

Monitor ESXi, Synology, Docker, PiHole, Plex and Raspberry Pi and Windows using Grafana, InfluxDB and Telegraf

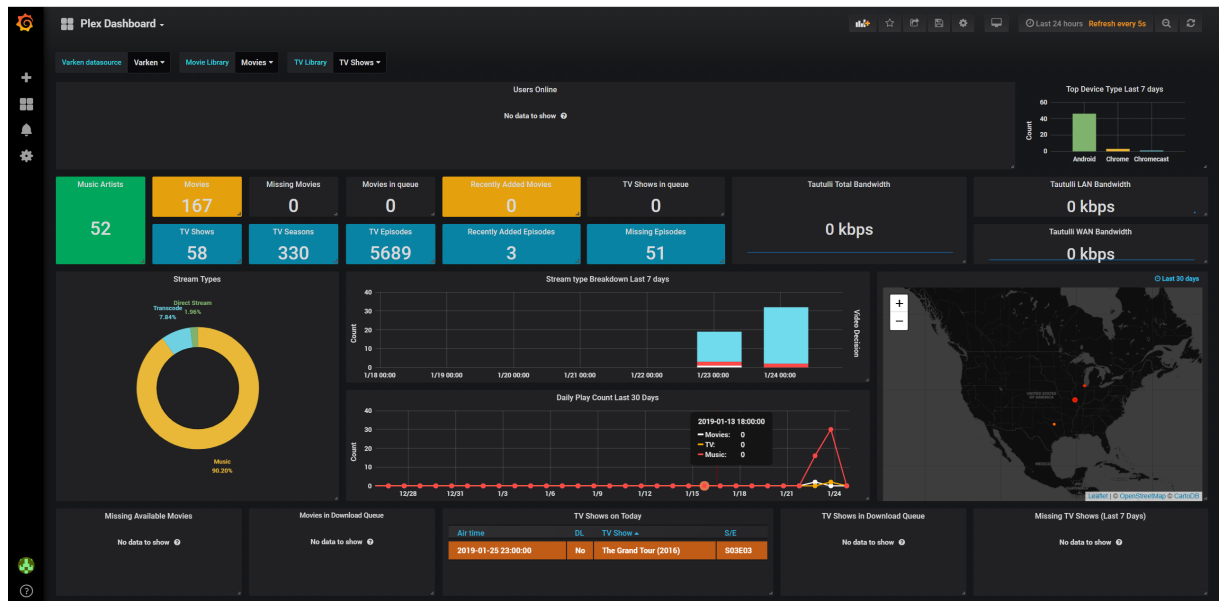
Demo

<https://grafana.challa.co>

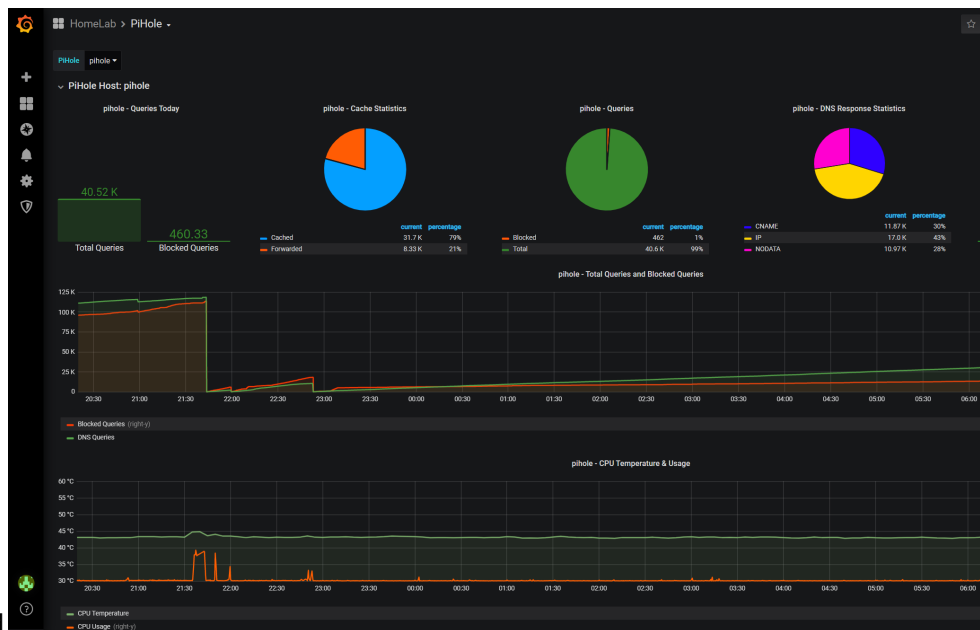
Screenshots:



Synology Dashboard

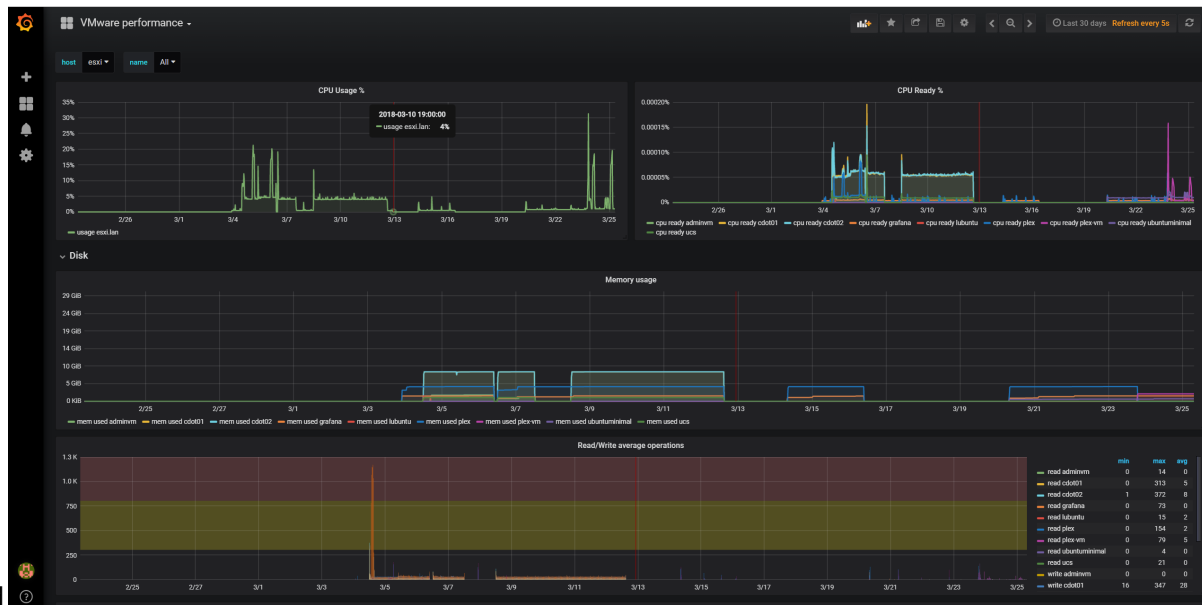


Plex:

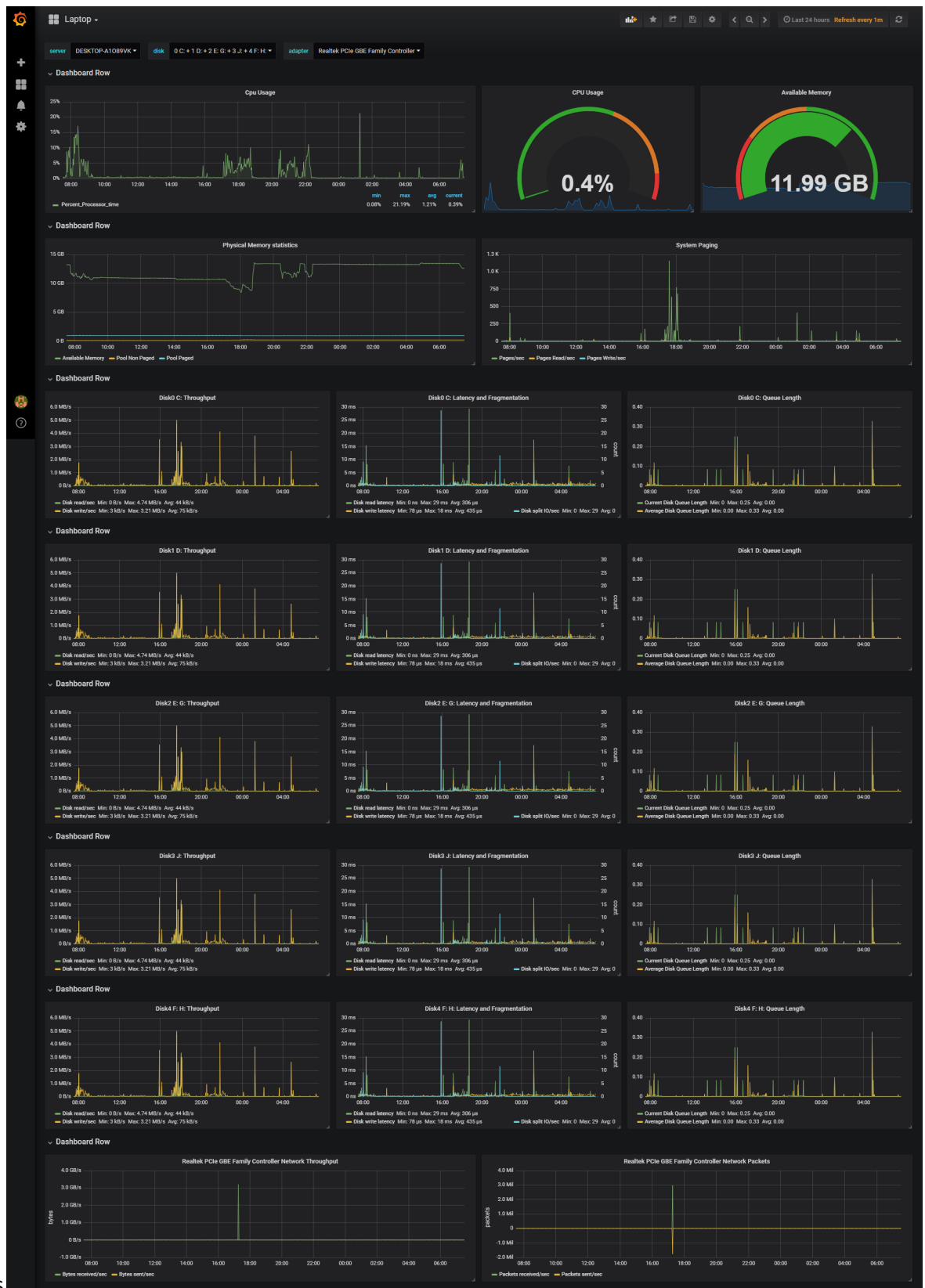


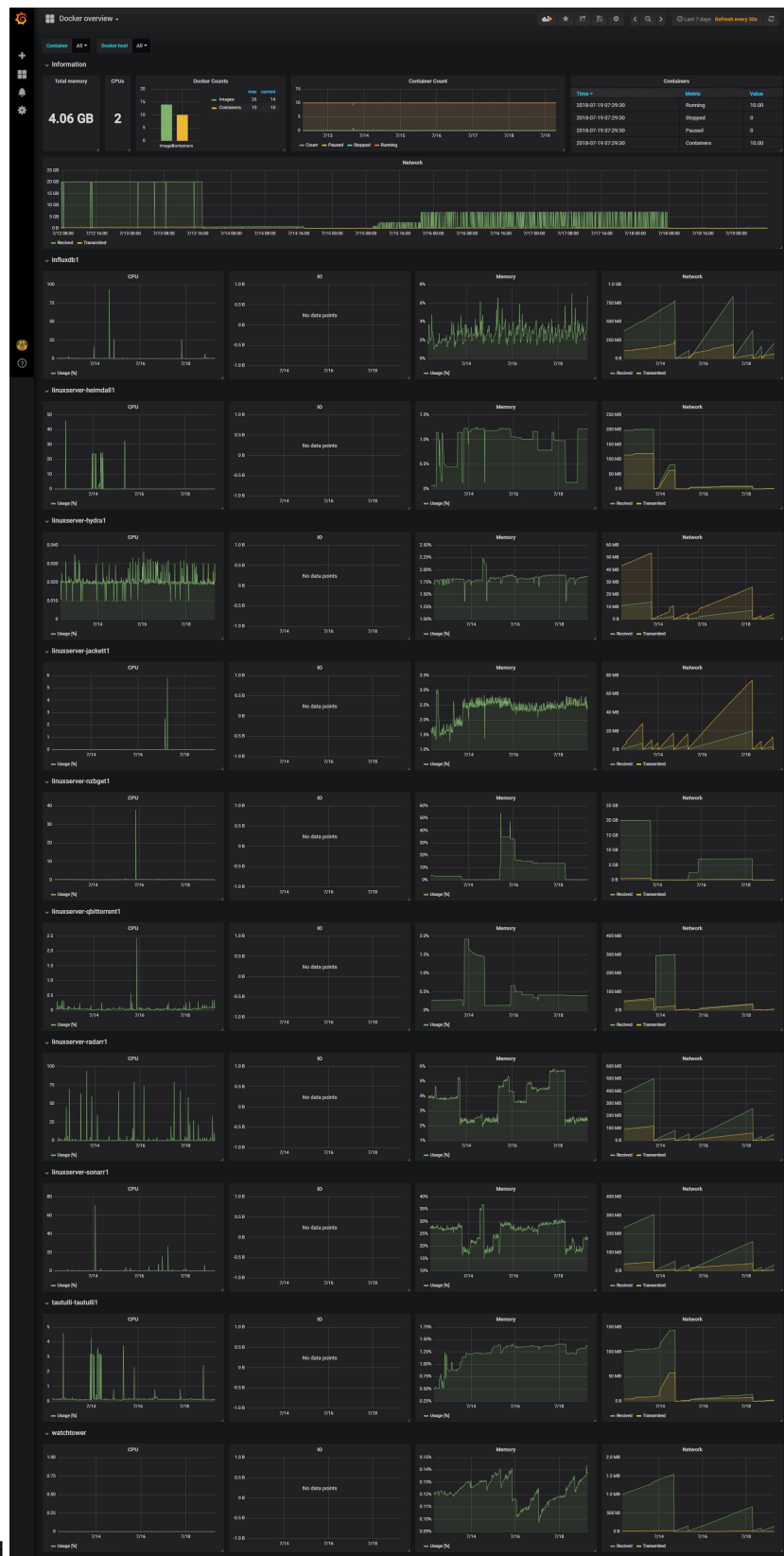
PiHole and Raspberry Pi Dashboard

ESXi Dashboard

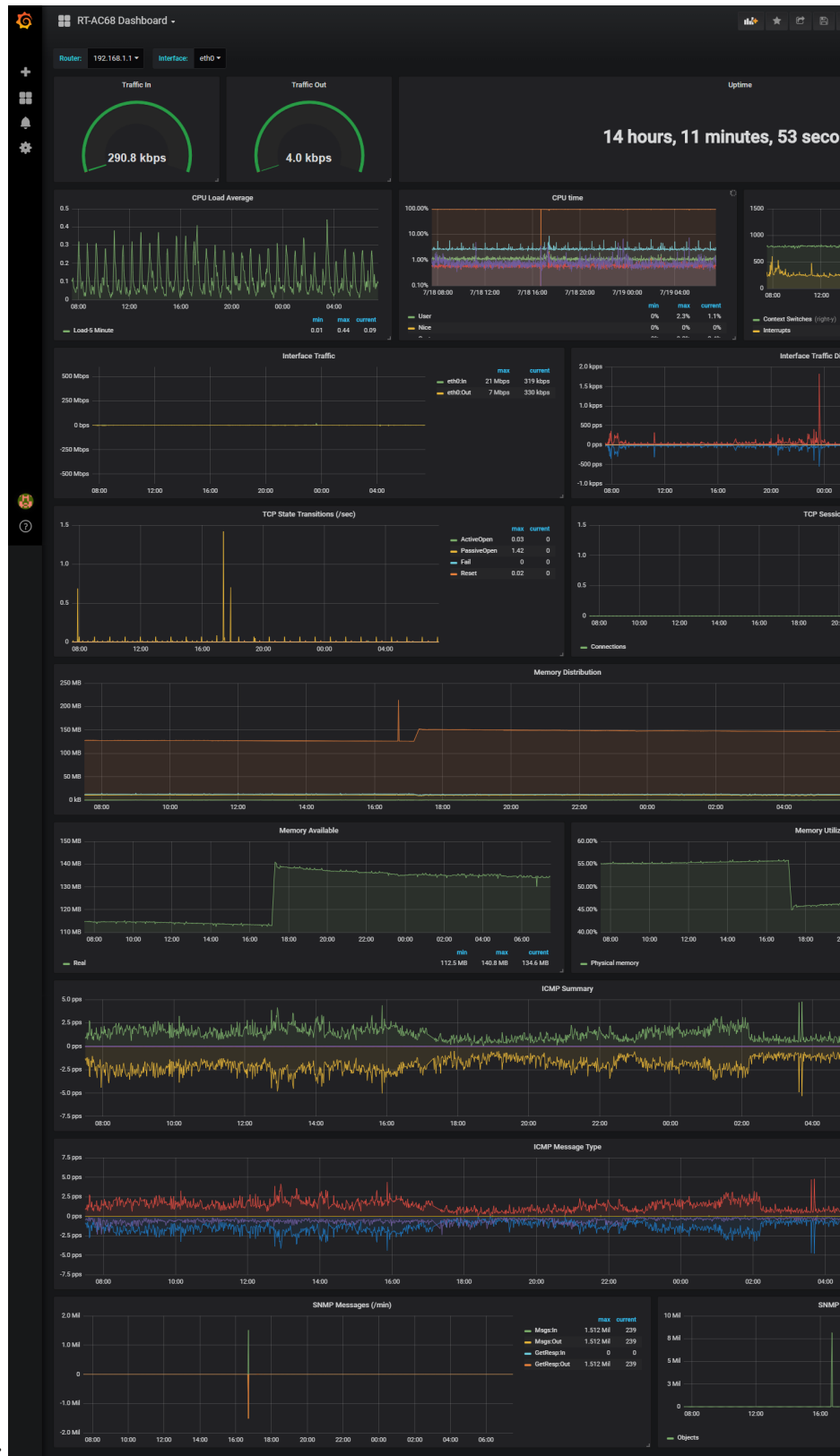


Windows





Docker Dashboard



Asusrt-Merlin Router (ASUS RT-AC68):

Raspberry Pi

Download “telegraf_pi_temp.sh” and ‘chmod +x’ the script. Then call it from within telegraf using “[[inputs.exec]]” (already included telegraf.conf in this repo).

PiHole

Change the address of PiHole to your PiHole’s address in telegraf.conf

4. Restart telegraf

Synology NAS

Based on: https://github.com/jperillo/Synology_dashboard_grafana

Make sure snmp-mibs-downloader is already installed on your telegraf host. It will download and install additional MIBs during install.

```
1 apt-get install snmp-mibs-downloader
```

then

```
1 download-mibs
```

1. edit /etc/snmp/snmp.conf and comment out the ‘mibs:’ line. Here is what mine looks like:

```
1 GNU nano 2.7.4
2 File: /etc/snmp/snmp.conf
3 # As the snmp packages come without MIB files due to license
  reasons, loading
4 # of MIBs is disabled by default. If you added the MIBs you can
  reenable
5 # loading them by commenting out the following line.
6 # mibs :
```

2. Now when you do a SNMPwalk, it will automatically translate OIDs to Names. This is what it looks like against my Synology:

```
1 kumar@raspberrypi:~ $ snmpwalk -c public -v 2c 192.168.1.5
2 SNMPv2-MIB::sysDescr.0 = STRING: Linux DiskStation 3.10.102 SMP
  Fri Jan 26 06:46:44 CST 2018 x86_64
3 SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10
4 DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (90908998) 10
  days, 12:31:29.98
5 SNMPv2-MIB::sysContact.0 = STRING: Redacted
6 SNMPv2-MIB::sysName.0 = STRING: Synology DS416Play
```

```
7 SNMPv2-MIB::sysLocation.0 = STRING: Home
```

3. Get your MIBs (from here, for Synology Synology MIB download) and drop them in either of the below locations:

`/home/$USER/.snmp/mibs` or `user/share/snmp/mibs`

4. Go through the linked page above or the snmpwalk output and make a list of OIDs you want to monitor (grep is your friend here)
5. Add them to telegraf.conf using examples others have provided elsewhere in this thread
6. Restart telegraf and test with the '-test' flag. To verify everything is working as expected.

VMWare

Telegraf introduced a new vsphere plugin. I will be using this instead of a custom script going forward. This plugin and details can be found here:

<https://github.com/influxdata/telegraf/tree/master/plugins/inputs/vsphere>

Dashboards for the metrics can be found here:

<https://github.com/jorgedlcruz/vmware-grafana>

Docker

DockerHost: Synology NAS (DS416Play) Telegraf: Raspberry Pi

Since I did not want to mess around with exposing docker.sock file to a remote client, I went with exposing a TCP endpoint on docker host to a remote telegraf agent.

To do this:

1. On Docker Host (Synology):

Add the endpoint details to /var/packages/Docker/etc/dockerd.json like so:

```
1 admin@DiskStation:~$ cat /var/packages/Docker/etc/dockerd.json
2 {
3     "hosts" : [ "tcp://synology.lan:2375", "unix:///var/run/docker.sock" ],
4     "registry-mirrors" : []
5 }
```

In the above snippet, `tcp://synology.lan:2375` is the end point definition we have to add

Note: Don't change any part of the `"unix:///var/run/docker.sock"` definition. Synology uses to run the Docker GUI. Also, since this is a JSON file, all lines except the last line have a `" , "` at the end. Also note the `" , "` after the TCP definition.

If you want to be doubly sure, you can use <https://jsonlint.com> to validate the JSON contents.

Once this is done, restart the docker package from within DSM's Package center

2. On Telegraf: Add the below lines to your input plugins:

```
1 # Synology Docker
2 [[inputs.docker]]
3   endpoint = "tcp://synology.lan:2375"
4   container_names = []
```

3. Grafana Dashboard:

Grafana Dashboard JSON is included in this repository. Simply import it, define your data source and you should be good to go.

Plex

Plex can be monitored using Verken linked below. I have included my customized dashboard in the repo for reference.

<https://github.com/Boerderij/Varken>