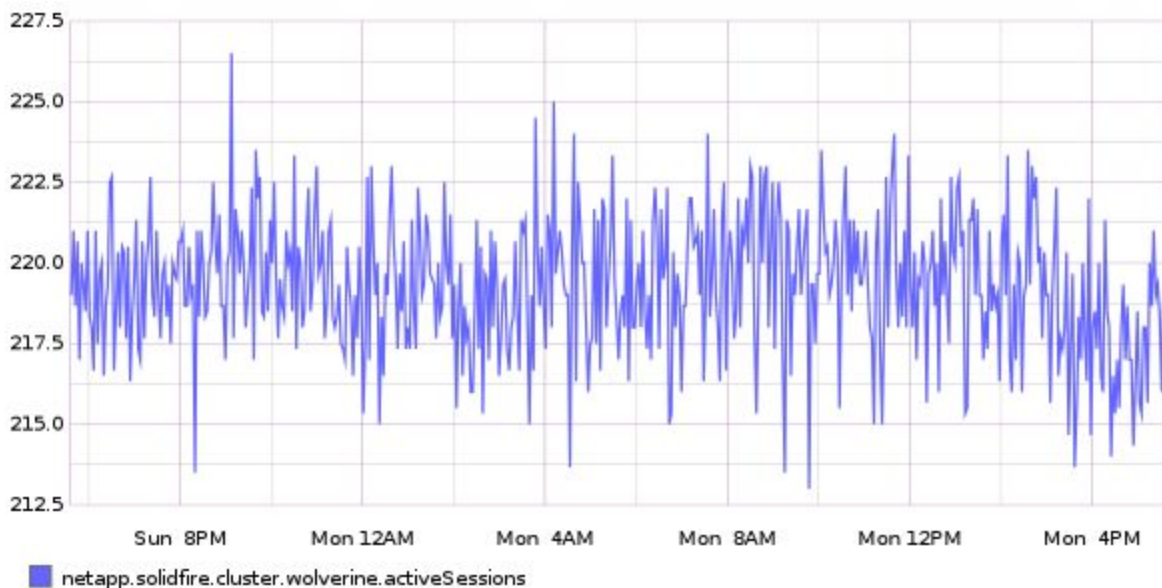
Validating SolidFire metrics are being pushed to Graphite

The format for displaying cluster metrics is:

```
http://<docker VM IP>:8080/render?target=netapp.solidfire.cluster.<cluster
name>.<metric>
```

For example, to see cluster activeSessions:

http://172.27.96.14:8080/render?target=netapp.solidfire.cluster.wolverine.activeSessions

netapp.solidfire.cluster.wolverine.activeSessions

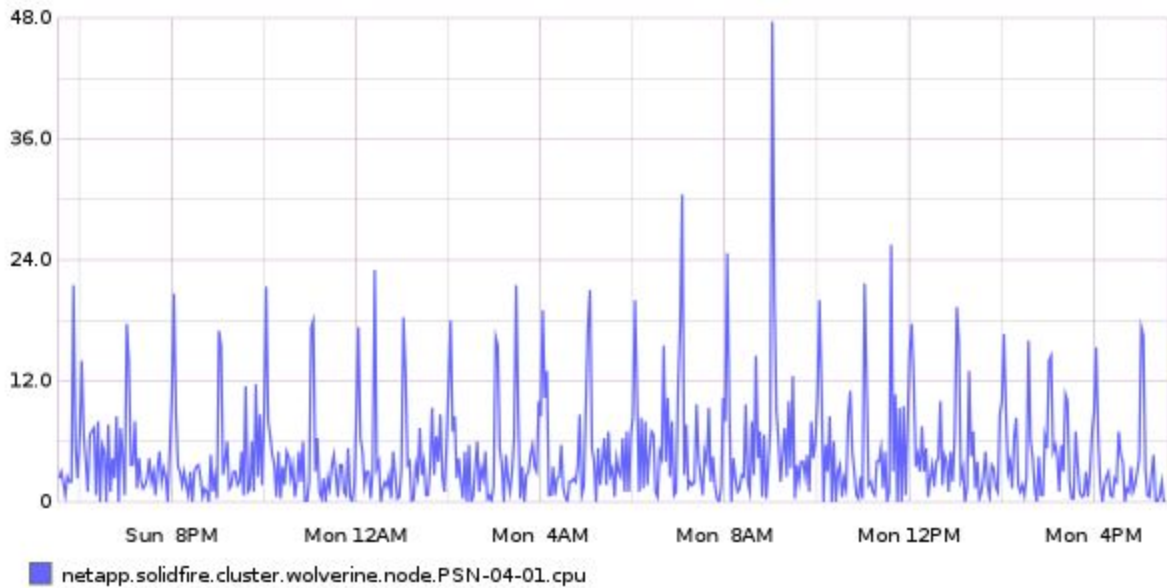To display node metrics:
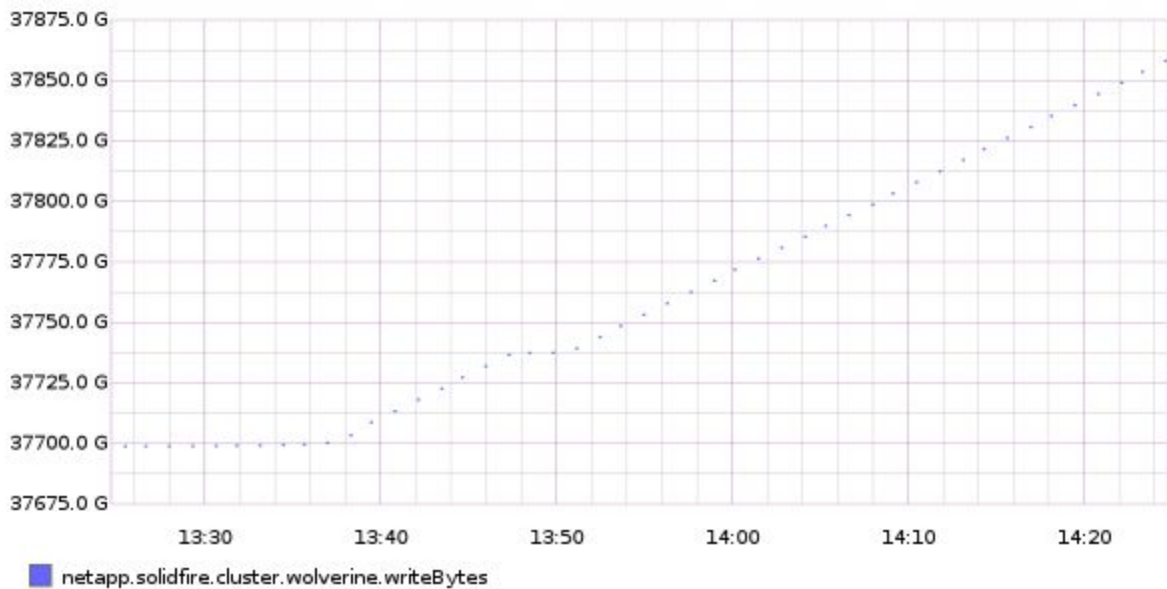
```
http://<docker VM IP>:8080/render?target=netapp.solidfire.cluster.<cluster
name>.node.<node name>.<metric>
```

For example, to see a cluster node CPU usage:

http://172.27.96.14:8080/render?target=netapp.solidfire.cluster.wolverine.node.PSN-04-01.cpu

netapp.solidfire.cluster.wolverine.node.PSN-04-01.cpu

To display only the metrics from the last hour, add the $from=-<time window> argument
http://172.27.96.26:8080/render?target=netapp.solidfire.cluster.wolverine.writeBytes&from=-1hour

netapp.solidfire.cluster.wolverine.writeBytes

Checking sfcollector logs

If it becomes necessary to connect to the sfcollector to troubleshoot, the ENTRYPOINT for the container will need to be overriden

```
docker run --entrypoint "/bin/bash" -it sfcollector_sfcollector
```

Logs for the collector are stored in the /tmp directory.

Rebuilding a Container

If you need to make a change to a single container in the docker-compose setup (for instance to change the collector wrapper script) that can be done without taking down all the containers.

```
#List the services
root@hci-grafana01:/opt/github/sfcollector# docker-compose config
--services
graphite
vsphere-collector
sfcollector-alpine
grafana

#Stop the service
docker-compose stop sfcollector-alpine #this is the service name
<make changes>

#Start the service
docker-compose up -d --no-deps --build sfcollector-alpine
```

This will take down only the container that needs to be edited, you can then make the changes, rebuild the container and run it.

Removing stale metrics from the Whisper database

If there are stale metrics in the Whisper database the corresponding metric files have to be removed from the graphite container persistent storage. If using storage allocated via the NDVP we must start the graphite container with the persistent volume attached and then make the required changes. For example, to remove all the metrics for the "ultron" cluster from graphite, do the following.

```
#Stop the sfcollector-alpine container
root@vmgrafana01-0:/opt/github/sfcollector/collector#
docker-compose down
Stopping sfcollector_sfcollector_1 ... done
Stopping sfcollector_grafana_1 ... done
Stopping sfcollector_graphite_1 ... done
Removing sfcollector_sfcollector_1 ... done
Removing sfcollector_grafana_1 ... done
Removing sfcollector_graphite_1 ... done
```

```
#Start the graphite container with persistent storage
docker run --rm -it --entrypoint "/bin/bash" --volume
graphite-db:/opt/graphite/storage/whisper sfcollector_graphite

#Remove old stats for the ultron cluster
root@ed7dbf28f424:/# ls
/opt/graphite/storage/whisper/netapp/solidfire/cluster/
ultron  wolverine

root@ed7dbf28f424:/# rm -rf
/opt/graphite/storage/whisper/netapp/solidfire/cluster/ultron
```

At this point, the collector can be restarted.

Cleaning Up
To remove all untagged (dangling) images
```
docker image prune
```

Appendix A - Source Material

#Install Docker
https://docs.docker.com/engine/installation/linux/ubuntu

#NDVP Quick Start
http://netappdvp.readthedocs.io/en/latest/quick_start.html

#Composing Graphite server w/Docker
https://thepracticalsysadmin.com/composing-a-graphite-server-with-docker/

#SolidFire Collector for Graphite
https://github.com/cbiebers/solidfire-graphite-collector

#vSphere Graphite collector
https://github.com/cblomart/vsphere-graphite

Appendix B - Stats Definitions

This is a (potentially non-exhaustive) list of statistics used by sfcollector.

API	Stat Name	Description	Calc	Type	Ver
clusterStats	actualIOPS	Current actual IOPS for the entire cluster in the last 500 milliseconds	Point in Time	Integer	9,10
	clientQueueDepth	The number of outstanding read and write operations to the cluster	NA	Integer	9.10
	clusterUtilization	Cluster capacitiy being utilized	NA	Float	9,10
	latencyUSec	The average time, in microseconds to complete operations to a cluster in the last 500 milliseconds	Point in Time	Integer	9,10
	normalizedIOPS	Average number of IOPS for the entire cluster in the last 500 milliseconds	Point in Time	Integer	10
	readBytes	The total cumulative bytes read from the cluster since the creation of the cluster	Monotonic	Integer	9,10
	readBytesLastSample	The total number of bytes read from the cluster during the last sample period	Point in Time	Integer	9,10
	readLatencyUSec	The average time, in microseconds, to complete read operations to the cluster in the last 500 milliseconds.	Point in Time	Integer	9,10
	readOps	The total cumulative read operations to the cluster since the creation of the cluster	Monotonic	Integer	9,10
	readOpsLastSample	The total number of read operations during the last sample period.	Point in Time	Integer	9,10
	unalignedReads	The total cumulative unaligned read operations to a cluster since the creation of the cluster.	Monotonic	Integer	9,10
	unalignedWrites	The total cumulative unaligned write operations to a cluster since the creation of the cluster.	Monotonic	Integer	9,10
	writeLatencyUSec	The average time, in microseconds, to complete write operations to the cluster in the last 500 milliseconds.	Point in Time	Integer	9,10
	writeOps	The total cumulative write operations to the cluster since the creation of the cluster	Monotonic	Integer	9,10
	writeOpsLastSample	The total number of write operations during the last sample period.	Point in Time	Integer	9,10

	writeBytes	The total cumulative bytes write from the cluster since the creation of the cluster	Monotonic	Integer	9,10
	writeBytesLastSample	The total number of bytes write from the cluster during the last sample period	Point in Time	Integer	9,10
clusterCapacity	activeBlockSpace	The amount of space on the block drives. This includes additional information such as metadata entries and space which can be cleaned up.	NA	Integer	9.10
	activeSessions	The number of active iSCSI sessions	NA	Integer	9.10
	averageIOPS	The average IOPS for the cluster since midnight UTC	NA	Integer	9,10
	ClusterRecentIOSize	The average size of IOPS to all volumes in the cluster	NA	Integer	9.10
	currentIOPS	The average IOPS for all volumes in the cluster over the last five seconds	NA	Integer	9,10
	maxIOPS	The estimated maximum IOPS capability of the current cluster	NA	Integer	9,10
	maxOverProvisionableSpace	The maximum amount of provisionable space.	NA	Integer	9,10
	maxProvisionedSpace	The total amount of provisionable space if all volumes are filled to 100%	NA	Integer	9,10
	maxUsedMetadataSpace	The number of bytes on volulme drives used to store metadata	NA	Integer	9,10
	maxUsedSpace	The total amount of space on all active block drives	NA	Integer	9,10
	nonZeroBlock	The total number of 4KiB blocks that contain data after the last garbage collection.	NA	Integer	9,10
	peakActiveSessions	The peak number of iSCSI connections since midnight UTC	NA	Integer	9,10
	peakIOPS	The hightest value for currentIOPS since midnight UTC	NA	Integer	9,10
	provisionedSpace	The total amount of space provisioned in all volumes on the cluster	NA	Integer	9,10
	totalOps	The total number of I/O operations performed throughout the lifetime fo the clsuter	NA	Integer	9,10
	uniqueBlocks	The total number of blocks stored on the block drives, including replicated blocks.	NA	Integer	9,10
	uniqueBlocksUsedSpace	The total amount of data the uniqueBlocks take up on the block drives.	NA	Integer	9,10
	usedMetadataSpace	The total number of bytes on volume drives used to store	NA	Integer	9,10

		metadata			
	usedMetadataSpace InSnapshots	The number of bytes on volumes drives used for storing unique data in snapshots. This provides an estimate of how much metadata space would be regained by deleting all snapshots on the system	NA	Integer	9,10
	usedSpace	The total amount of space used by all block drives in the system	NA	Integer	9,10
	zeroBlocks	The total number of empty 4KiB blocks without data after the last round of garbage collection.	NA	Integer	9,10
nodeStat s	cpu	CPU usage in %	NA	Integer	9,10
	cBytesIn	Bytes in on the cluster interface	NA	Integer	9,10
	cBytesOut	Bytes out on the cluster interface	NA	Integer	9,10
	sBytesIn	Bytes in on the storage interface	NA	Integer	9,10
	sBytesOut	Bytes out on the storage interface	NA	Integer	9,10
	mBytesIn	Bytes in on the management interface	NA	Integer	9,10
	mBytesOut	Bytes out on the management interface	NA	Integer	9,10
	networkUtilizationCl uster	Network interface utilization (%) for the cluster network interface	NA	Integer	9,10
	networkUtilizationSt orage	Network interface utilization (%) for the storage network interface	NA	Integer	9,10
	readOps	Monotonically increasing value of the total read operations to a node	NA	Integer	10
	usedMemory	Total usage in bytes	NA	Integer	9,10
	writeOps	Monotonically increasing value of the total write operations to a node	NA	Integer	10
volumeS tats	accountID	The ID of the account of the volume owner	NA	Integer	9,10
	actualIOPS	The current actual IOPS to the volume in the last 500 milliseconds	Point in Time	Integer	9,10
	averageIOPSize	The average size in bytes of the recent IO to the volume in the last 500 miliseconds	Point in Time	Integer	9,10
	burstIOPSCredit	The total number of IOP credits available.	NA	Integer	9,10

	clientQueueDepth	The number of outstanding read and write operations to the volume	NA	Integer	9,10
	latencyUSec	The average time, in microseconds, to complete operations to the volume in the last 500 milliseconds.	Point in Time	Integer	9,10
	readBytes	The total cumulative bytes read from the volume since the creation of the volume	Monoto nic	Integer	9,10
	readBytesLastSamp le	The total number of bytes read from the volume during the last sample period	Point in Time	Integer	9,10
	readLatencyUSec	The average time, in microseconds, to complete read operations to the volume in the last 500 milliseconds.	Point in Time	Integer	9,10
	readOps	The total cumulative read operations to the volume since the creation of the volume	Monoto nic	Integer	9,10
	readOpsLastSample	The total number of read operations during the last sample period.	Point in Time	Integer	9,10
	throttle	A flaoting value between 0 and 1 that represents how much the system is throttling clients below their maxIOPS because of replication of data, transient errors and snapshots taken.	NA	Float	9,10
	unalignedReads	The total cumulative unaligned read operations to a volume since the creation of the volume.	Monoto nic	Integer	9,10
	unalignedWrites	The total cumulative unaligned write operations to a volume since the creation of the volume.	Monoto nic	Integer	9,10
	volumeUtilization	A floating value that describes how much the client is using the volume. 0 = the client is not using the volume; 1 = the client is using their max; >1 = the client is using their burst	NA	Float	9,10
	writeLatencyUSec	The average time, in microseconds, to complete write operations to the volume in the last 500 milliseconds.	Point in Time	Integer	9,10
	writeOps	The total cumulative write operations to the volume since the creation of the volume	Monoto nic	Integer	9,10
	writeOpsLastSampl e	The total number of write operations during the last sample period.	Point in Time	Integer	9,10
	writeBytes	The total cumulative bytes write from the volume since the creation of the volume	Monoto nic	Integer	9,10
	writeBytesLastSamp le	The total number of bytes write from the volume during the last sample period	Point in Time	Integer	9,10

	zeroBlocks	The total number of empty 4KiB blocks without data after the last round of garbage collection.	NA	Integer	9,10