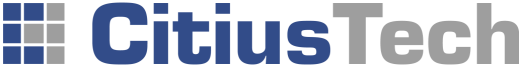


Deployment & Configuration Guide

Consolidated Mirth Dashboard

Version 1.0



Copyright

This document is Client Confidential and contains proprietary information, including trade secrets of CitiusTech. Neither the document nor any of the information contained in it may be reproduced or disclosed to any unauthorized person under any circumstances without the express written permission of Client.

Revision History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Document Version # | Revision Date | Prepared By | Approved By | Approval Date | Summary of Changes |
| 1.0 | 22-07-2020 | Suchetana Shetty | Shanawaz Khan | Click here to enter a date. | First Version |
| 1.1 | Click here to enter a date. |  |  | Click here to enter a date. |  |
| 1.2 | Click here to enter a date. |  |  | Click here to enter a date. |  |

Contents

[1 Introduction 5](#_Toc34406597)

[2 Deployment Pre-Requisites 6](#_Toc34406598)

[2.1 Software Requirements 6](#_Toc34406599)

[2.2 Hardware Requirements 6](#_Toc34406600)

[2.3 Networking Requirements 6](#_Toc34406601)

[2.4 Security Requirements 6](#_Toc34406602)

[2.5 Dependencies 6](#_Toc34406603)

[2.6 Pre-Deployment Configuration 7](#_Toc34406604)

[3 Deployment Procedure 8](#_Toc34406605)

[3.1 Scope 8](#_Toc34406606)

[3.2 Environment 8](#_Toc34406607)

[3.3 Backup Procedure 8](#_Toc34406608)

[3.4 Deployment Procedure 8](#_Toc34406609)

[3.5 Incremental Upgrade Procedure 8](#_Toc34406610)

[4 Post Deployment 9](#_Toc34406611)

[4.1 Configuration 9](#_Toc34406612)

[4.2 Validation 9](#_Toc34406613)

[5 Troubleshooting & Support 10](#_Toc34406614)

[5.1 Problem 1 10](#_Toc34406615)

[5.2 Problem 2 10](#_Toc34406616)

[5.3 Known Issues 10](#_Toc34406617)

[5.4 24x7 Support 10](#_Toc34406618)

[6 Roll Back Procedure 11](#_Toc34406619)

[7 Uninstallation Procedure 12](#_Toc34406620)

# Introduction

There are multiple Mirth Instances implemented for a client, to monitor multiple instances monitoring team should go and check separate dashboards of each mirth instances.

As part of this IAG, a mirth consolidated dashboard has been created where in all the mirth instances’ data scraped into a single dashboard.

This document is intended to monitoring team which helps them deploy the application on to the monitoring server.

# Deployment Pre-Requisites

Mirth Connect should be present

## Software Requirements

Linux OS

Go1.14.2

Grafana 7.0.4

Prometheus 2.18.0

Node Exporter 1.0

Mirth Exporter

## Hardware Requirements

**Minimum Hardware Requirements**

* Processor: 4 cores
* Processor speed: 1 GHz
* Random access memory (RAM): 8 GB
* Hard disk capacity: 500GB

**Recommended Hardware Requirements**

* Processor: 8 cores
* Processor speed: 3 GHz
* Random access memory (RAM): 16 GB

## Networking Requirements

**Default ports:**

1. Grafana = :3000
2. Prometheus = :9090
3. Mirth exporter = :9140
4. Node exporter = :9100

## Security Requirements

Root user access or sudo privileges are required

## Dependencies

Mirth Connect should be there with deployed channels.

## Pre-Deployment Configuration

1. Root user or sudo privileges is required

2. Access to ports 3000, 9090, 9140, 9100, 9140 is required

# Deployment Procedure

The following section provides the detailed steps to successfully deploy each module of the product.

## Scope

Deployment of Go compiler

Grafana Set up

Prometheus Set up

Mirth Exporter Configuration

Node Exporter Configuration

## Environment

Not Applicable

## Backup Procedure

Backup of Grafana database can be taken to ensure that you can always rollback to your previous version. This can be done by taking backup of grafana.db file. This is usually located at /var/lib/grafana/grafana.db on Unix systems. If you are unsure what database you use and where it is stored check you grafana configuration file. If you installed grafana to custom location using a binary tar/zip it is usually in <grafana\_install\_dir>/data.

## Deployment Procedure

**GO set up:**

1. Download the GO tarball using wget command:

wget <https://dl.google.com/go/go1.14.2.linux-amd64.tar.gz>

1. Extract the downloaded tarball using following command inside /usr/local directory. Ensure that you should run below command as a root user

sudo tar -C /usr/local -xzf go1.14.2.linux-amd64.tar.gz

1. Set path environment variable in order to find Go executable binaries by system. For that we need to open /.bash\_profile using below command:

sudo nano ~/.bash\_profile

Now add the below lines to it:

export PATH=$PATH:/usr/local/go/bin:$GOPATH/bin

Save and close the file. After that run below command to reload running profile:

source ~/.bash\_profile

1. Installation of GO can be verified Go version command:

go version

**Grafana set up:**

1. Add Grafana yum repository

Run the commands below as user with sudo privileges or as root user to add repository content.

cat <<EOF | sudo tee /etc/yum.repos.d/grafana.repo

[grafana]

name=grafana

baseurl=https://packages.grafana.com/oss/rpm

repo\_gpgcheck=1

enabled=1

gpgcheck=1

gpgkey=https://packages.grafana.com/gpg.key

sslverify=1

sslcacert=/etc/pki/tls/certs/ca-bundle.crt

EOF

1. Cache index for available packages can be updated by following command:

sudo dnf makecache

1. Install Grafana using below command:

sudo yum install grafana

1. Start Grafana server and enable it to start on boot

sudo systemctl enable --now grafana-server.service

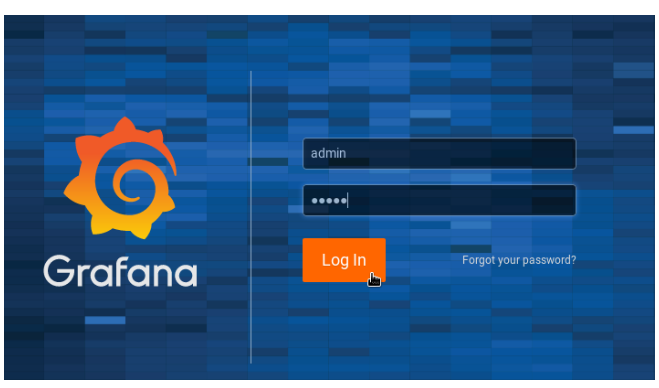
1. To verify status of grafana server following command can be used:

systemctl status grafana-server.service

1. Default port of Grafana is 3000. To allow port 3000 for access to the dashboard execute following command:

sudo firewall-cmd --add-port=3000/tcp --permanent

sudo firewall-cmd –reload

1. To open the Grafana Web UI, navigate to [http://localhost:3000](http://localhost:3000/) with your web browser. 
2. Login page appears. The default login username and password are admin and it allows to change username and password during 1st time login

**Mirth Exporter set up:**

1. Download the file mirth\_exporter.go



1. Save mirth\_exporter.go in in Mirth Connect folder
2. To execute mirth\_exporter.go following command can be used:

go run mirth\_exporter.go &

**Node Exporter set up:**

In order to install Node Exporter login with root user or user with sudo privileges.

1. Using wget command download node\_exporter tarball using following command

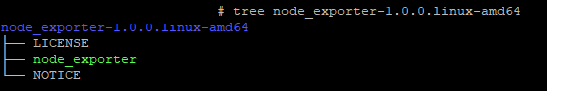
wget <https://github.com/prometheus/node_exporter/releases/download/v1.0.0/node_exporter-1.0.0.linux-amd64.tar.gz>

1. Extract the file using command:

tar -zxpvf node\_exporter-1.0.0.linux-amd64.tar.gz

1. Contents of the extracted folder can be checked using command:

tree node\_exporter-1.0.0.linux-amd64.tar.gz



1. Copy the binary file **node\_exporter** to /usr/local/bin path using command:

cp node\_exporter-1.0.0.linux-amd64/node\_exporter /usr/local/bin

1. Set the file permissions of the node\_exporter file using command:

chown root /usr/local/bin/node\_exporter

1. To configure node\_exporter to run as a service a systemd service file to be created using command:

vi /etc/systemd/system/node\_exporter.service

then paste below code and save the file:

[Unit]

Description=Prometheus Node Exporter

Wants=network-online.target

After=network-online.target

[Service]

User=node\_exporter

Group=node\_exporter

Type=simple

ExecStart=/usr/local/bin/node\_exporter

[Install]

WantedBy=multi-user.target

1. reload systemd manager with the following command:

systemctl daemon-reload

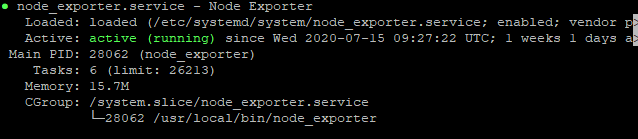
1. To start and enable node\_exporter service use following commands:

systemctl start node\_exporter

systemctl enable node\_exporter

1. To check the status of service, execute following command:

systemctl status node\_exporter



1. By default, node\_exporter uses port 9100. To check the same use following command:

netstat -tunlp



1. To open port 9100 in firewall, use following commands:

firewall-cmd --add-port=9100/tcp --permanent

firewall-cmd –reload

**Prometheus set up:**

In order to install Prometheus, login with root user or user with sudo privileges.

1. Create configuration directories for Prometheus using following command:

mkdir /etc/prometheus

mkdir /var/lib/Prometheus

1. Set the ownership on /var/lib/Prometheus using command:

chown root /var/lib/prometheus/

1. Prometheus tar file can be downloaded using command:

dnf install wget -y

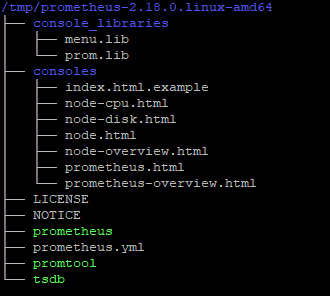
wget https://github.com/prometheus/prometheus/releases/download/v2.18.0/prometheus-2.18.0.linux-amd64.tar.gz -P /tmp

1. Once the download completes, extract the tarball file using command:

tar -zxpvf prometheus-2.18.0.linux-amd64.tar.gz

1. Directory structure can be seen using tree command as following:

tree /tmp/prometheus-2.18.0.linux-amd64



1. The extracted directory contains 2 binary files prometheus & promtool and copy those files to the /usr/local/bin path using commands:

cd /tmp/prometheus-2.14.0.linux-amd64

cp prometheus /usr/local/bin

cp promtool /usr/local/bin

1. To set configurations for Prometheus create a prometheus.yml file:

vi /etc/prometheus/prometheus.yml

1. Paste below configurations into the .yml file and save it

# Global config

global:

scrape\_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.

evaluation\_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.

scrape\_timeout: 15s # scrape\_timeout is set to the global default (10s).

# A scrape configuration containing exactly one endpoint to scrape:# Here it's Prometheus itself.

scrape\_configs:

# The job name is added as a label `job=<job\_name>` to any timeseries scraped from this config.

- job\_name: 'prometheus'

# metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

- targets: ['localhost:9090']

- job\_name: 'mirth'

# metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

- targets: ['localhost:9140']

- job\_name: 'node exporter'

# metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

- targets: ['localhost:9100']

1. To open port to external connections type following commands:

firewall-cmd --add-port=9090/tcp --permanent

firewall-cmd –reload

10. To configure prometheus to run as a service a systemd service file to be created using command:

vi /etc/systemd/system/prometheus.service

then paste below code and save the file:

[Unit]

Description=Prometheus Time Series Collection and Processing Server

Wants=network-online.target

After=network-online.target

[Service]

User=prometheus

Group=prometheus

Type=simple

ExecStart=/usr/local/bin/prometheus \

--config.file /etc/prometheus/prometheus.yml \

--storage.tsdb.path /var/lib/prometheus/ \

--web.console.templates=/etc/prometheus/consoles \

--web.console.libraries=/etc/prometheus/console\_libraries

[Install]

WantedBy=multi-user.target

1. reload systemd manager with the following command:

systemctl daemon-reload

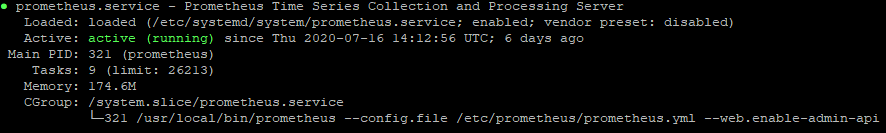
1. To start and enable prometheus service use following commands:

systemctl start prometheus

systemctl enable prometheus

1. To check the status of service, execute following command:

systemctl status Prometheus



1. By default, Prometheus uses port 9090. Check the same using command

netstat -tunlp



## Incremental Upgrade Procedure

**Prometheus Upgradation:**

1. Stop all the running services using command:

systemctl stop prometheus.service

1. Download the Prometheus archive and checksum using command:

wget https://github.com/prometheus/prometheus/releases/download/v2.19.2/prometheus-2.19.2.linux-amd64.tar.gz

1. Now, unpack the archive:

tar -xvf prometheus-2.19.2.linux-amd64.tar.gz

cd prometheus-2.19.2.linux-amd64/

1. Copy the prometheus and promtool executables to the /usr/local/bin directory:

cp prometheus-2.19.2.linux-amd64/{prometheus,promtool} /usr/local/bin/

chown root /usr/local/bin/{prometheus,promtool}

1. Restart all the services :

systemctl daemon-reload

systemctl start prometheus

systemctl enable Prometheus

**Grafana Upgradation:**

1. Before upgrading backup of Grafana database can be taken. This will ensure that you can always rollback to your previous version. This can be done by taking backup of grafana.db file. This is usually located at /var/lib/grafana/grafana.db on Unix systems. If you are unsure what database you use and where it is stored check you grafana configuration file. If you installed grafana to custom location using a binary tar/zip it is usually in <grafana\_install\_dir>/data.
2. Stop grafana service: systemctl stop grafana-server.service
3. Follow the same installation steps mentioned in above section 3.4 (Grafana Set up) and execute command yum install with new package:

sudo yum update Grafana

1. Restart grafana service

systemctl daemon-reload

systemctl start grafana-server.service

systemctl enable grafana-server.service

# Post Deployment

## Configuration

**Prometheus Configuration:** Configuration of prometheus.yml file to scrape metrics from Mirth Exporter and node exporter:

1. Open prometheus.yml file using command

vi /etc/prometheus/prometheus.yml

2. Add below lines to prometheus.yml file

- job\_name: 'mirth'

static\_configs:

- targets: ['localhost:9140']

- job\_name: 'node exporter'

static\_configs:

- targets: ['localhost:9100']

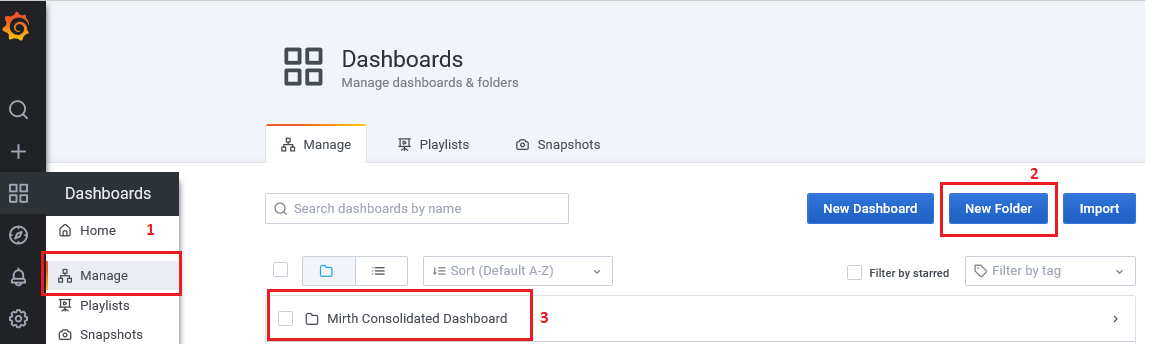
3. Save the file

**Mirth Dashboard Configuration on Grafana:** Configuration of Mirth Dashboard can be done using following steps:

1. Login to Grafana using url <http://localhost:3000> or http://<server ip where Grafana is installed>:3000

2. Default Username and password is admin. Login using same credentials

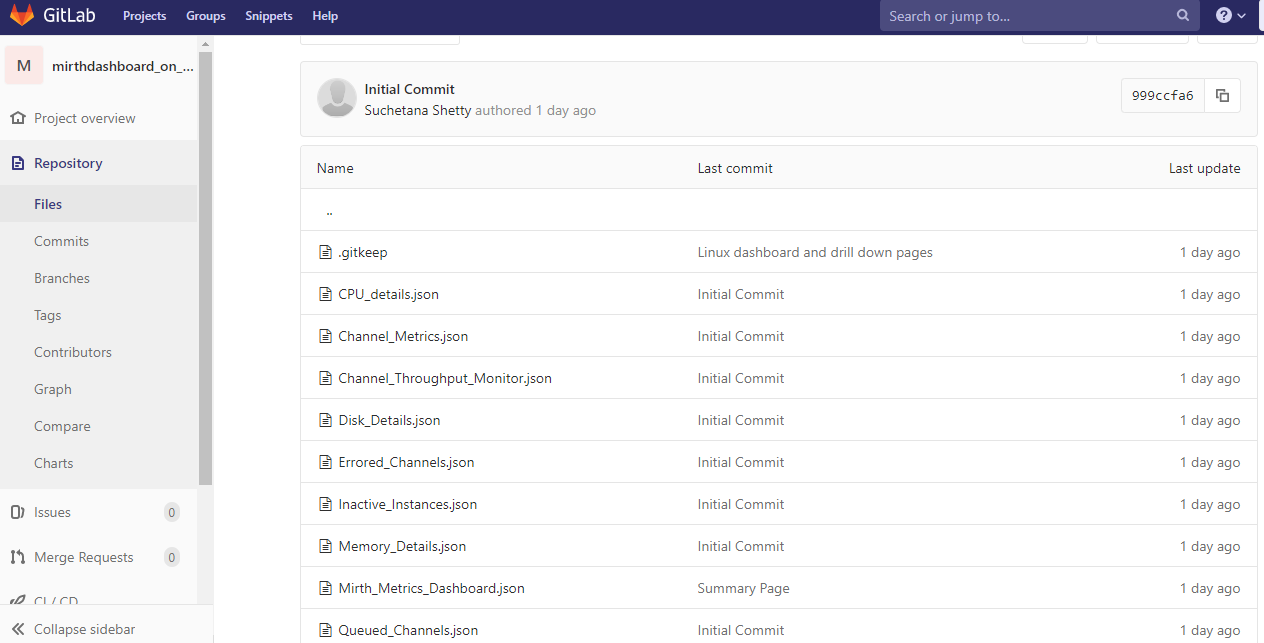
3. Go to **Dashboard** 🡪 **Manage**. Create **New folder** with name ‘Mirth Consolidated Dashboard’



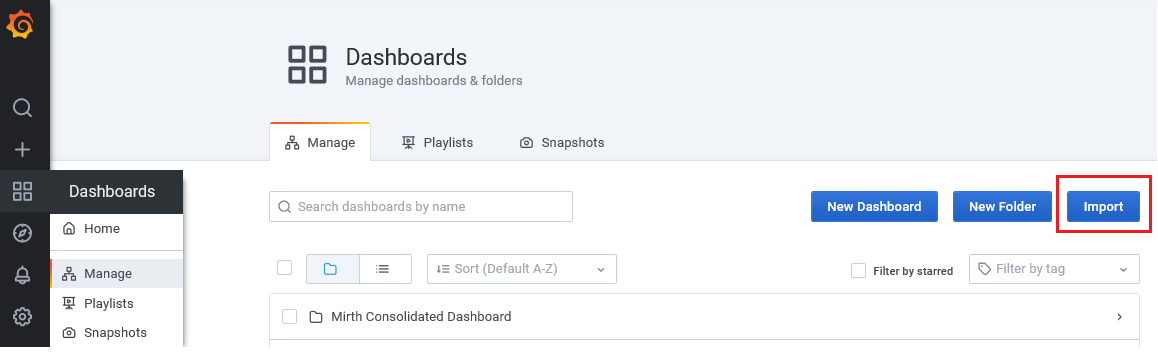
4. Download all json files which are under:

<https://github.com/SalmanCitiustech/MirthDashboard/tree/MirthDashboard_Linux>

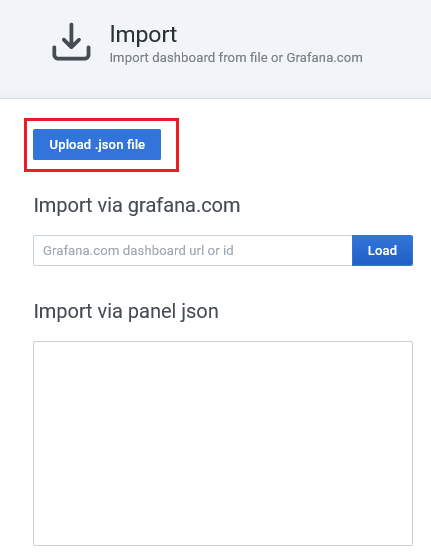
and save the files. There is one summary page ‘Mirth\_Metrics\_Dashboard.json’ and 8 drill down pages present as in below image:



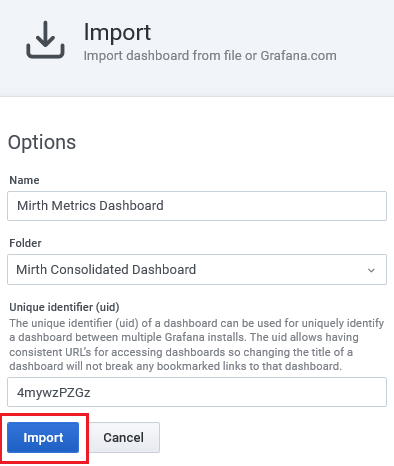
5. On Grafana go to **Dashboard** 🡪 **Manage** and click on **Import**



6. Click on Upload .json file option



6. Browse downloaded .json file mentioned in Step 4 above. Choose folder name as ‘Mirth Consolidated Dashboard’ in below window and click **Import**:

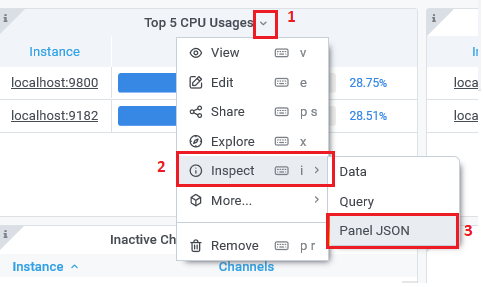


7. Repeat steps from 5 to 6 for all downloaded .json files

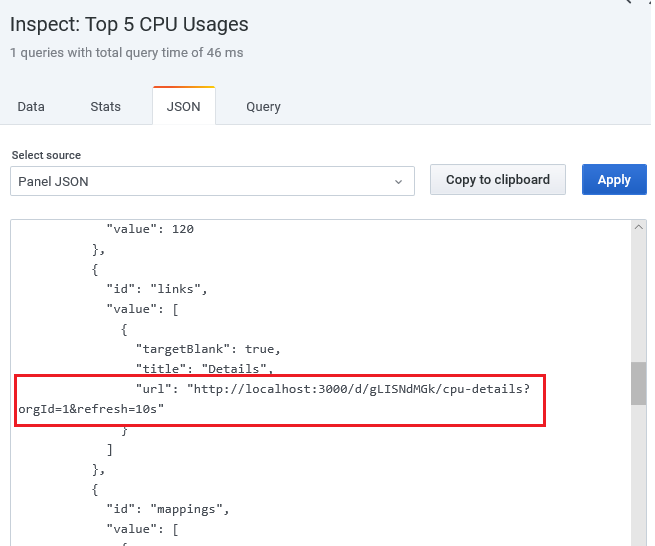
**Updating urls of Grafana Dashboard and drill down pages:**

To update urls present in panels of dashboard follow below steps:

1. Click on arrow which is in the title of the panel and go to Inspect 🡪 Panel Json. This will open json panel



2. Search for url:

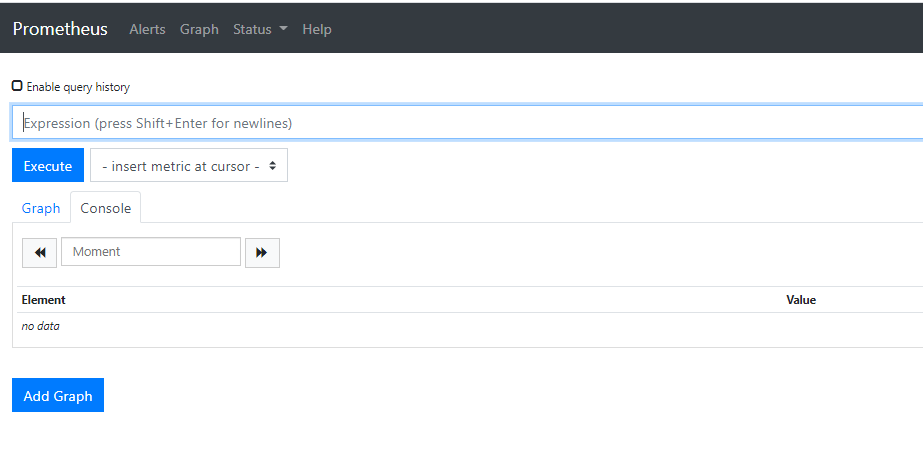


3. Replace <http://localhost:3000> with the server ip or server name where your Grafana is installed and save the file.

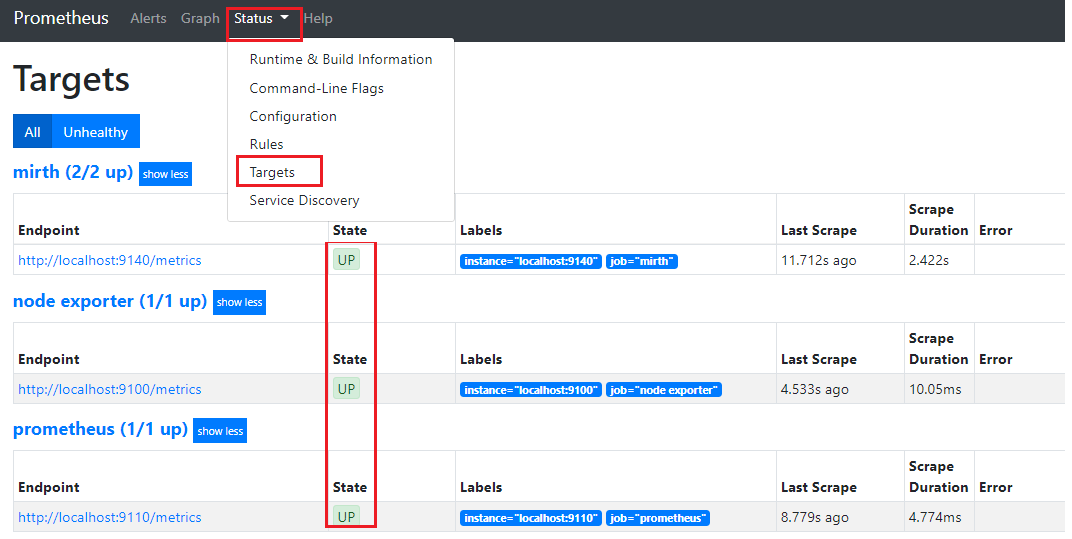
4. Repeat above 3 steps for all the panels of dashboard and drill down pages.

## Validation

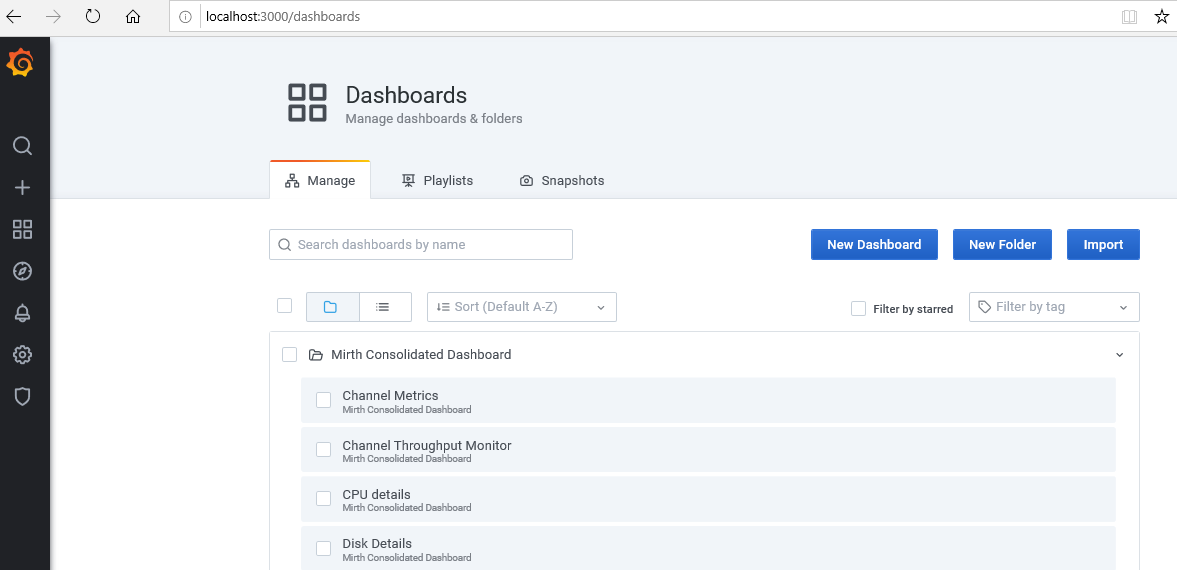
1. Validation of Prometheus : Access the port 9090 of the server where Prometheus is installed and following screen should appear



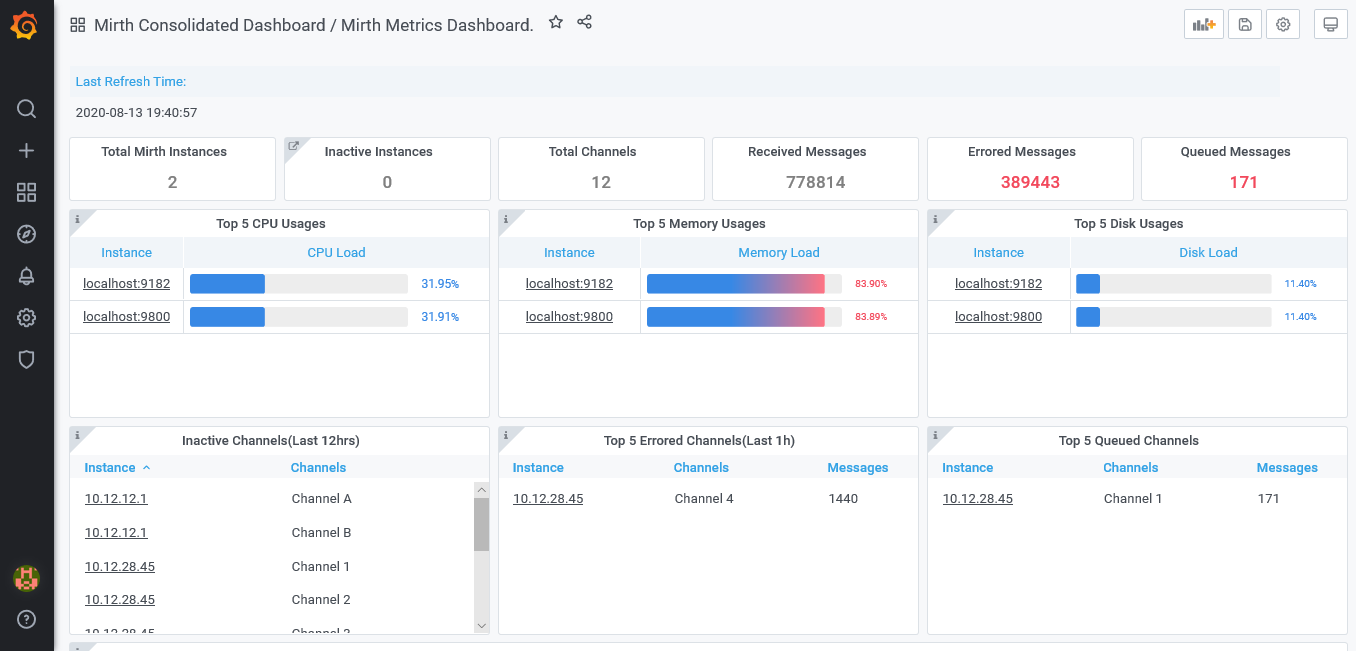
2. Validation of exporters: Check the ‘Targets’ under ‘Status’ tab of Prometheus. All the status of exporters should be ‘UP’



3. Validation of Grafana: Access http://<server ip where Grafana is installed>/dashboards and following screen should appear



4. Select Mirth Metrics Dashboard under Mirth Consolidated Dashboard folder and the following screen should appear:



5. Check for all the drill downs by clicking on Instance from all the panels containing ‘Instance’ column

# Troubleshooting & Support

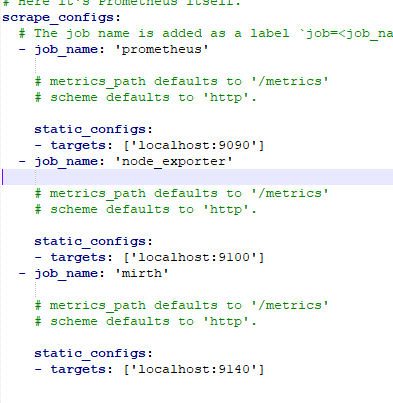
The following section describes list of problems that may appear during or post installation of the system and how to resolve them. Also provide the details

## Problem 1

Situation: Prometheus service is inactive

Consequences: Prometheus is not able to access and not able to scrape any metrics

Action: Add all the required targets following proper spacings and syntaxes. A single miss spacing cause prometheus.yml failed to execute and stops Prometheus service



## Problem 2

Not Applicable

## Known Issues

Not Applicable

## 24x7 Support

Not Applicable

# Roll Back Procedure

The following section describes the set of steps to be performed to roll back the system to an older or pervious version:

1. Stop grafana service: systemctl stop grafana-server.service
2. Follow the same installation steps mentioned in above section 3.4 (Grafana Set up) and execute command yum install with older package:

sudo yum update Grafana

1. Copy backup of Grafana db file mentioned in section 3.5 to /var/lib/grafana/grafana.db
2. Restart grafana service

systemctl daemon-reload

systemctl start grafana-server.service

systemctl enable grafana-server.service

# Uninstallation Procedure

The following section describes the procedure to uninstall the system.

**Uninstallation of GO**

Type the below mentioned commands:

sudo rm -rf /usr/local/go

sudo apt-get uninstall purge golang\*

**Uninstallation of Grafana**

Type the below mentioned commands:

To remove Grafana: sudo apt-get remove grafana

To remove Grafana and its dependencies: sudo apt-get remove --auto-remove Grafana

**Uninstallation of Mirth Exporter**

Remove file mirth\_exporter.go from Mirth Connect folder using command:

rm mirth\_exporter.go

**Uninstallation of Node Exporter**

Type the below mentioned commands:

sudo rm -rf /usr/local/bin/node\_exporter

**Uninstallation of Prometheus**

Type the below mentioned commands:

sudo rm -rf /usr/local/bin/prometheus

sudo rm -rf /usr/local/bin/promtool

sudo rm -rf /etc/prometheus