Observability, Monitoring and Alerting Across Multiple Oracle Private Cloud Appliance X9-2 systems–Part 1

A step-by-step guide to deploying an external Grafana Server service to provide a common monitoring and alerting framework across multiple Oracle Private Cloud Appliance X9-2 systems

September, 2023, Version 1.0.1

Purpose statement

This document outlines the steps necessary for the creation, and use, of an external Grafana Server service to provide a single, central, and common monitoring and alerting framework for multiple Oracle Private Cloud Appliance X9-2 systems.

It is intended solely to help you assess the business benefits of using such an approach and to plan your information technology projects accordingly.

Disclaimer

This document in any form, software, or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply.

This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle.

This document is not part of your license agreement, nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle. Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

Table of contents

List of images

List of tables

No table of figures entries found.

Introduction

Oracle Private Cloud Appliance (PCA) is an Oracle Cloud Infrastructure (OCI)-compatible Engineered System, providing a fast and efficient infrastructure for modern software and business applications. Oracle Private Cloud Appliance has the same infrastructure constructs with APIs and SDKs compatible with OCI. This enables customers to adopt a “develop once, deploy anywhere” approach to rapidly design and develop high-performance applications and middleware.

Advantages of Oracle Private Cloud Appliance

Oracle Private Cloud Appliance (PCA) is an Oracle Engineered System designed for implementing the application and middleware tiers. PCA is an integrated hardware and software system that reduces infrastructure complexity and deployment time for virtualized workloads in private clouds. It is a complete platform for a wide range of application types and workloads, with built-in management, compute, storage, and networking resources. PCA provides excellent performance and other system properties for hosting a broad range of applications.

Oracle Private Cloud Appliance X9-2 is the latest member of the Oracle Private Cloud Appliance product family. It provides cloud and administrative services for a supporting range of workloads including cloud native applications. It makes use of a modern microservices architecture, Kubernetes, and related technologies, for a future-proofed software stack.

A key new feature of Oracle Private Cloud Appliance X9-2, compared to previous versions, is that it delivers private cloud infrastructure and architecture consistent with Oracle Cloud Infrastructure (OCI). Oracle Private Cloud Appliance brings APIs and SDKs compatible with Oracle Cloud Infrastructure (OCI) to an on-premises implementation at rack scale, making workloads, user experience, tool sets and skills portable between private and public clouds. Oracle Private Cloud Appliance can be paired with Oracle Exadata to create an ideal infrastructure for scalable, multitier applications. Customers preferring or requiring an on-premises solution can realize the operational benefits of public cloud deployments using Oracle Private Cloud Appliance X9-2.

Scope and Content

This document consists of four separate sections, each providing a detailed step-by-step guide on a specific key area as to how customers with one or more Oracle Private Cloud Appliance X9-2 systems can install and setup a single, external Grafana Server service to provide monitoring and alerting services across their Oracle Private Cloud Appliance X9-2 estate.

- The first two sections provide instructions on base system preparation, installation, and the base configuration of the external Grafana Server service.
- The final two sections provide instructions on the detailed configuration of the external Grafana Server service to receive telemetry from multiple Oracle Private Cloud Appliance X9-2 systems and display metrics in Grafana dashboards.

Monitoring and Alerting on Oracle Private Cloud Appliance X9-2

Oracle Private Cloud Appliance X9-2 provides monitoring and alerting capabilities through a fully integrated Grafana service. For customers with multiple Oracle Private Cloud Appliance X9-2 racks, this capability can be further expanded to monitor multiple systems through a single, centralized, external Grafana instance.

For customers who are not using Grafana widely within their organization, a new Grafana instance is likely to be needed to act as the centralized "single window" into their Oracle Private Cloud Appliance X9-2 estate.

- The first section of this document provides a step-by-step guide to creating a new host system for the external Grafana instance. In the example below, this is based on an Oracle Linux 9 update 1 virtual machine and uses a Podman (Docker) Container to provide the runtime Grafana environment.
- The second section of this document provides a step-by-step guide on the installation and configuration of a containerized Grafana Server service based on the basic Oracle Linux system configured previously.
- The third section of this document provides a step-by-step guide to adding multiple Oracle Private Cloud Appliance X9-2 systems to an external Grafana Server service as Prometheus-based data sources.
- The fourth, and final, section of this document provides a step-by-step guide for importing existing Grafana Dashboards from a single Oracle Private Cloud Appliance X9-2 to be used to monitor all the systems via the external Grafana Server service.

Providing a Host System

Create a suitable host system to act as the external Grafana Server. This can be either a physical server, or a virtual machine.

Virtual Host System

In this instance, a virtual machine-based Oracle Linux 9 update 1 system was created to act as the host platform for the external Grafana services.

This system consists of the following:

- 4 vCPU
- 16GB RAM
- 50GB Boot Volume
- 250GB Data Volume

The relevant proxy settings were enabled for:

- dnf / yum services - add the environment variable 'proxy=<local proxy server:port> within the'/etc/yum.conf' configuration file; include the 'proxy_username=' and 'proxy_password=' variables & values if required
- system services - add the environment variables HTTP_PROXY, HTTPS_PROXY & NO_PROXY to the system, with any user / password values required. In this example an '/etc/profile.d/proxy.sh' configuration file was created with the following settings:

```
HTTP_PROXY=<protocol://proxy_host_FQDN:port>
HTTPS_PROXY=<protocol://proxy_host_FQDN:port>
NO_PROXY="127.0.0.1, localhost, <required domains, comma separated>"
export HTTP_PROXY
export HTTPS_PROXY
export NO_PROXY
```

A full system update ('dnf update') was run to bring the system up to the latest rpm package levels.

Install Container Management Tools

The container management tools need to be installed. For Oracle Linux, Podman is used.

Podman provides a lightweight utility to run and manage Open Container Initiative (OCI) compatible containers. As such, a Podman deployment can reuse existing container images that are designed for Kubernetes, Oracle Container Runtime for Docker, and Oracle Cloud Native Environment.

Podman is also intended as a drop-in replacement for Oracle Container Runtime for Docker, so the command-line interface (CLI) functions the same way if the `podman-docker` package is installed.

Install the following rpm packages:

- podman-docker
- container-tools

Use the command `'dnf install podman-docker container-tools'` to complete this task.

The following output should be seen:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# DNF INSTALL PODMAN-DOCKER CONTAINER-TOOLS
LAST METADATA EXPIRATION CHECK: 0:14:51 AGO ON MON 16 JAN 2023 16:23:17 GMT.
DEPENDENCIES RESOLVED.
===================================================================================================================================
 PACKAGE                             ARCHITECTURE               VERSION                REPOSITORY                              SIZE
===================================================================================================================================
INSTALLING:
 CONTAINER-TOOLS                     NOARCH                     1-12.0.1.EL9           OL9_APPSTREAM                          7.9 K
 PODMAN-DOCKER                       NOARCH                     2:4.2.0-7.0.1.EL9_1    OL9_APPSTREAM                           41 K
INSTALLING DEPENDENCIES:
 PODMAN-REMOTE                       X86_64                     2:4.2.0-7.0.1.EL9_1    OL9_APPSTREAM                          8.1 M
 PYTHON3-PODMAN                      NOARCH                     3:4.2.1-1.EL9_1        OL9_APPSTREAM                          265 K
 PYTHON3-PYXDG                       NOARCH                     0.27-3.EL9             OL9_APPSTREAM                          144 K
 PYTHON3-TOML                        NOARCH                     0.10.2-6.0.1.EL9       OL9_APPSTREAM                           61 K
 SKOPEO                              X86_64                     2:1.9.4-0.1.EL9_1      OL9_APPSTREAM                          6.6 M
 UDICA                               NOARCH                     0.2.6-4.EL9            OL9_APPSTREAM                           74 K
TRANSACTION SUMMARY
===================================================================================================================================
INSTALL  8 PACKAGES
TOTAL DOWNLOAD SIZE: 15 M
INSTALLED SIZE: 52 M
IS THIS OK [Y/N]: Y
DOWNLOADING PACKAGES:
(1/8): PODMAN-DOCKER-4.2.0-7.0.1.EL9_1.NOARCH.RPM                                                      84 KB/S |  41 KB     00:00
(2/8): CONTAINER-TOOLS-1-12.0.1.EL9.NOARCH.RPM                                                         16 KB/S | 7.9 KB     00:00
(3/8): PYTHON3-PODMAN-4.2.1-1.EL9_1.NOARCH.RPM                                                        1.6 MB/S | 265 KB     00:00
(4/8): PYTHON3-PYXDG-0.27-3.EL9.NOARCH.RPM                                                            721 KB/S | 144 KB     00:00
(5/8): PYTHON3-TOML-0.10.2-6.0.1.EL9.NOARCH.RPM                                                       694 KB/S |  61 KB     00:00
(6/8): PODMAN-REMOTE-4.2.0-7.0.1.EL9_1.X86_64.RPM                                                     8.6 MB/S | 8.1 MB     00:00
(7/8): UDICA-0.2.6-4.EL9.NOARCH.RPM                                                                   315 KB/S |  74 KB     00:00
(8/8): SKOPEO-1.9.4-0.1.EL9_1.X86_64.RPM                                                               13 MB/S | 6.6 MB     00:00
-----------------------------------------------------------------------------------------------------------------------------------
TOTAL                                                                                                  12 MB/S |  15 MB     00:01
RUNNING TRANSACTION CHECK
TRANSACTION CHECK SUCCEEDED.
RUNNING TRANSACTION TEST
TRANSACTION TEST SUCCEEDED.
RUNNING TRANSACTION
  PREPARING        :                                                                                                            1/1
  INSTALLING       : UDICA-0.2.6-4.EL9.NOARCH                                                                                   1/8
  INSTALLING       : SKOPEO-2:1.9.4-0.1.EL9_1.X86_64                                                                            2/8
  INSTALLING       : PYTHON3-TOML-0.10.2-6.0.1.EL9.NOARCH                                                                       3/8
  INSTALLING       : PYTHON3-PYXDG-0.27-3.EL9.NOARCH                                                                            4/8
  INSTALLING       : PYTHON3-PODMAN-3:4.2.1-1.EL9_1.NOARCH                                                                      5/8
  INSTALLING       : PODMAN-REMOTE-2:4.2.0-7.0.1.EL9_1.X86_64                                                                   6/8
  INSTALLING       : PODMAN-DOCKER-2:4.2.0-7.0.1.EL9_1.NOARCH                                                                   7/8
  INSTALLING       : CONTAINER-TOOLS-1-12.0.1.EL9.NOARCH                                                                        8/8
  RUNNING SCRIPTLET: CONTAINER-TOOLS-1- 12.0.1.EL9.NOARCH                                                                       8/8
  VERIFYING        : CONTAINER-TOOLS-1-12.0.1.EL9.NOARCH                                                                        1/8
  VERIFYING        : PODMAN-DOCKER-2:4.2.0-7.0.1.EL9_1.NOARCH                                                                   2/8
  VERIFYING        : PODMAN-REMOTE-2:4.2.0-7.0.1.EL9_1.X86_64                                                                   3/8
  VERIFYING        : PYTHON3-PODMAN-3:4.2.1-1.EL9_1.NOARCH                                                                      4/8
  VERIFYING        : PYTHON3-PYXDG-0.27-3.EL9.NOARCH                                                                            5/8
  VERIFYING        : PYTHON3-TOML-0.10.2-6.0.1.EL9.NOARCH                                                                       6/8
  VERIFYING        : SKOPEO-2:1.9.4-0.1.EL9_1.X86_64                                                                            7/8
  VERIFYING        : UDICA-0.2.6-4.EL9.NOARCH                                                                                   8/8
INSTALLED:
  CONTAINER-TOOLS-1-12.0.1.EL9.NOARCH         PODMAN-DOCKER-2:4.2.0-7.0.1.EL9_1.NOARCH    PODMAN-REMOTE-2:4.2.0-7.0.1.EL9_1.X86_64

  PYTHON3-PODMAN-3:4.2.1-1.EL9_1.NOARCH       PYTHON3-PYXDG-0.27-3.EL9.NOARCH             PYTHON3-TOML-0.10.2-6.0.1.EL9.NOARCH

  SKOPEO-2:1.9.4-0.1.EL9_1.X86_64             UDICA-0.2.6-4.EL9.NOARCH
COMPLETE!
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Verify the status of the podman service using the `'podman info'` command:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN INFO
HOST:
  ARCH: AMD64
  BUILDAHVERSION: 1.27.1
  CGROUPCONTROLLERS:
  - CPUSET
  - CPU
  - IO
  - MEMORY
  - HUGETLB
  - PIDS
  - RDMA
  - MISC
  CGROUPMANAGER: SYSTEMD
  CGROUPVERSION: V2
  CONMON:
    PACKAGE: CONMON-2.1.4-1.EL9.X86_64
    PATH: /USR/BIN/CONMON
    VERSION: 'CONMON VERSION 2.1.4, COMMIT: 72C22139FBCA75534D8C39FD72457900F499CE2B'
  CPUUTILIZATION:
    IDLEPERCENT: 97.76
    SYSTEMPERCENT: 1.2
    USERPERCENT: 1.04
  CPUS: 4
  DISTRIBUTION:
    DISTRIBUTION: '"OL"'
    VARIANT: SERVER
    VERSION: "9.1"
  EVENTLOGGER: JOURNALD
  HOSTNAME: SRD-EXTERNAL-GRAFANA
  IDMAPPINGS:
    GIDMAP: NULL
    UIDMAP: NULL
  KERNEL: 5.15.0-6.80.3.1.EL9UEK.X86_64
  LINKMODE: DYNAMIC
  LOGDRIVER: JOURNALD
  MEMFREE: 14865276928
  MEMTOTAL: 16292057088
  NETWORKBACKEND: NETAVARK
  OCIRUNTIME:
    NAME: CRUN
    PACKAGE: CRUN-1.5-1.EL9.X86_64
    PATH: /USR/BIN/CRUN
    VERSION: |-
      CRUN VERSION 1.5
      COMMIT: 54EBB8CA8BF7E6DDAE2EB919F5B82D1D96863DEA
      SPEC: 1.0.0
      +SYSTEMD +SELINUX +APPARMOR +CAP +SECCOMP +EBPF +CRIU +YAJL
  OS: LINUX
  REMOTESOCKET:
    PATH: /RUN/PODMAN/PODMAN.SOCK
  SECURITY:
    APPARMORENABLED: FALSE
    CAPABILITIES:
CAP_NET_RAW,CAP_CHOWN,CAP_DAC_OVERRIDE,CAP_FOWNER,CAP_FSETID,CAP_KILL,CAP_NET_BIND_SERVICE,CAP_SETFCAP,CAP_SETGID,CAP_SETPCAP,CAP_S
ETUID,CAP_SYS_CHROOT
    ROOTLESS: FALSE
    SECCOMPENABLED: TRUE
    SECCOMPPROFILEPATH: /USR/SHARE/CONTAINERS/SECCOMP.JSON
    SELINUXENABLED: TRUE
  SERVICEISREMOTE: FALSE
  SLIRP4NETNS:
    EXECUTABLE: /USR/BIN/SLIRP4NETNS
    PACKAGE: SLIRP4NETNS-1.2.0-2.EL9_0.X86_64
    VERSION: |-
      SLIRP4NETNS VERSION 1.2.0
      COMMIT: 656041D45CFCA7A4176F6B7EED9E4FE6C11E8383
      LIBSLIRP: 4.4.0
      SLIRP_CONFIG_VERSION_MAX: 3
      LIBSECCOMP: 2.5.2
  SWAPFREE: 0
  SWAPTOTAL: 0
  UPTIME: 0H 11M 6.00S
PLUGINS:
  AUTHORIZATION: NULL
  LOG:
  - K8S-FILE
  - NONE
  - PASSTHROUGH
  - JOURNALD
  NETWORK:
  - BRIDGE
  - MACVLAN
  VOLUME:
  - LOCAL
REGISTRIES:
  SEARCH:
  - CONTAINER-REGISTRY.ORACLE.COM
  - DOCKER.IO
STORE:
  CONFIGFILE: /ETC/CONTAINERS/STORAGE.CONF
  CONTAINERSTORE:
    NUMBER: 0
    PAUSED: 0
    RUNNING: 0
    STOPPED: 0
  GRAPHDRIVERNAME: OVERLAY
  GRAPHOPTIONS:
    OVERLAY.MOUNTOPT: NODEV,METACOPY=ON
  GRAPHROOT: /VAR/LIB/CONTAINERS/STORAGE
  GRAPHROOTALLOCATED: 51954630656
  GRAPHROOTUSED: 5100552192
  GRAPHSTATUS:
    BACKING FILESYSTEM: XFS
    NATIVE OVERLAY DIFF: "FALSE"
    SUPPORTS D_TYPE: "TRUE"
    USING METACOPY: "TRUE"
  IMAGECOPYTMPDIR: /VAR/TMP
  IMAGESTORE:
    NUMBER: 0
  RUNROOT: /RUN/CONTAINERS/STORAGE
  VOLUMEPATH: /VAR/LIB/CONTAINERS/STORAGE/VOLUMES
VERSION:
  APIVERSION: 4.2.0
  BUILT: 1669064937
  BUILTTIME: MON NOV 21 21:08:57 2022
  GITCOMMIT: ""
  GOVERSION: GO1.18.4
  OS: LINUX
  OSARCH: LINUX/AMD64
  VERSION: 4.2.0

[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Enable the podman service to auto start on system reboot.

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# SYSTEMCTL ENABLE --NOW PODMAN
CREATED SYMLINK /ETC/SYSTEMD/SYSTEM/DEFAULT.TARGET.WANTS/PODMAN.SERVICE →
/USR/LIB/SYSTEMD/SYSTEM/PODMAN.SERVICE.
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Finally, check the default podman network created during this system build:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN NETWORK LS
NETWORK ID    NAME        DRIVER
2F259BAB93AA  PODMAN      BRIDGE
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN NETWORK INSPECT PODMAN
[
     {
          "NAME": "PODMAN",
          "ID":
"2F259BAB93AAAAA2542BA43EF33EB990D0999EE1B9924B557B7BE53C0B7A1BB9",
          "DRIVER": "BRIDGE",
          "NETWORK_INTERFACE": "PODMAN0",
          "CREATED": "2023-01-17T10:28:53.881947417Z",
          "SUBNETS": [
               {
                    "SUBNET": "10.88.0.0/16",
                    "GATEWAY": "10.88.0.1"
               }
          ],
          "IPV6_ENABLED": FALSE,
          "INTERNAL": FALSE,
          "DNS_ENABLED": FALSE,
          "IPAM_OPTIONS": {
               "DRIVER": "HOST-LOCAL"
          }
     }
]
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

This system is now ready for the download (pull) and operation of podman (docker) containers.

Enable Additional Services

There are several additional services to configure prior to starting work with Grafana. These are optional but will be used elsewhere within this document and have been included for completeness.

Sendmail

As part of this step-by-step guide, the use of email alerting will be shown. To enable this, the `'sendmail'` packages need to be installed and configured.

First, install the sendmail rpm and any dependencies:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# DNF INSTALL SENDMAIL
LAST METADATA EXPIRATION CHECK: 0:47:51 AGO ON TUE 17 JAN 2023 09:23:45 GMT.
DEPENDENCIES RESOLVED.
================================================================================================================================
 PACKAGE                      ARCHITECTURE                VERSION                      REPOSITORY                          SIZE
================================================================================================================================
INSTALLING:
 SENDMAIL                     X86_64                      8.16.1-10.0.1.EL9            OL9_APPSTREAM                      788 K
INSTALLING DEPENDENCIES:
 PROCMAIL                     X86_64                      3.22-56.EL9                  OL9_APPSTREAM                      212 K
 TINYCDB                      X86_64                      0.78-18.EL9                  OL9_APPSTREAM                       39 K

TRANSACTION SUMMARY
================================================================================================================================
INSTALL  3 PACKAGES

TOTAL DOWNLOAD SIZE: 1.0 M
INSTALLED SIZE: 2.0 M
IS THIS OK [Y/N]: Y
DOWNLOADING PACKAGES:
(1/3): TINYCDB-0.78-18.EL9.X86_64.RPM                                                         83 KB/S |  39 KB     00:00
(2/3): PROCMAIL-3.22-56.EL9.X86_64.RPM                                                       326 KB/S | 212 KB     00:00
(3/3): SENDMAIL-8.16.1-10.0.1.EL9.X86_64.RPM                                                 1.1 MB/S | 788 KB     00:00
--------------------------------------------------------------------------------------------------------------------------------
TOTAL                                                                                        1.4 MB/S | 1.0 MB     00:00
RUNNING TRANSACTION CHECK
TRANSACTION CHECK SUCCEEDED.
RUNNING TRANSACTION TEST
TRANSACTION TEST SUCCEEDED.
RUNNING TRANSACTION
  PREPARING        :                                                                                                    1/1
  INSTALLING       : TINYCDB-0.78-18.EL9.X86_64                                                                         1/3
  INSTALLING       : PROCMAIL-3.22-56.EL9.X86_64                                                                        2/3
  RUNNING SCRIPTLET: SENDMAIL-8.16.1-10.0.1.EL9.X86_64                                                                  3/3
  INSTALLING       : SENDMAIL-8.16.1-10.0.1.EL9.X86_64                                                                  3/3
  RUNNING SCRIPTLET: SENDMAIL-8.16.1-10.0.1.EL9.X86_64                                                                  3/3
  VERIFYING        : PROCMAIL-3.22-56.EL9.X86_64                                                                        1/3
  VERIFYING        : SENDMAIL-8.16.1-10.0.1.EL9.X86_64                                                                  2/3
  VERIFYING        : TINYCDB-0.78-18.EL9.X86_64                                                                         3/3

INSTALLED:
  PROCMAIL-3.22-56.EL9.X86_64                  SENDMAIL-8.16.1-10.0.1.EL9.X86_64                  TINYCDB-0.78-18.EL9.X86_64
COMPLETE!
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Then configure sendmail to enable any podman containers to access the host system sendmail service.

This is achieved by adding the podman network into the sendmail configuration. Add a new line into the 'etc/mail/access' configuration file to include the podman default network '10.88.0.0/24' subnet as a RELAY address range.

For example:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# CAT /ETC/MAIL/ACCESS
# CHECK THE /USR/SHARE/DOC/SENDMAIL/README.CF FILE FOR A DESCRIPTION
# OF THE FORMAT OF THIS FILE. (SEARCH FOR ACCESS_DB IN THAT FILE)
# THE /USR/SHARE/DOC/SENDMAIL/README.CF IS PART OF THE SENDMAIL-DOC
# PACKAGE.
#
# IF YOU WANT TO USE AUTHINFO WITH "M:PLAIN LOGIN", MAKE SURE TO HAVE THE
# CYRUS-SASL-PLAIN PACKAGE INSTALLED.
#
# BY DEFAULT WE ALLOW RELAYING FROM LOCALHOST...
CONNECT:LOCALHOST.LOCALDOMAIN           RELAY
CONNECT:LOCALHOST                       RELAY
CONNECT:127.0.0.1                       RELAY
CONNECT:10.88.0                         RELAY
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Email forwarding will now be available to the Grafana Server instance.

For the purpose of this step-by-step guide, a default, but insecure, sendmail configuration is being used.

When deploying within a customer's data center, the appropriate secure configuration should be used.

Cockpit

Within the standard Oracle Linux base installation, the web-based management tool Cockpit (https://cockpit-project.org/) is installed, but unconfigured by default.

This provides a browser-based administration service to enable a GUI-based systems administration capability.

All that is required is to enable the default Cockpit service and configure to auto-start on system reboot.

Running the simple command

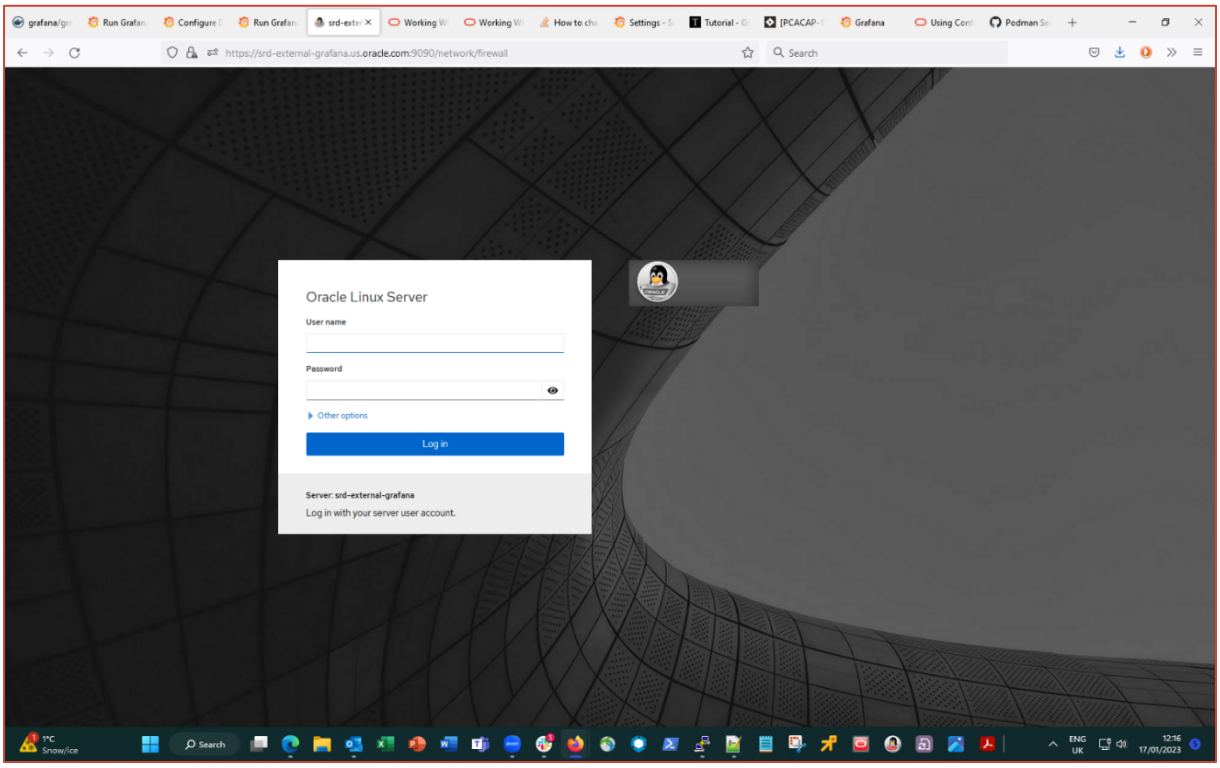
```
systemctl enable --now cockpit.socket
```

Is sufficient to enable this service, which is accessible using the `'https://<system_name>:9090'` URL address.

On some systems, the appropriate firewall setting may need to be enabled as well.

```
firewall-cmd --add-service=cockpit --permanent
```

This then provides the following browser-based access screen.



https://srd-external-grafana.us.oracle.com:9090/network/firewall

Oracle Linux Server

User name

Password

Other options

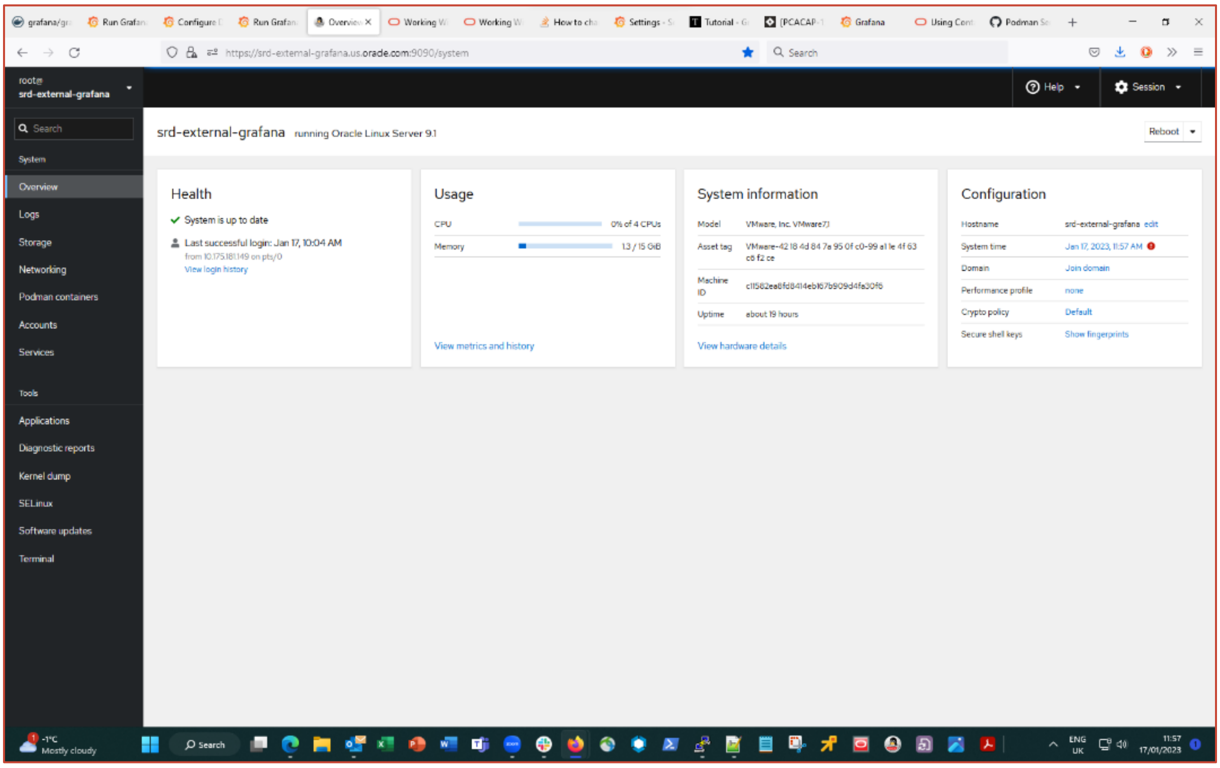
Log in

Server: srd-external-grafana

Log in with your server user account.

Host System – Cockpit login screen

Logging into the Cockpit GUI will present the default Systems Overview dashboard page:



srd-external-grafana running Oracle Linux Server 9.1

Health

✓ System is up to date

Last successful login: Jan 17, 10:04 AM from 10.175.181.149 on pts/0

View login history

Usage

CPU 0% of 4 CPUs

Memory 1.3 / 15 GiB

View metrics and history

System information

Model VMware, Inc. VMware7,1

Asset tag VMware-42 18 4d 84 7a 95 0f c0-99 a1 1e 4f 63 c6 f2 ce

Machine ID c11582ea8fd8414eb167b909d4fa30f6

Uptime about 19 hours

View hardware details

Configuration

Hostname srd-external-grafana edit

System time Jan 17, 2023, 11:57 AM

Domain Join domain

Performance profile none

Crypto policy Default

Secure shell keys Show fingerprints

Host System – Cockpit Systems Overview dashboard

There will be several occasions where the Cockpit Systems Administration tool will be used within this step-by-step guide.

Performance Co-Pilot

Performance Co-Pilot (https://pcp.io/) is a lightweight systems performance toolkit providing an extensible framework to collect, collate, and report on a wide number of system metrics.

Additional 'plugin agents' can be used to extend this toolkit to collect metrics for:

- Databases
- Web servers
- Cluster infrastructure
- Mail systems
- Cisco routers

Details about the capabilities of the Performance Co-Pilot (pcp) framework is well documented. See the following URL for further information:

https://pcp.readthedocs.io/en/latest/

For the Grafana Server host system, the following additional components need to be installed and configured.

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# DNF INSTALL PCP COCKPIT-PCP
LAST METADATA EXPIRATION CHECK: 0:03:24 AGO ON TUE 17 JAN 2023 11:13:56 GMT.
DEPENDENCIES RESOLVED.
==================================================================================================================
 PACKAGE                      ARCHITECTURE      VERSION                       REPOSITORY                     SIZE
==================================================================================================================
INSTALLING:
 COCKPIT-PCP                  X86_64            276.1-1.0.1.EL9               OL9_APPSTREAM                  78 K
 PCP                          X86_64            5.3.7-7.0.2.EL9               OL9_APPSTREAM                 1.7 M
UPGRADING:
 COCKPIT-BRIDGE               X86_64            276.1-1.0.1.EL9               OL9_BASEOS_LATEST             297 K
INSTALLING DEPENDENCIES:
 CYRUS-SASL-SCRAM             X86_64            2.1.27-20.EL9                 OL9_BASEOS_LATEST              28 K
 PCP-CONF                     X86_64            5.3.7-7.0.2.EL9               OL9_APPSTREAM                  38 K
 PCP-LIBS                     X86_64            5.3.7-7.0.2.EL9               OL9_APPSTREAM                 610 K
 PCP-SELINUX                  X86_64            5.3.7-7.0.2.EL9               OL9_APPSTREAM                  39 K

TRANSACTION SUMMARY
==================================================================================================================
INSTALL  6 PACKAGES
UPGRADE  1 PACKAGE

TOTAL DOWNLOAD SIZE: 2.8 M
IS THIS OK [Y/N]: Y
DOWNLOADING PACKAGES:
(1/7): CYRUS-SASL-SCRAM-2.1.27-20.EL9.X86_64.RPM                                        53 KB/S |  28 KB     00:00
(2/7): COCKPIT-PCP-276.1-1.0.1.EL9.X86_64.RPM                                          131 KB/S |  78 KB     00:00
(3/7): PCP-CONF-5.3.7-7.0.2.EL9.X86_64.RPM                                             209 KB/S |  38 KB     00:00
(4/7): PCP-SELINUX-5.3.7-7.0.2.EL9.X86_64.RPM                                          424 KB/S |  39 KB     00:00
(5/7): PCP-LIBS-5.3.7-7.0.2.EL9.X86_64.RPM                                             2.1 MB/S | 610 KB     00:00
(6/7): PCP-5.3.7-7.0.2.EL9.X86_64.RPM                                                  1.7 MB/S | 1.7 MB     00:01
(7/7): COCKPIT-BRIDGE-276.1-1.0.1.EL9.X86_64.RPM                                       1.2 MB/S | 297 KB     00:00
------------------------------------------------------------------------------------------------------------------
TOTAL                                                                                  2.6 MB/S | 2.8 MB     00:01
RUNNING TRANSACTION CHECK
TRANSACTION CHECK SUCCEEDED.
RUNNING TRANSACTION TEST
TRANSACTION TEST SUCCEEDED.
RUNNING TRANSACTION
  PREPARING        :                                                                                          1/1
  UPGRADING        : COCKPIT-BRIDGE-276.1-1.0.1.EL9.X86_64                                                    1/8
  INSTALLING       : PCP-SELINUX-5.3.7-7.0.2.EL9.X86_64                                                       2/8
  RUNNING SCRIPTLET: PCP-SELINUX-5.3.7-7.0.2.EL9.X86_64                                                       2/8


  INSTALLING       : PCP-CONF-5.3.7-7.0.2.EL9.X86_64                                                          3/8
  INSTALLING       : PCP-LIBS-5.3.7-7.0.2.EL9.X86_64                                                          4/8
  INSTALLING       : CYRUS-SASL-SCRAM-2.1.27-20.EL9.X86_64                                                    5/8
  RUNNING SCRIPTLET: PCP-5.3.7-7.0.2.EL9.X86_64                                                               6/8
  INSTALLING       : PCP-5.3.7-7.0.2.EL9.X86_64                                                               6/8
  RUNNING SCRIPTLET: PCP-5.3.7-7.0.2.EL9.X86_64                                                               6/8
  INSTALLING       : COCKPIT-PCP-276.1-1.0.1.EL9.X86_64                                                       7/8
  RUNNING SCRIPTLET: COCKPIT-PCP-276.1-1.0.1.EL9.X86_64                                                       7/8
  CLEANUP          : COCKPIT-BRIDGE-264.1-1.0.3.EL9.X86_64                                                    8/8
  RUNNING SCRIPTLET: COCKPIT-BRIDGE-264.1-1.0.3.EL9.X86_64                                                    8/8
  VERIFYING        : CYRUS-SASL-SCRAM-2.1.27-20.EL9.X86_64                                                    1/8
  VERIFYING        : COCKPIT-PCP-276.1-1.0.1.EL9.X86_64                                                       2/8
  VERIFYING        : PCP-5.3.7-7.0.2.EL9.X86_64                                                               3/8
  VERIFYING        : PCP-CONF-5.3.7-7.0.2.EL9.X86_64                                                          4/8
  VERIFYING        : PCP-LIBS-5.3.7-7.0.2.EL9.X86_64                                                          5/8
  VERIFYING        : PCP-SELINUX-5.3.7-7.0.2.EL9.X86_64                                                       6/8
  VERIFYING        : COCKPIT-BRIDGE-276.1-1.0.1.EL9.X86_64                                                    7/8
  VERIFYING        : COCKPIT-BRIDGE-264.1-1.0.3.EL9.X86_64                                                    8/8

UPGRADED:
  COCKPIT-BRIDGE-276.1-1.0.1.EL9.X86_64
INSTALLED:
  COCKPIT-PCP-276.1-1.0.1.EL9.X86_64  CYRUS-SASL-SCRAM-2.1.27-20.EL9.X86_64  PCP-5.3.7-7.0.2.EL9.X86_64
  PCP-CONF-5.3.7-7.0.2.EL9.X86_64     PCP-LIBS-5.3.7-7.0.2.EL9.X86_64        PCP-SELINUX-5.3.7-7.0.2.EL9.X86_64

COMPLETE!
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Once installed, a few simple configuration changes need to be made.

First, edit the `'/etc/pcp/pmlogger/control.d/local'` configuration file to enable archive log compression. In this example, the default 3-day retention of uncompressed logs is set by removing the hash ('#') at the start of the PCP_COMPRESSAFTER=3' line:

```
#
# PCP ARCHIVE LOGGING CONFIGURATION/CONTROL
#
# SEE ../CONTROL FOR A DESCRIPTION OF THE FORMAT
#

# === VARIABLE ASSIGNMENTS ===
#
# DO NOT REMOVE OR EDIT THE FOLLOWING LINE
$VERSION=1.1

# UNCOMMENT ONE OF THE LINES BELOW TO ENABLE/DISABLE COMPRESSION BEHAVIOUR
# THAT IS DIFFERENT TO THE PMLOGGER_DAILY DEFAULT.
# VALUE IS DAYS BEFORE COMPRESSING ARCHIVES, 0 IS IMMEDIATE COMPRESSION,
# "NEVER" OR "FOREVER" SUPPRESSES COMPRESSION.
#
#$PCP_COMPRESSAFTER=0
$PCP_COMPRESSAFTER=3
#$PCP_COMPRESSAFTER=NEVER

# === LOGGER CONTROL SPECIFICATIONS ===
#
#HOST           P?  S?  DIRECTORY                       ARGS

# LOCAL PRIMARY LOGGER
LOCALHOSTNAME   Y   N   PCP_ARCHIVE_DIR/LOCALHOSTNAME   -R -T24H10M -C CONFIG.DEFAULT -V 100MB
```

Then, append the following text `'-i <system IP address>'` to the `'/etc/pcp/pmcd/pmcd.options'` configuration file to enable the pmcd service to be externally addressable by an external collector system.

With this example, the basic system level metrics will be collected using the `'pmcd'` service, and locally retained and archived using the `'pmlogger'` services.

An external system will act as the collection point for multiple system hosts with the `'pmcd'` service operational.

To enable the various pcp services required, use the following commands:

```
systemctl enable --now pmcd pmlogger
```

To check the status of the pmcd services, issue the command `'pcp'`:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PCP
PERFORMANCE CO-PILOT CONFIGURATION ON SRD-EXTERNAL-GRAFANA:

 PLATFORM: LINUX SRD-EXTERNAL-GRAFANA 5.15.0-6.80.3.1.EL9UEK.X86_64 #2 SMP TUE JAN 10 12:56:46 PST 2023 X86_64
 HARDWARE: 4 CPUS, 2 DISKS, 1 NODE, 15537MB RAM
 TIMEZONE: GMT
 SERVICES: PMCD
     PMCD: VERSION 5.3.7-7, 9 AGENTS, 2 CLIENTS
     PMDA: ROOT PMCD PROC PMPROXY XFS LINUX MMV KVM JBD2
 PMLOGGER: PRIMARY LOGGER: /VAR/LOG/PCP/PMLOGGER/SRD-EXTERNAL-GRAFANA/20230117.10.09
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

This shows that the Performance Metrics Collector Daemon (pmcd) has nine associated Performance Metrics Domain Agents (pmda) associated with the service and that the pcp logger service 'pmlogger' is active and recording metrics into the `'/var/log/pcp/pmlogger'` directory.

The command `'pminfo -t'` will list the available metrics (over 2,500+) being collected that can then be reviewed for further analysis.

Please see the Performance Co-Pilot documentation for further information regarding the additional pmda agents that can be installed and enabled.

On some systems, the appropriate firewall setting may need to be enabled as well for external collection & collation of the pmcd metrics.

```
firewall-cmd --add-service=pmcd --permanent
firewall-cmd --add-service=pmproxy --permanent
```

Section References

The following URL's provide links to additional documentation:

- Oracle Linux – Reference Library – https://docs.oracle.com/en/operating-systems/oracle-linux/9/
- Oracle Linux – Podman User Guide – https://docs.oracle.com/en/operating-systems/oracle-linux/podman/
- Oracle Linux – Using the Cockpit Web Console – https://docs.oracle.com/en/operating-systems/oracle-linux/cockpit/
- Cockpit documentation – https://cockpit-project.org/documentation.html
- Performance Co-Pilot documentation – https://pcp.readthedocs.io/en/latest/
- Performance Co-Pilot documentation – Quick Reference Guide – https://pcp.readthedocs.io/en/latest/QG/QuickReferenceGuide.html

Creating an external Grafana Server Service

The first section covered the initial setup of the host system to be used for the external Grafana Server service. Now the external Grafana Server service needs to be enabled.

Grafana provides its own thorough documentation and tutorials on how to install, configure, and administer a Grafana Server instance.

Please see the links at the end of this section for more details.

Download and Initiate (run) the Grafana Container

Once the base system preparations are complete, log into the Docker registry, find, and pull down the Grafana Container image:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN LOGIN DOCKER.IO
USERNAME: STDENNI03
PASSWORD:
LOGIN SUCCEEDED!
[ROOT@SRD-EXTERNAL-GRAFANA ~]#

[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN SEARCH GRAFANA
NAME                                                   DESCRIPTION
DOCKER.IO/GRAFANA/GRAFANA                              THE OFFICIAL GRAFANA DOCKER CONTAINER
DOCKER.IO/GRAFANA/LOKI                                 LOKI - CLOUD NATIVE LOG AGGREGATION BY GRAFA...
DOCKER.IO/GRAFANA/PROMTAIL
DOCKER.IO/GRAFANA/GRAFANA-OSS                          THE REPO FOR THE GRAFANA OSS EDITION
DOCKER.IO/GRAFANA/GRAFANA-IMAGE-RENDERER
DOCKER.IO/GRAFANA/GRAFANA-ENTERPRISE                   OFFICIAL GRAFANA ENTERPRISE DOCKER IMAGE
DOCKER.IO/GRAFANA/FLUENT-BIT-PLUGIN-LOKI
DOCKER.IO/GRAFANA/MIMIR                                HORIZONTALLY SCALABLE, HIGHLY AVAILABLE, MUL...
DOCKER.IO/GRAFANA/AGENT
DOCKER.IO/GRAFANA/TEMPO
DOCKER.IO/GRAFANA/LOKI-CANARY
DOCKER.IO/GRAFANA/GRAFANA-DEV
DOCKER.IO/GRAFANA/GRAFANA-OSS-DEV                      GRAFANA OSS MASTER BUILDS
DOCKER.IO/GRAFANA/TEMPO-QUERY
DOCKER.IO/GRAFANA/LOKI-DOCKER-DRIVER
DOCKER.IO/GRAFANA/GRAFANA-ARM32V7-LINUX
DOCKER.IO/GRAFANA/BUILD-CONTAINER                      CONTAINER USED FOR BUILDING GRAFANA.
DOCKER.IO/GRAFANA/AGENT-OPERATOR
DOCKER.IO/GRAFANA/MIMIRTOOL                            TOOL TO INTERACT WITH USER-FACING GRAFANA MI...
DOCKER.IO/GRAFANA/SYNTHETIC-MONITORING-AGENT           BLACKBOX EXPORTER AGENT WHICH EXECUTES CHECK...
DOCKER.IO/GRAFANA/GRAFANA-IMAGE-TAGS                   REPOSITORY FOR HOLDING PER-ARCHITECTURE IMAG...
DOCKER.IO/GRAFANA/GRAFANA-ENTERPRISE-DEV
DOCKER.IO/GRAFANA/ONCALL
DOCKER.IO/GRAFANA/CORTEX-TOOLS
DOCKER.IO/GRAFANA/GRAFANA-ENTERPRISE-IMAGE-TAGS        REPOSITORY FOR HOLDING PER-ARCHITECTURE IMAG...
[ROOT@SRD-EXTERNAL-GRAFANA ~]#

[ROOT@SRD-EXTERNAL-GRAFANA ~]#  PODMAN PULL GRAFANA/GRAFANA-OSS
¿ DOCKER.IO/GRAFANA/GRAFANA-OSS:LATEST
TRYING TO PULL DOCKER.IO/GRAFANA/GRAFANA-OSS:LATEST...
GETTING IMAGE SOURCE SIGNATURES
COPYING BLOB 9621F1AFDE84 DONE
COPYING BLOB D8D1A816D728 DONE
COPYING BLOB 0F9C8679DE96 DONE
COPYING BLOB 2D9C2CFEF851 DONE
COPYING BLOB AFB29F8D05C6 DONE
COPYING BLOB DB7FBC631880 DONE
COPYING BLOB 09AAC180A7FC DONE
COPYING BLOB 0B0714AC27D7 DONE
COPYING BLOB 1C5EC5C4E84A DONE
COPYING CONFIG 83F377CC32 DONE
WRITING MANIFEST TO IMAGE DESTINATION
STORING SIGNATURES
83F377CC32A015315824550D2D1F22FF313467D4B5D9D23E6E4D95E0345D6169
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

In this instance we are using the Alpine-based Grafana Server OSS latest image (v9.3.2).

Running the Grafana Server service as a Podman (Docker) container provides a repeatable 'runtime' environment. Data persistence is required to help ensure that the configuration and any plugins and static date are retained. To this end, several Podman (storage) volumes need to be created:

A simple '`podman volume ls`' command will show the current defined volumes (of which there are none):

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN VOLUME LIST
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Three independent volumes need to be created for:

- Configuration - '/etc/grafana'
- Data - '/var/lib/grafana'
- Logging - '/var/log/grafana'

The following commands show the creation and details for each of these three volumes:

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN VOLUME LS
DRIVER      VOLUME NAME
LOCAL       GRAFANA-CONFIG
LOCAL       GRAFANA-LOGS
LOCAL       GRAFANA-STORAGE
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN VOLUME INSPECT GRAFANA-CONFIG
[
     {
          "NAME": "GRAFANA-CONFIG",
          "DRIVER": "LOCAL",
          "MOUNTPOINT": "/VAR/LIB/CONTAINERS/STORAGE/VOLUMES/GRAFANA-CONFIG/_DATA",
          "CREATEDAT": "2023-01-16T17:35:06.399889029Z",
          "LABELS": {},
          "SCOPE": "LOCAL",
          "OPTIONS": {},
          "UID": 472,
          "MOUNTCOUNT": 0
     }
]
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN VOLUME INSPECT GRAFANA-LOGS
[
     {
          "NAME": "GRAFANA-LOGS",
          "DRIVER": "LOCAL",
          "MOUNTPOINT": "/VAR/LIB/CONTAINERS/STORAGE/VOLUMES/GRAFANA-LOGS/_DATA",
          "CREATEDAT": "2023-01-16T17:35:16.160771296Z",
          "LABELS": {},
          "SCOPE": "LOCAL",
          "OPTIONS": {},
          "UID": 472,
          "MOUNTCOUNT": 0,
          "NEEDSCOPYUP": TRUE
     }
]
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN VOLUME INSPECT GRAFANA-STORAGE
[
     {
          "NAME": "GRAFANA-STORAGE",
          "DRIVER": "LOCAL",
          "MOUNTPOINT": "/VAR/LIB/CONTAINERS/STORAGE/VOLUMES/GRAFANA-STORAGE/_DATA",
          "CREATEDAT": "2023-01-16T17:35:23.093881926Z",
          "LABELS": {},
          "SCOPE": "LOCAL",
          "OPTIONS": {},
          "UID": 472,
          "MOUNTCOUNT": 0
     }
]
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

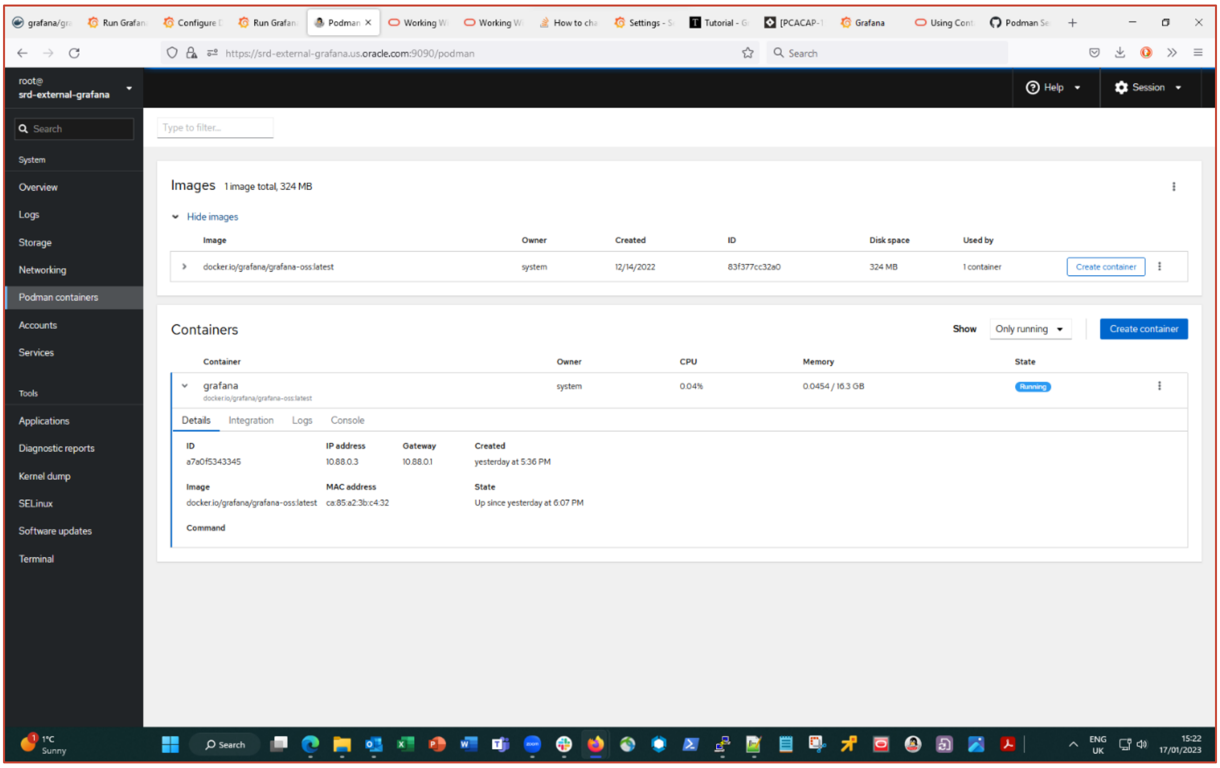
Having created the persistent storage volumes to be utilized by the Grafana Server container, a simple command line invocation of the 'podman run' command will enable this Grafana Server container:

```
[ROOT@SRD-EXTERNAL-GRAFANA VOLUMES]# PODMAN  RUN -D -P 3000:3000 --NAME=GRAFANA -V grafana-config:/etc/grafana -V grafana-
storage:/var/lib/grafana -V grafana-logs:/var/log/grafana GRAFANA/GRAFANA-OSS
A7A0F5343345D8177412CE979983BFC6BB2F39B2B967E670A1796CCD2C1CA125
[ROOT@SRD-EXTERNAL-GRAFANA VOLUMES]#
```

This bring up the Grafana Server container, named `'grafana'` on the host system, with persistent storage being provided for the three file system mount points outlined above.

```
[ROOT@SRD-EXTERNAL-GRAFANA ~]# PODMAN CONTAINER LS
CONTAINER
ID  IMAGE                               COMMAND     CREATED       STATUS           PORTS                  NAMES
A7A0F5343345  docker.io/grafana/grafana-oss:latest             22 HOURS AGO  UP 21 HOURS AGO  0.0.0.0:3000-
>3000/TCP  GRAFANA
[ROOT@SRD-EXTERNAL-GRAFANA ~]#
```

Using the Cockpit Web Console, a more detailed inspection of this container can be achieved:



https://srd-external-grafana.us.oracle.com:9090/podman

root@ srd-external-grafana

Images 1 image total, 324 MB

Hide images

	Image	Owner	Created	ID	Disk space	Used by	
›	docker.io/grafana/grafana-oss:latest	system	12/14/2022	83f377cc32a0	324 MB	1 container	Create container

Containers — Show: Only running — Create container

	Container	Owner	CPU	Memory	State
⌄	grafana docker.io/grafana/grafana-oss:latest	system	0.04%	0.0454 / 16.3 GB	Running

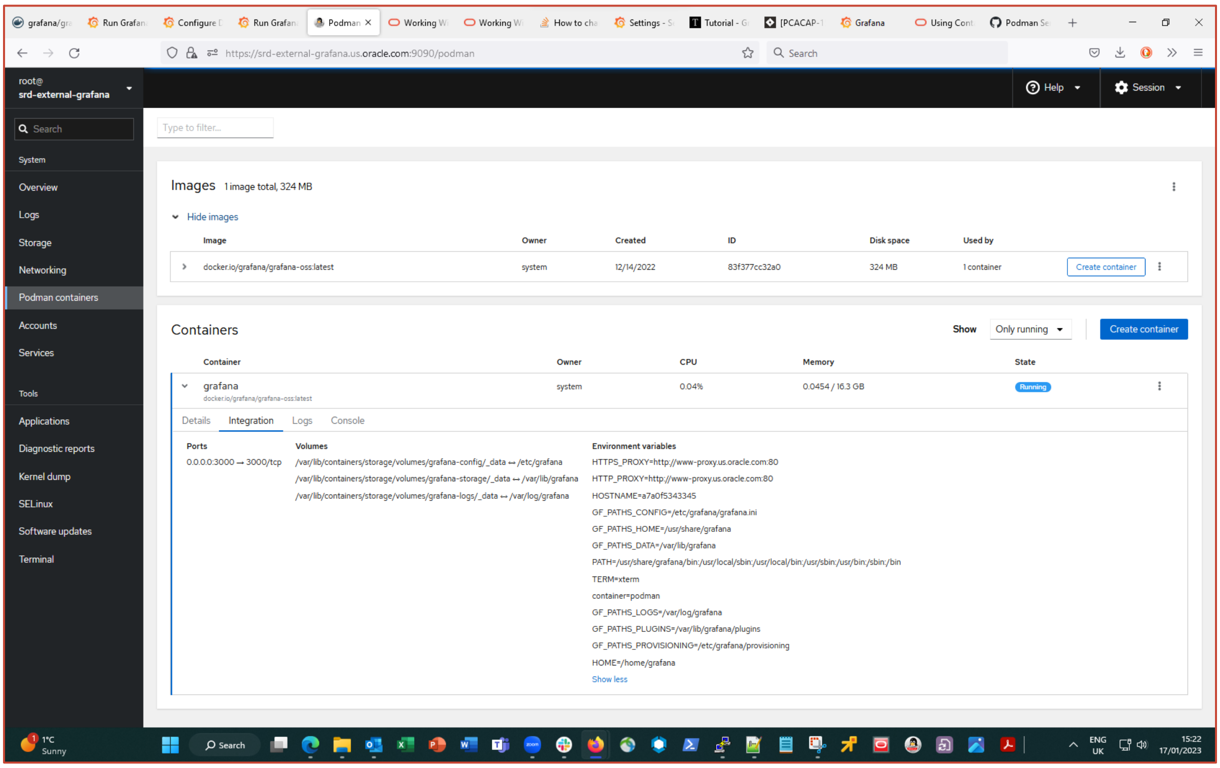
Details | Integration | Logs | Console

ID	IP address	Gateway	Created
a7a0f5343345	10.88.0.3	10.88.0.1	yesterday at 5:36 PM

Image	MAC address	State
docker.io/grafana/grafana-oss:latest	ca:85:a2:3b:c4:32	Up since yesterday at 6:07 PM

Command

Grafana Server – Cockpit Podman Container - Details tab



https://srd-external-grafana.us.oracle.com:9090/podman

root@ srd-external-grafana

Images 1 image total, 324 MB

Hide images

	Image	Owner	Created	ID	Disk space	Used by	
›	docker.io/grafana/grafana-oss:latest	system	12/14/2022	83f377cc32a0	324 MB	1 container	Create container

Containers — Show: Only running — Create container

	Container	Owner	CPU	Memory	State
⌄	grafana docker.io/grafana/grafana-oss:latest	system	0.04%	0.0454 / 16.3 GB	Running

Details | Integration | Logs | Console

Ports

0.0.0.0:3000 → 3000/tcp

Volumes

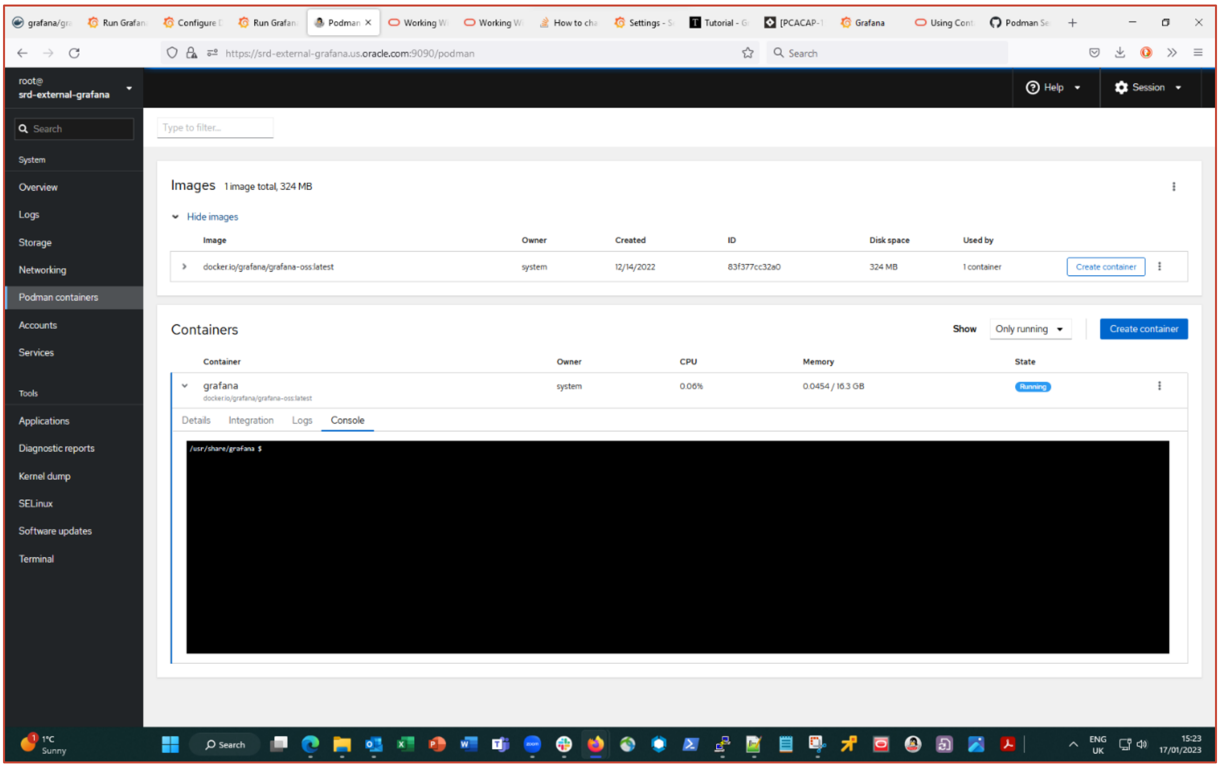
/var/lib/containers/storage/volumes/grafana-config/_data ↔ /etc/grafana
/var/lib/containers/storage/volumes/grafana-storage/_data ↔ /var/lib/grafana
/var/lib/containers/storage/volumes/grafana-logs/_data ↔ /var/log/grafana

Environment variables

```
HTTPS_PROXY=http://www-proxy.us.oracle.com:80
HTTP_PROXY=http://www-proxy.us.oracle.com:80
HOSTNAME=a7a0f5343345
GF_PATHS_CONFIG=/etc/grafana/grafana.ini
GF_PATHS_HOME=/usr/share/grafana
GF_PATHS_DATA=/var/lib/grafana
PATH=/usr/share/grafana/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
TERM=xterm
container=podman
GF_PATHS_LOGS=/var/log/grafana
GF_PATHS_PLUGINS=/var/lib/grafana/plugins
GF_PATHS_PROVISIONING=/etc/grafana/provisioning
HOME=/home/grafana
```

Show less

Grafana Server – Cockpit Podman Container - Integration tab



Images 1 image total, 324 MB

Image	Owner	Created	ID	Disk space	Used by
docker.io/grafana/grafana-oss:latest	system	12/14/2022	83f377cc32a0	324 MB	1 container

Containers

Container	Owner	CPU	Memory	State
grafana docker.io/grafana/grafana-oss:latest	system	0.06%	0.0454 / 16.3 GB	Running

```
/usr/share/grafana $
```

Grafana Server – Cockpit Podman Container - Console tab

As can be seen above, the Cockpit Web Console provides a simple to use graphical view into the Linux Container runtime environment on the host system.

Configure the Grafana Server Services

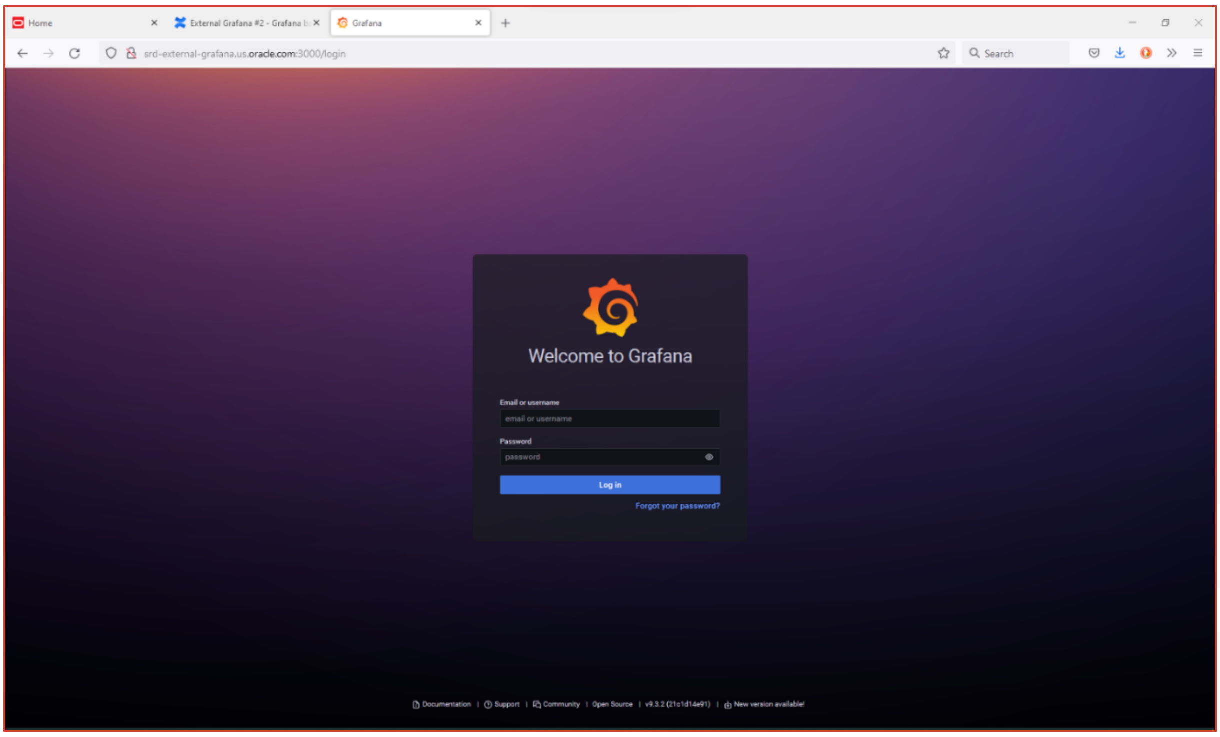
A series of configuration steps will now be described:

Initial Access to the Grafana Server Home Page

Now the podman Grafana Server container is running, access to the Grafana Server service requires a simple login to the correct URL:

http://srd-external-grafana.us.oracle.com:3000/login

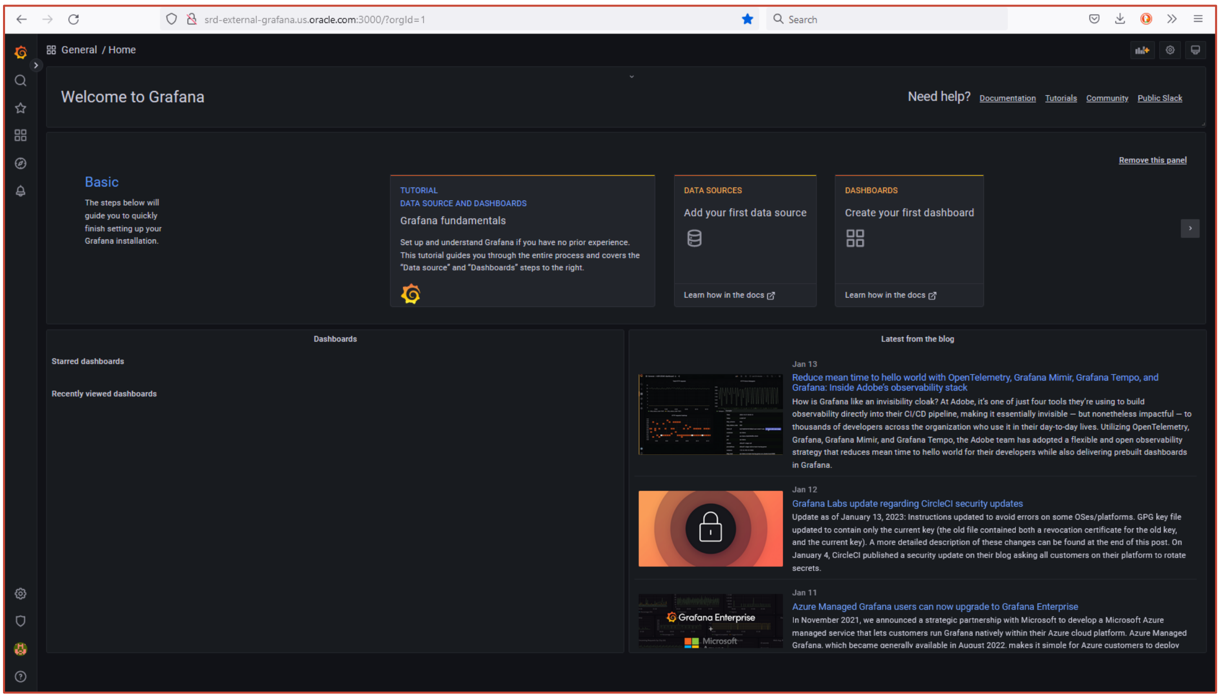
This presents the initial login screen:


Home
External Grafana #2 - Grafana
Grafana
srd-external-grafana.us.oracle.com:3000/login
Search
Welcome to Grafana
Email or username
email or username
Password
password
Log in
Forgot your password?
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91) | New version available!

Grafana Server - Login Screen

By default, the container image has a single user 'admin', with a temporary password 'admin', that will be changed upon first login.

Once the 'admin' user has been logged in and a new password set, the basic Grafana Home screen is presented:



The steps below will guide you to quickly finish setting up your Grafana installation.

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

Jan 13

Reduce mean time to hello world with OpenTelemetry, Grafana Mimir, Grafana Tempo, and Grafana: Inside Adobe's observability stack

How is Grafana like an invisibility cloak? At Adobe, it's one of just four tools they're using to build observability directly into their CI/CD pipeline, making it essentially invisible — but nonetheless impactful — to thousands of developers across the organization who use it in their day-to-day lives. Utilizing OpenTelemetry, Grafana, Grafana Mimir, and Grafana Tempo, the Adobe team has adopted a flexible and open observability strategy that reduces mean time to hello world for their developers while also delivering prebuilt dashboards in Grafana.

Jan 12

Grafana Labs update regarding CircleCI security updates

Update as of January 13, 2023: Instructions updated to avoid errors on some OSes/platforms. GPG key file updated to contain only the current key (the old file contained both a revocation certificate for the old key, and the current key). A more detailed description of these changes can be found at the end of this post. On January 4, CircleCI published a security update on their blog asking all customers on their platform to rotate secrets.

Jan 11

Azure Managed Grafana users can now upgrade to Grafana Enterprise

In November 2021, we announced a strategic partnership with Microsoft to develop a Microsoft Azure managed service that lets customers run Grafana natively within their Azure cloud platform. Azure Managed Grafana, which became generally available in August 2022, makes it simple for Azure customers to deploy

Grafana Server - Initial Home Page

Edit the Grafana Server configuration file to enable email forwarding

Before starting the Grafana Server configuration through the Grafana GUI, there is one group of settings that need to be amended within the `'/etc/grafana/grafana.ini'` Grafana configuration file.

This requires a `'root'` user bash console session to the running Grafana Server container:

- Connect to the running podman container using the 'podman exec' command to access the container 'bash' environment:

```
[root@srd-external-grafana ~]# podman exec -u 0 -it grafana bash
bash-5.1#
```

- Edit the [smtp] section within the '/etc/grafana/grafana.ini' configuration file as follows:

```
[smtp]
enabled = true
host = <Host System Public IP>:25

skip_verify = true
from_address = admin@grafana.srd-external-grafana.us.oracle.com
from_name = Grafana
```

  - Note: use the public IP address of the host system as the 'host' address, with the default port 25 in use to enable email relaying

- Edit the '[emails] section within the '/etc/grafana/grafana.ini' configuration file as follows:

```
[emails]
welcome_email_on_sign_up = true
```

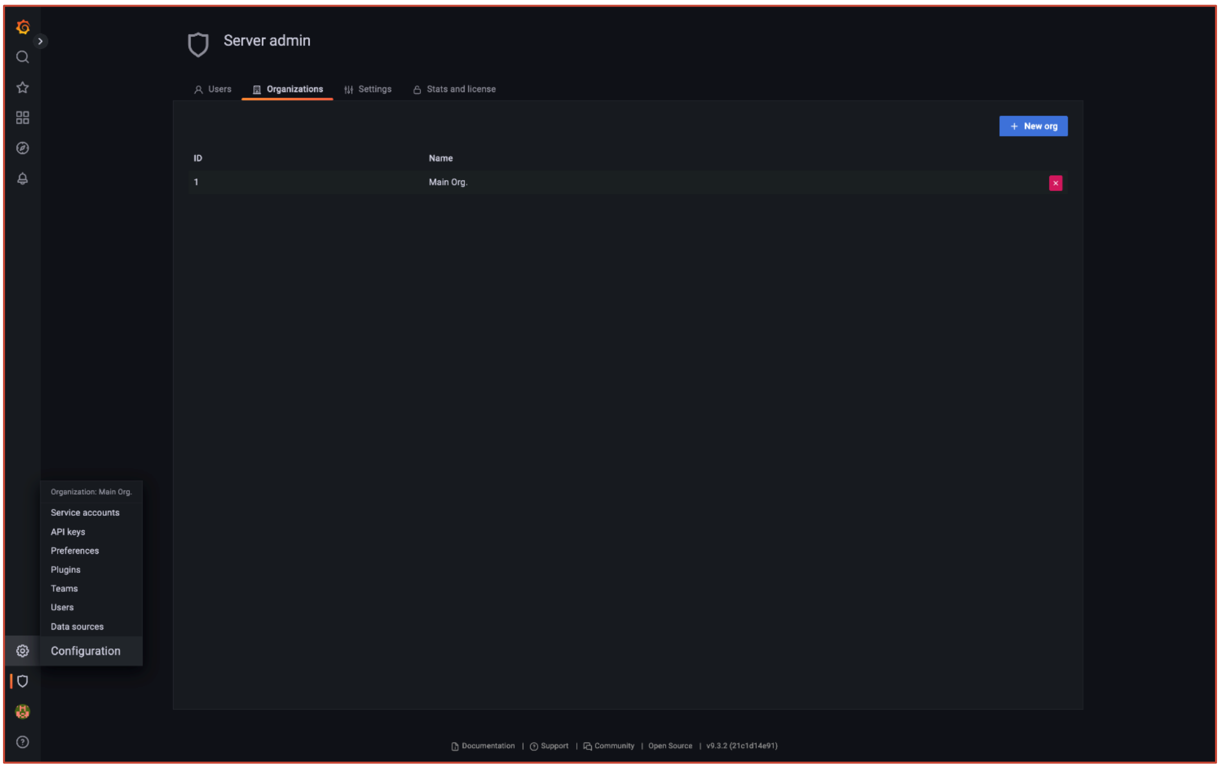
Exit the container 'bash' environment and restart the Grafana Server container to bring the new configuration into place.

Provide Organization, Teams / Group and User information

The following sub-sections describe how to setup the Grafana Server Organization, User and Teams / Group administration structure.

Organization

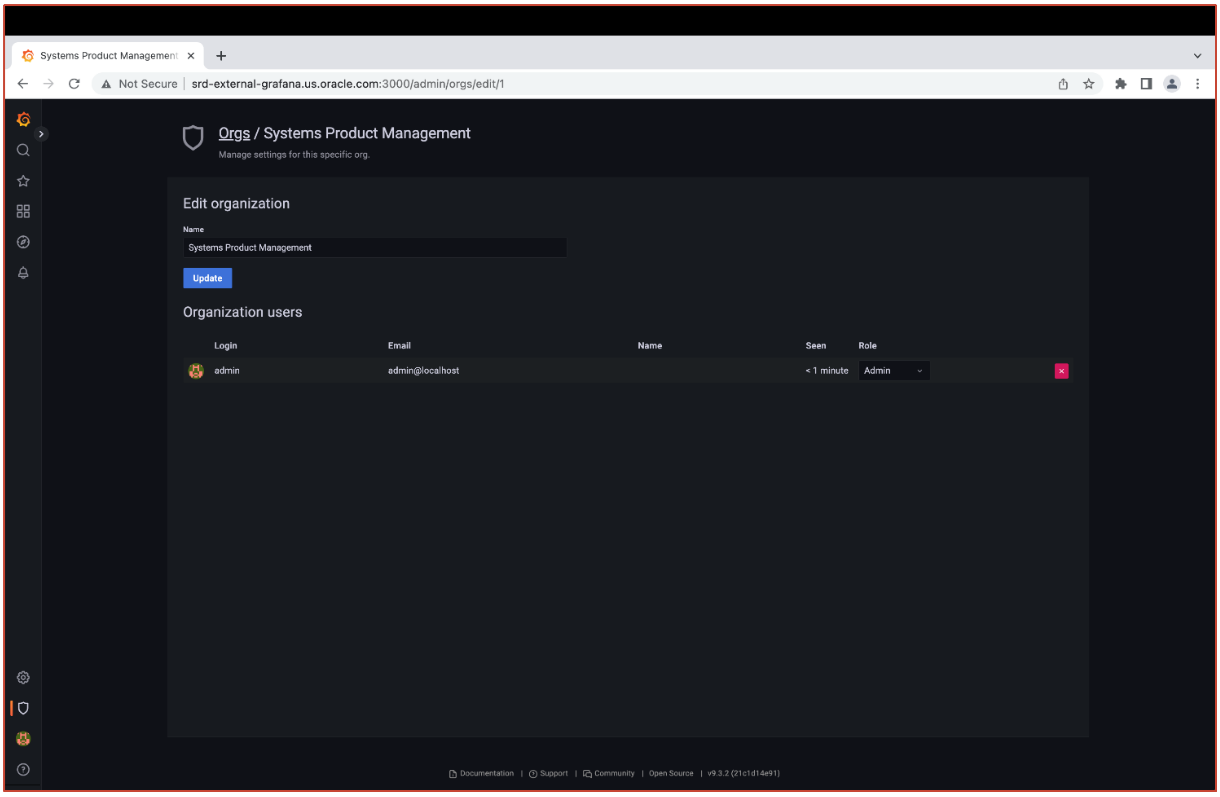
The default Grafana Server installation has a single default organization, 'main.org'



ID	Name
1	Main Org.

Grafana Server – Default Organization

Edit and update to reflect the overarching organization owner for this Grafana Server instance. Double click on the organization (id=1) name and edit as appropriate:



Systems Product Management
Not Secure
srd-external-grafana.us.oracle.com:3000/admin/orgs/edit/1

Orgs / Systems Product Management

Manage settings for this specific org.

Edit organization

Name: Systems Product Management

Update

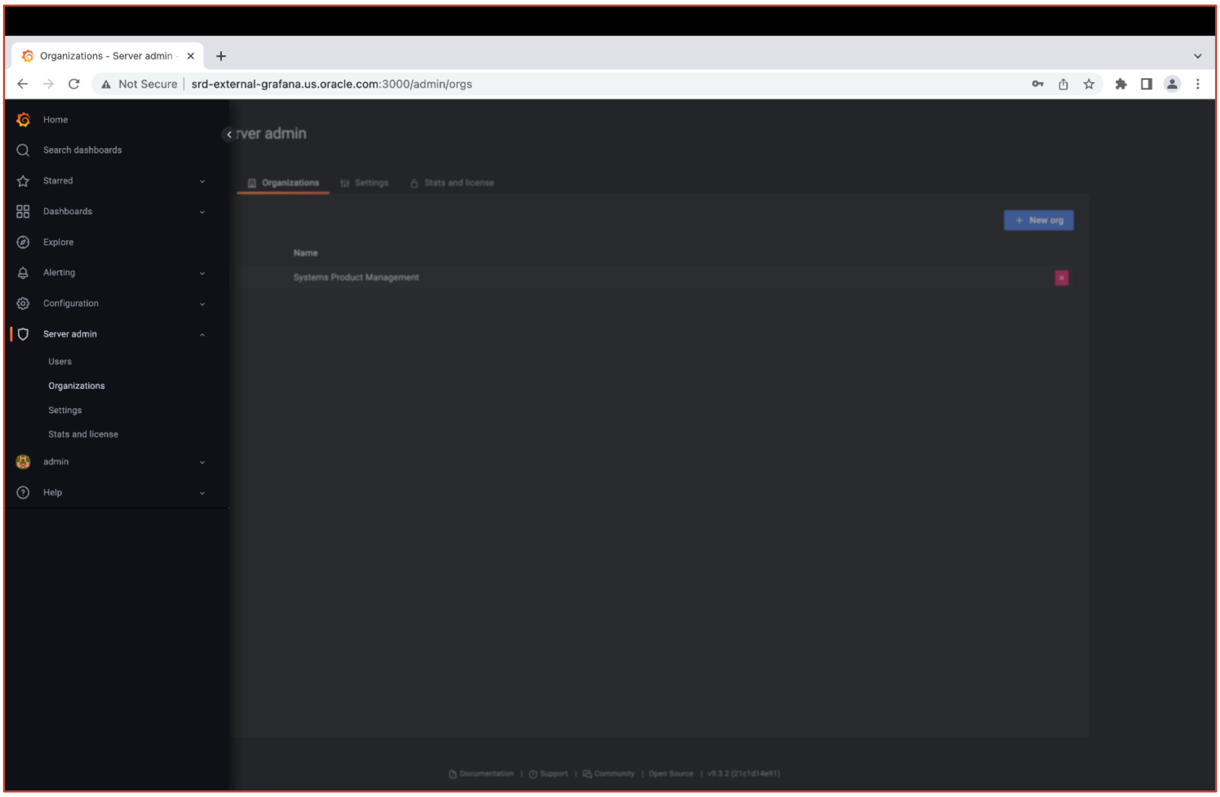
Organization users

Login	Email	Name	Seen	Role
admin	admin@localhost		< 1 minute	Admin

Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Edited Organization

The changed organization name is now displayed:



Name
Systems Product Management

Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Updated Organization details

Users

Now create additional users for accessing the external Grafana Server instance.

Within Grafana, named user and role assignment provide fine-level access control for users accessing the Grafana Server instance. Three levels of user role are available:

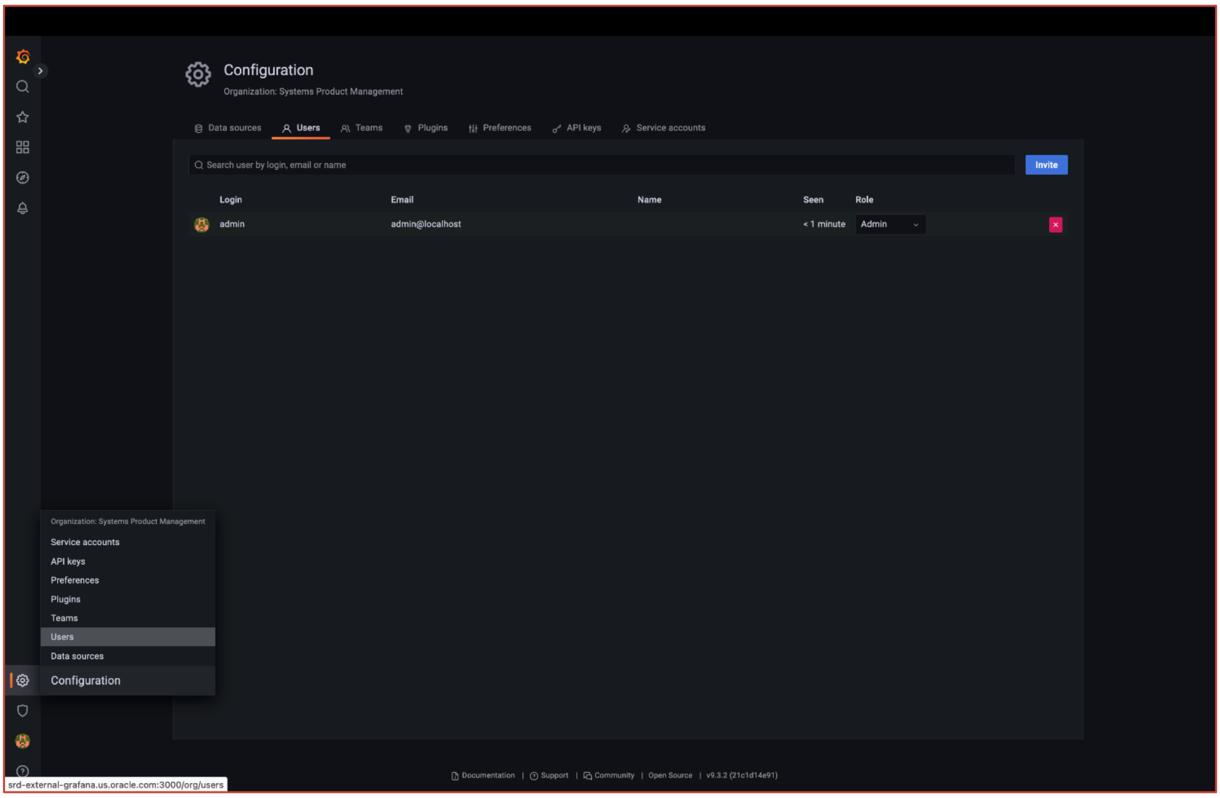
- Viewer
- Editor
- Admin

In addition, there is a fourth role available only to the default admin user (admin@localhost) account

- Grafana Admin

Create a range of local user accounts as needed. Assign the minimum level of access required - e.g., Viewer Access for users that only need to view Grafana dashboards.

Access the Grafana User Administration screen. Initially only the default local admin account is shown:



Configuration

Organization: Systems Product Management

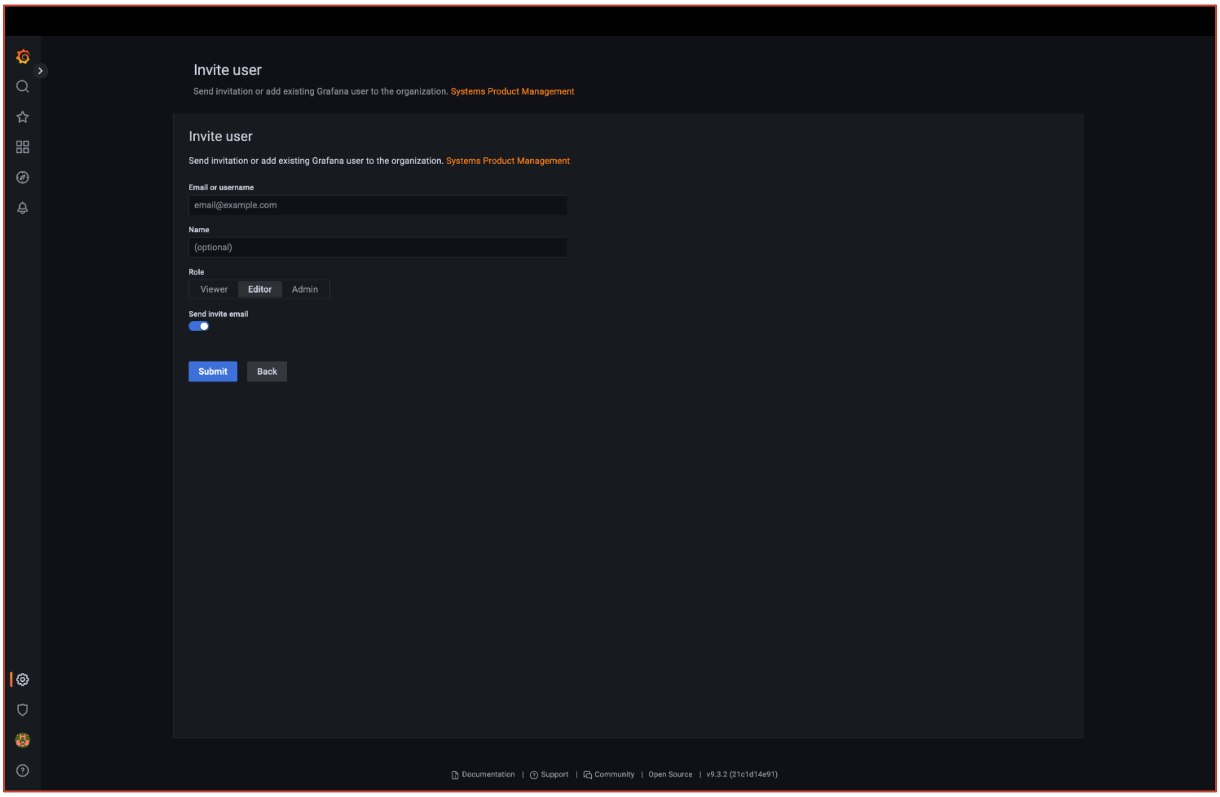
Login	Email	Name	Seen	Role
admin	admin@localhost		< 1 minute	Admin

Data sources
Users
Teams
Plugins
Preferences
API keys
Service accounts
Search user by login, email or name
Invite
Organization: Systems Product Management
Service accounts
API keys
Preferences
Plugins
Teams
Users
Data sources
Configuration
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)
srd-external-grafana.us.oracle.com:3000/org/users

Grafana Server – Default User

Since local email access has been enabled within this Grafana Server instance, use the "Invite" button to add new users.

This displays the screen below:



Invite user

Send invitation or add existing Grafana user to the organization. Systems Product Management

Invite user

Send invitation or add existing Grafana user to the organization. Systems Product Management

Email or username: email@example.com

Name: (optional)

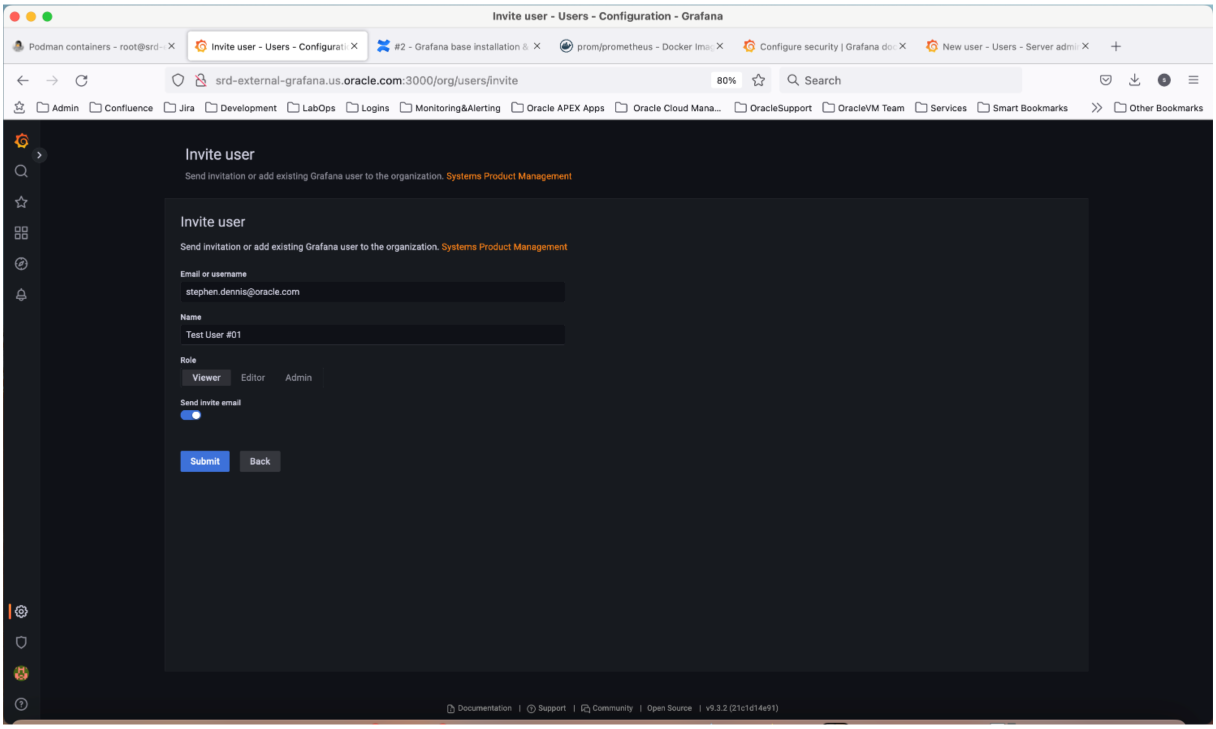
Role: Viewer [✓ Editor] Admin

Send invite email: ●

Submit
Back
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Invite New User

Note the option to select the correct role for each user and provide an email invitation, if required, as well as the username and email details for each new user.



Invite user - Users - Configuration - Grafana

srd-external-grafana.us.oracle.com:3000/org/users/invite

Invite user

Send invitation or add existing Grafana user to the organization. Systems Product Management

Invite user

Send invitation or add existing Grafana user to the organization. Systems Product Management

Email or username: stephen.dennis@oracle.com

Name: Test User #01

Role: [✓ Viewer] Editor Admin

Send invite email: ●

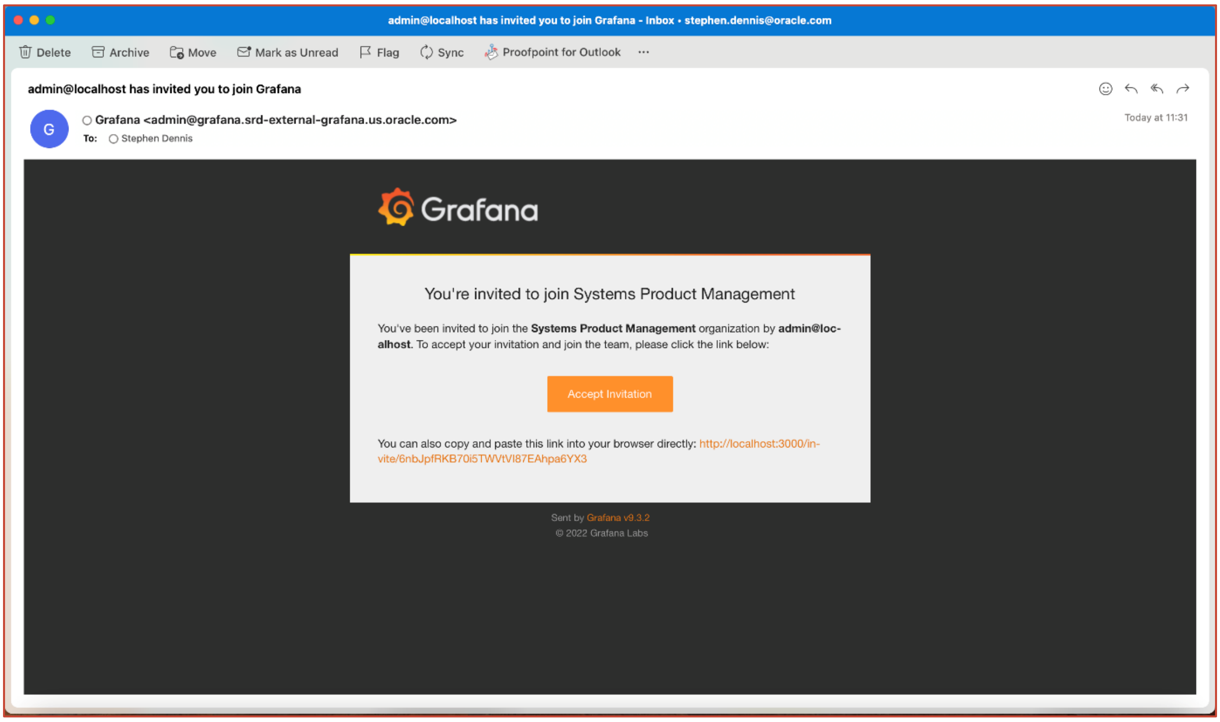
Submit
Back
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Add New User

A single test user with viewer privileges has been created and an email invite sent to the provided email address for acceptance.

A valid email address is required for this approach.

This will generate an email invitation to the new user account, see below:



admin@localhost has invited you to join Grafana

Grafana <admin@grafana.srd-external-grafana.us.oracle.com>

To: Stephen Dennis

Today at 11:31

You're invited to join Systems Product Management

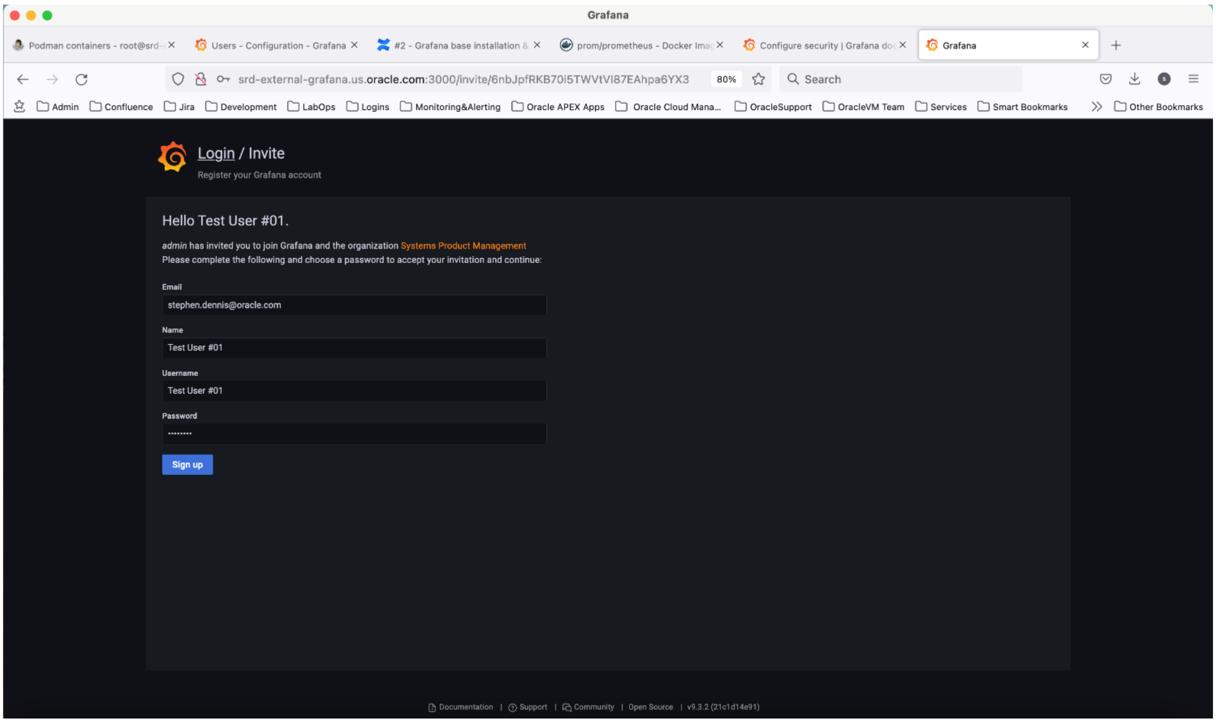
You've been invited to join the **Systems Product Management** organization by **admin@localhost**. To accept your invitation and join the team, please click the link below:

You can also copy and paste this link into your browser directly: http://localhost:3000/invite/6nbJpfRKB70i5TWVtVI87EAhpa6YX3

Sent by Grafana v9.3.2
© 2022 Grafana Labs

Grafana Server – Email Invite

Accepting the invitation enables the user to access the external Grafana Server instance and set several user values, such as username (can default to email address) and password:


Grafana
Podman containers - root@srd-
Users - Configuration - Grafana
#2 - Grafana base installation &
prom/prometheus - Docker Imag
Configure security | Grafana doc
Grafana
srd-external-grafana.us.oracle.com:3000/invite/6nbJpfRKB70i5TWVtVI87EAhpa6YX3
80%
Search
Admin
Confluence
Jira
Development
LabOps
Logins
Monitoring&Alerting
Oracle APEX Apps
Oracle Cloud Mana...
OracleSupport
OracleVM Team
Services
Smart Bookmarks
Other Bookmarks
Sign up
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Login / Invite

Register your Grafana account

Hello Test User #01.

admin has invited you to join Grafana and the organization Systems Product Management
Please complete the following and choose a password to accept your invitation and continue:

Email: stephen.dennis@oracle.com

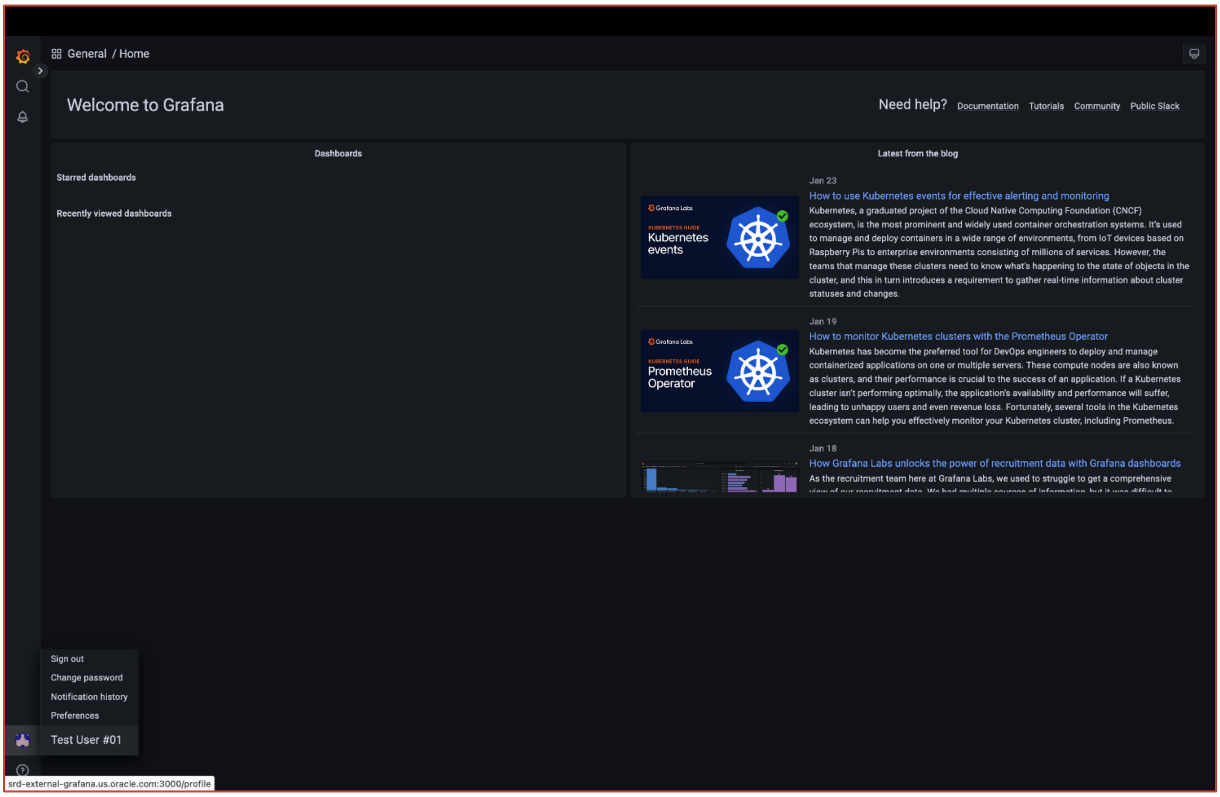
Name: Test User #01

Username: Test User #01

Password: ••••••••

Grafana Server – Accepted Email Invite; Setting Password

Once the password has been set and saved, the user is automatically logged into the Grafana Server instance.



General / Home

Welcome to Grafana

Need help? Documentation Tutorials Community Public Slack

Dashboards

Starred dashboards

Recently viewed dashboards

Latest from the blog

Jan 23

How to use Kubernetes events for effective alerting and monitoring

Kubernetes, a graduated project of the Cloud Native Computing Foundation (CNCF) ecosystem, is the most prominent and widely used container orchestration systems. It's used to manage and deploy containers in a wide range of environments, from IoT devices based on Raspberry Pis to enterprise environments consisting of millions of services. However, the teams that manage these clusters need to know what's happening to the state of objects in the cluster, and this in turn introduces a requirement to gather real-time information about cluster statuses and changes.

Jan 19

How to monitor Kubernetes clusters with the Prometheus Operator

Kubernetes has become the preferred tool for DevOps engineers to deploy and manage containerized applications on one or multiple servers. These compute nodes are also known as clusters, and their performance is crucial to the success of an application. If a Kubernetes cluster isn't performing optimally, the application's availability and performance will suffer, leading to unhappy users and even revenue loss. Fortunately, several tools in the Kubernetes ecosystem can help you effectively monitor your Kubernetes cluster, including Prometheus.

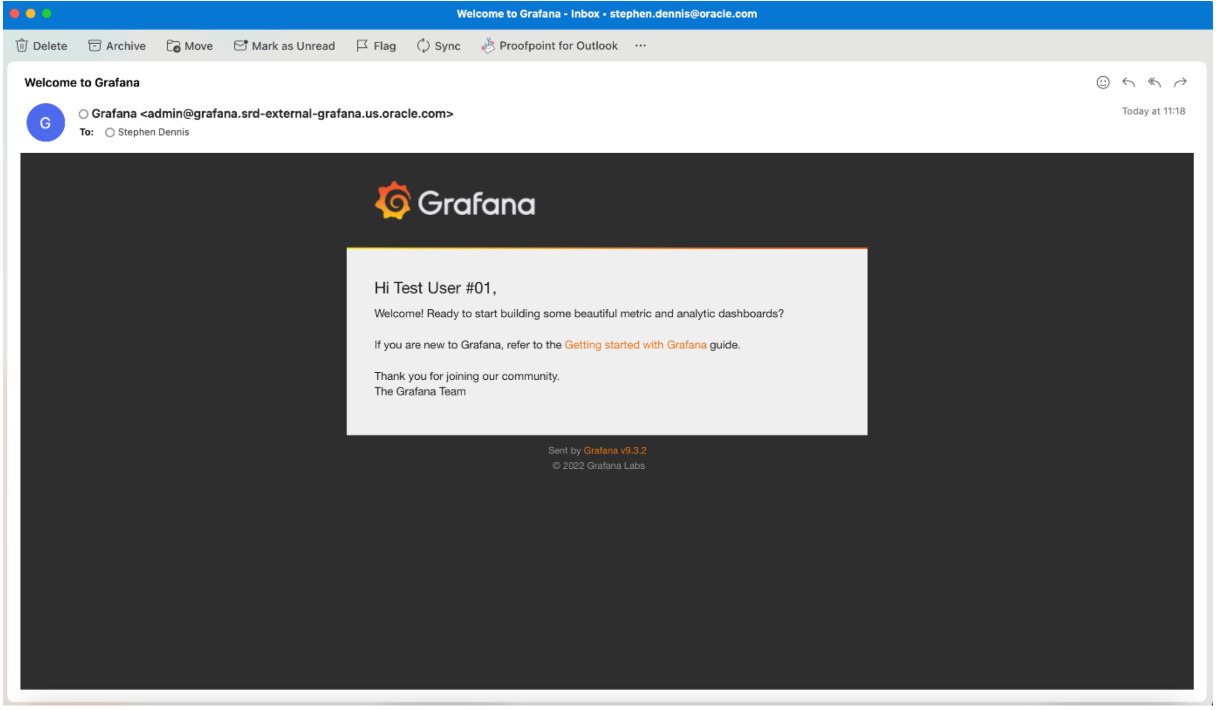
Jan 18

How Grafana Labs unlocks the power of recruitment data with Grafana dashboards

As the recruitment team here at Grafana Labs, we used to struggle to get a comprehensive

Grafana Server – New User Log In

A further email is generated linking to the Grafana documentation 'Getting Started' Guide:



Welcome to Grafana

○ Grafana <admin@grafana.srd-external-grafana.us.oracle.com>

To: ○ Stephen Dennis

Today at 11:18

Grafana

Hi Test User #01,

Welcome! Ready to start building some beautiful metric and analytic dashboards?

If you are new to Grafana, refer to the Getting started with Grafana guide.

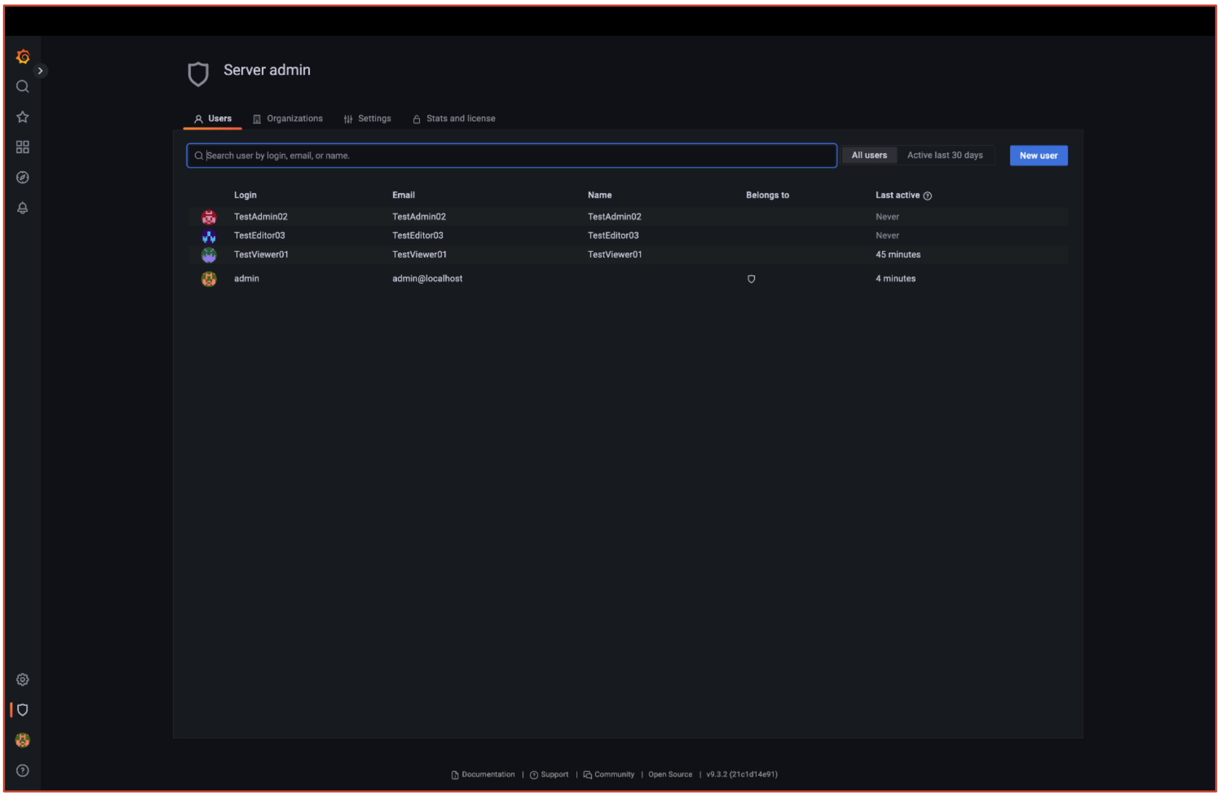
Thank you for joining our community.
The Grafana Team

Sent by Grafana v9.3.2
© 2022 Grafana Labs

Grafana Server – Welcome Email

Now repeat for the number and types of users required.

In this case, three new local user accounts have been created; one for each available Grafana Server user role:



Server admin

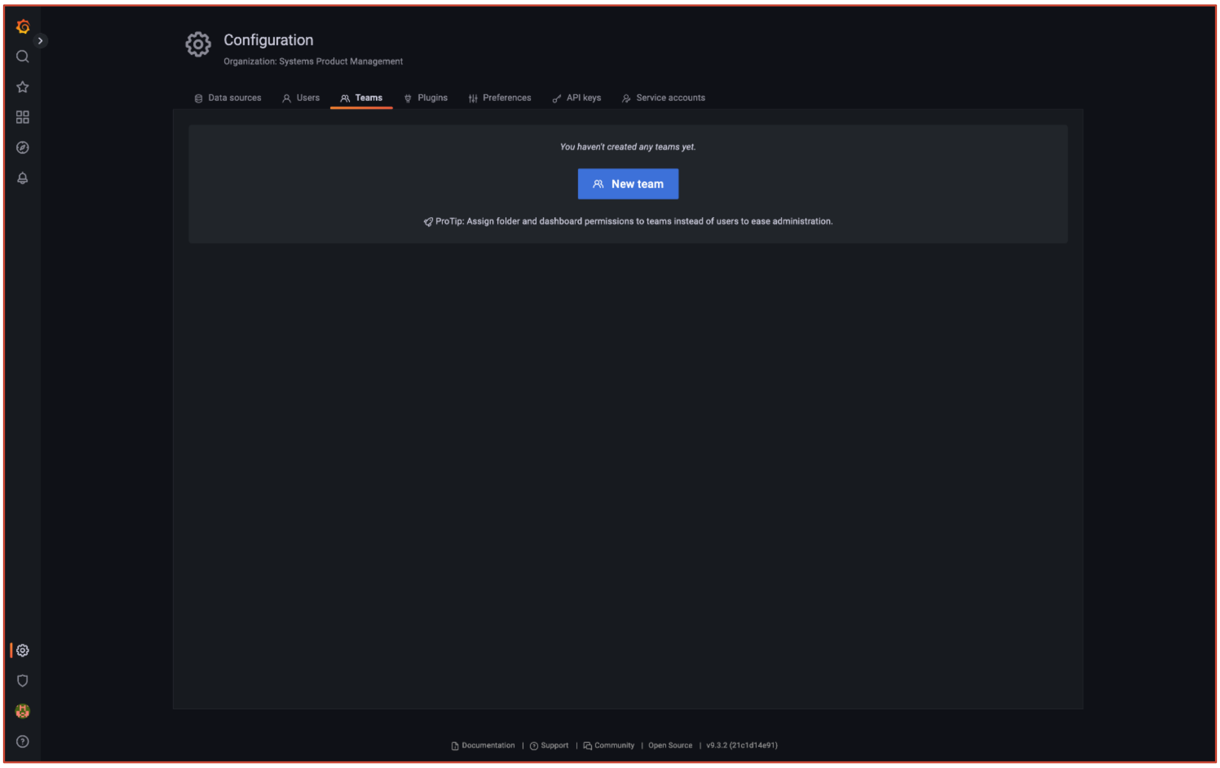
Users
Organizations
Settings
Stats and license
Search user by login, email, or name.
All users
Active last 30 days
New user
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Login	Email	Name	Belongs to	Last active
TestAdmin02	TestAdmin02	TestAdmin02		Never
TestEditor03	TestEditor03	TestEditor03		Never
TestViewer01	TestViewer01	TestViewer01		45 minutes
admin	admin@localhost		🛡	4 minutes

Grafana Server – Multiple Users

Teams / Groups

Now create teams within this organization structure:



Configuration

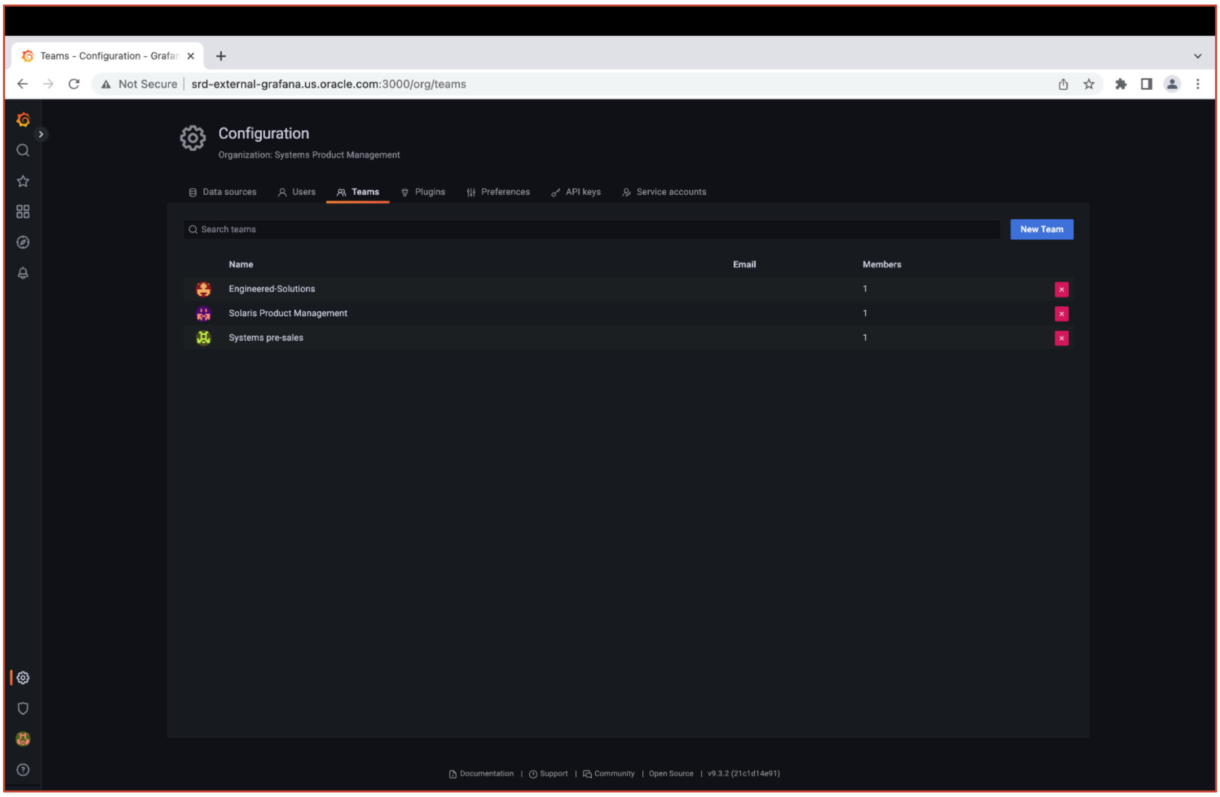
Organization: Systems Product Management

You haven't created any teams yet.

ProTip: Assign folder and dashboard permissions to teams instead of users to ease administration.

Grafana Server – Add Team

Multiple teams can be defined. In this example, three teams have been defined: -


Teams - Configuration - Grafan
Not Secure | srd-external-grafana.us.oracle.com:3000/org/teams
Data sources
Users
Teams
Plugins
Preferences
API keys
Service accounts
Search teams
New Team
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

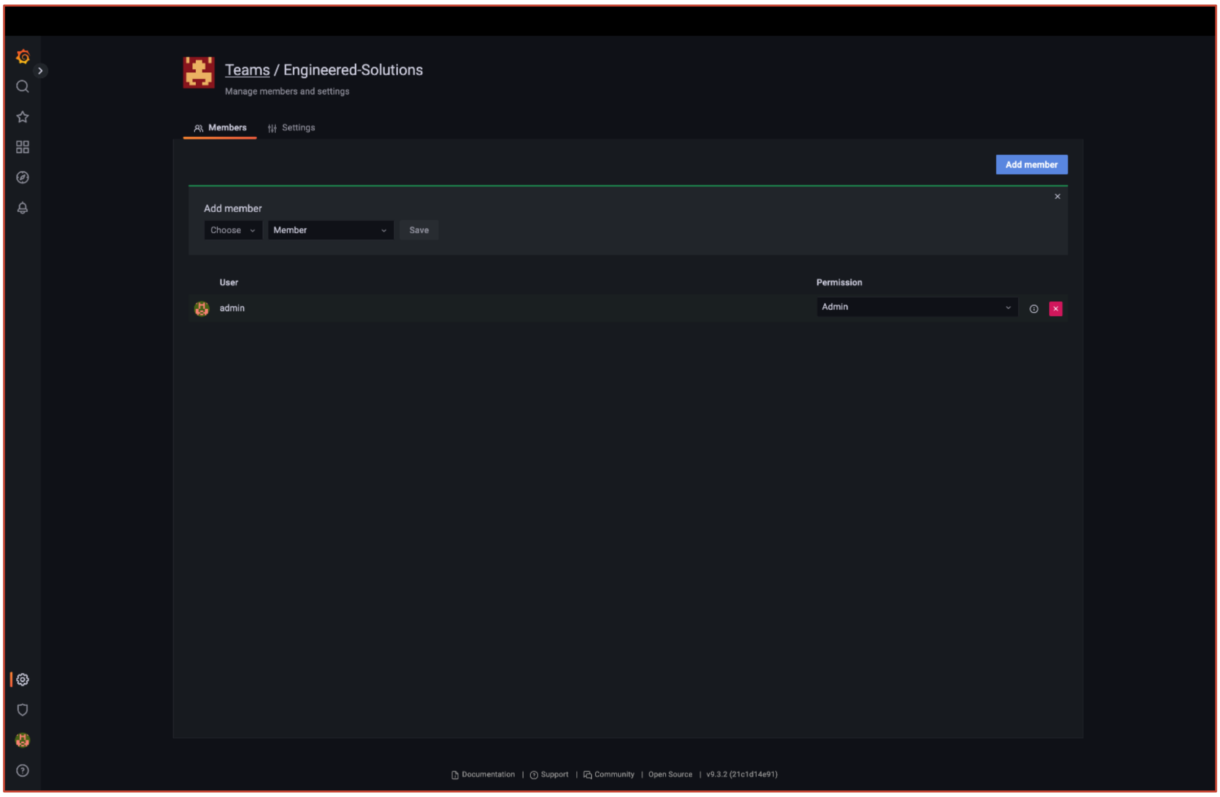
Configuration

Organization: Systems Product Management

Name	Email	Members
Engineered-Solutions		1
Solaris Product Management		1
Systems pre-sales		1

Grafana Server – Added Teams

Once the required teams have been created, the newly created users can be assigned to the relevant team(s):


Members
Settings
Add member
Save
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Teams / Engineered-Solutions

Manage members and settings

Add member

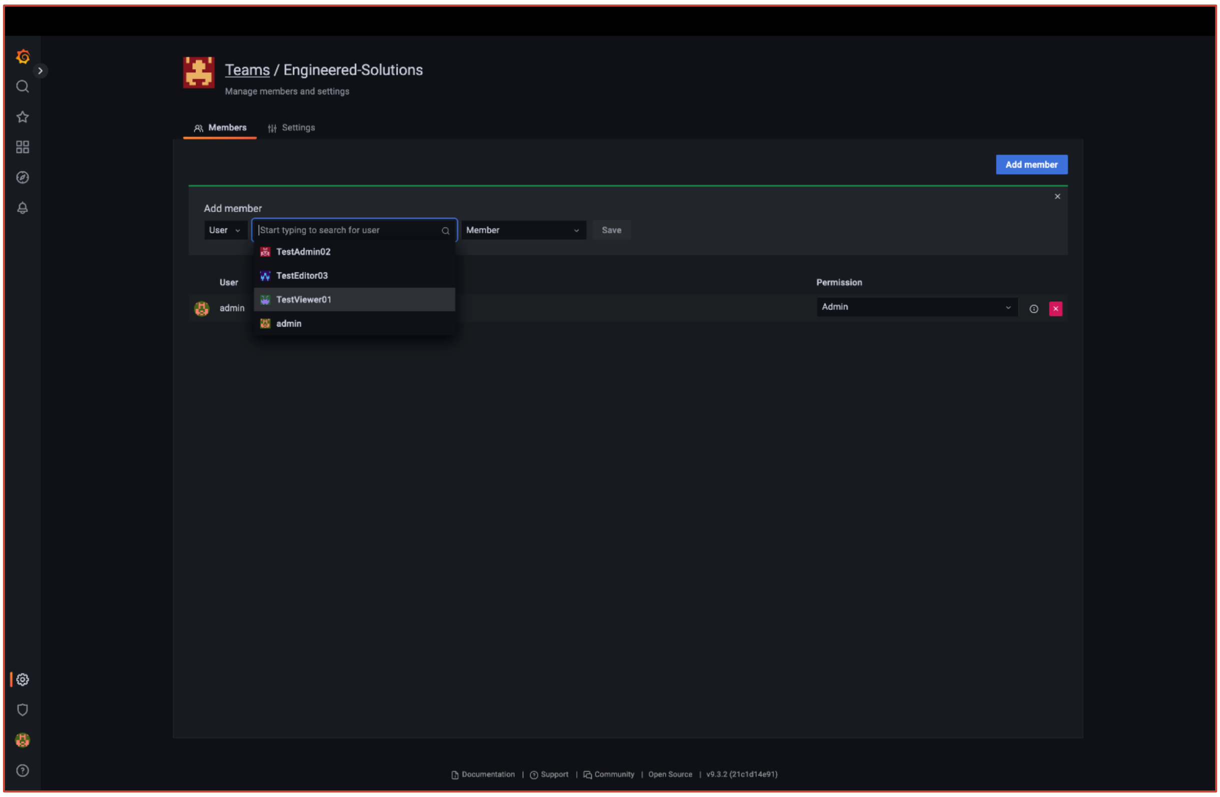
Choose · Member

User	Permission
admin	Admin

Grafana Server – Team Users

Use the Configuration → Teams menu to access this screen.

Now select the required user(s) to be included within this team: -



Teams / Engineered-Solutions

Manage members and settings

Members
Settings
Add member

Add member

User | Start typing to search for user | Member | Save

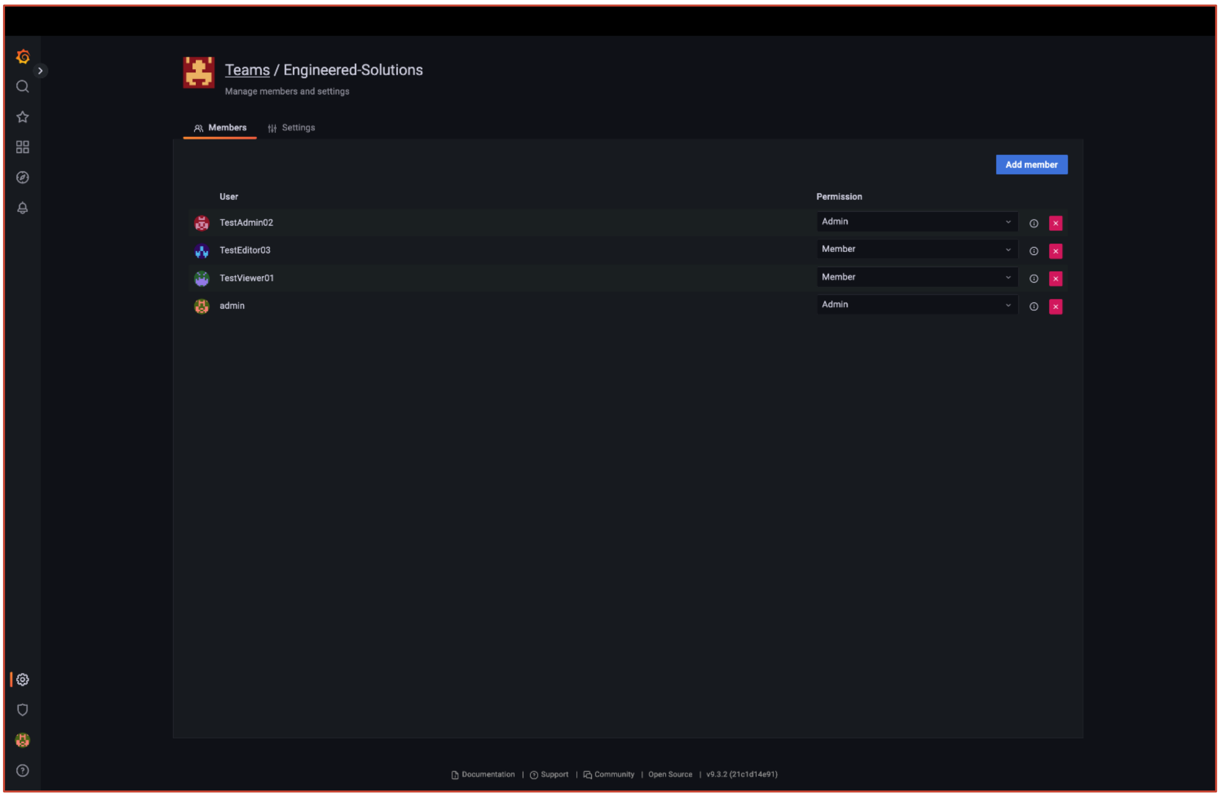
- TestAdmin02
- TestEditor03
- TestViewer01
- admin

User	Permission
admin	Admin

Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Team User Add

Confirm the users have been correctly assigned:



Teams / Engineered-Solutions

Manage members and settings

Members
Settings
Add member

User	Permission
TestAdmin02	Admin
TestEditor03	Member
TestViewer01	Member
admin	Admin

Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Teams and Users verified

Two areas to note:

- A user can be a member of multiple teams
- A user can be assigned either 'member' or 'admin' status for each team

Grafana Server Monitoring and Alerting

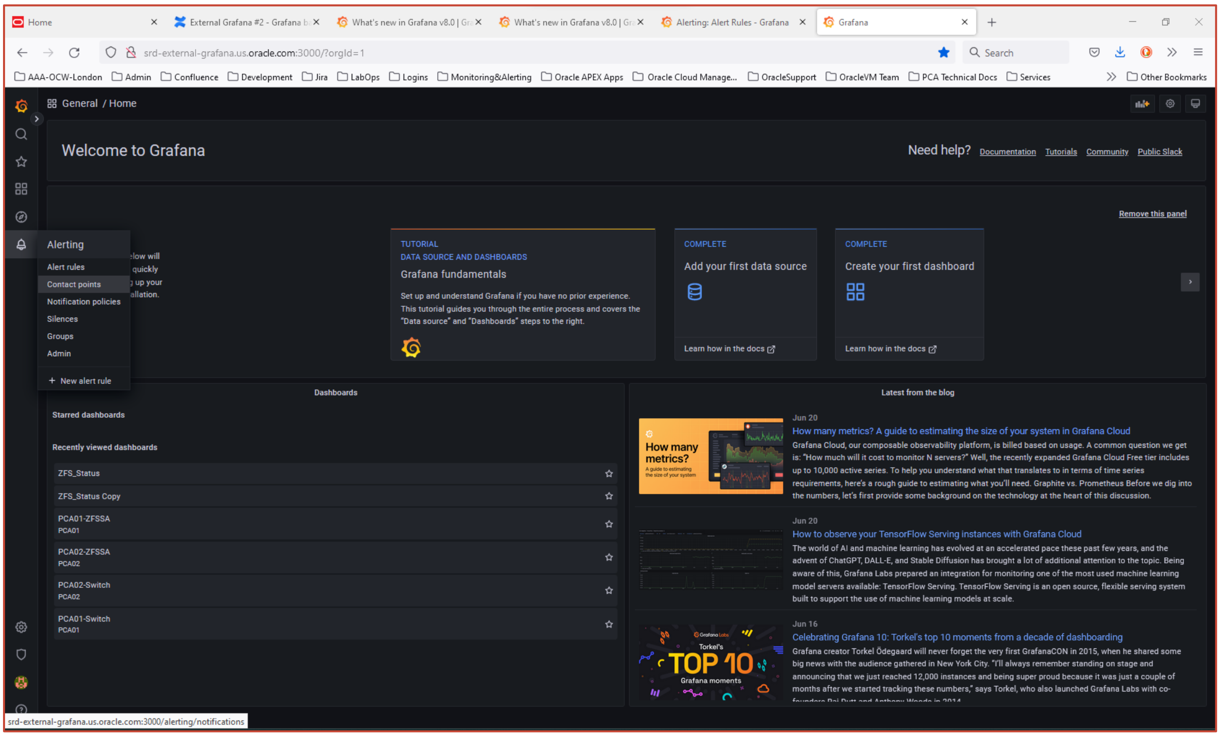
Previous versions of Grafana (v7.5 and below), provided an alerting framework which consisted of two components; ‘Alert Rules’, defined within individual Grafana dashboards, and ‘Notification Channels’, providing one (or more) mechanisms to send any alerts generated to end users.

Since Grafana Server version 8, and above, the capabilities of Grafana to provide a robust monitoring and alerting framework have been significantly improved. The following features are now available:

- Alert Rules – Standard Grafana Dashboard driven Alert Rules
- Notification Policies – New feature
- Contact Points – The replacement for Notification Channels
- Silences – New feature
- Groups – New feature
- Admin – New feature

Please refer to the Grafana Documentation for more details on these new capabilities (https://grafana.com/docs/grafana/latest/alerting/).

The screen shot below shows these alerting menu options:



Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the “Data source” and “Dashboards” steps to the right.

Jun 20
How many metrics? A guide to estimating the size of your system in Grafana Cloud
Grafana Cloud, our composable observability platform, is billed based on usage. A common question we get is: “How much will it cost to monitor N servers?” Well, the recently expanded Grafana Cloud Free tier includes up to 10,000 active series. To help you understand what that translates to in terms of time series requirements, here’s a rough guide to estimating what you’ll need. Graphite vs. Prometheus Before we dig into the numbers, let’s first provide some background on the technology at the heart of this discussion.

Jun 20
How to observe your TensorFlow Serving instances with Grafana Cloud
The world of AI and machine learning has evolved at an accelerated pace these past few years, and the advent of ChatGPT, DALL-E, and Stable Diffusion has brought a lot of additional attention to the topic. Being aware of this, Grafana Labs prepared an integration for monitoring one of the most used machine learning model servers available: TensorFlow Serving. TensorFlow Serving is an open source, flexible serving system built to support the use of machine learning models at scale.

Jun 16
Celebrating Grafana 10: Torkel's top 10 moments from a decade of dashboarding
Grafana creator Torkel Ödegaard will never forget the very first GrafanaCON in 2015, when he shared some big news with the audience gathered in New York City. “I’ll always remember standing on stage and announcing that we just reached 12,000 instances and being super proud because it was just a couple of months after we started tracking these numbers,” says Torkel, who also launched Grafana Labs with co-

Grafana Server – Alerting

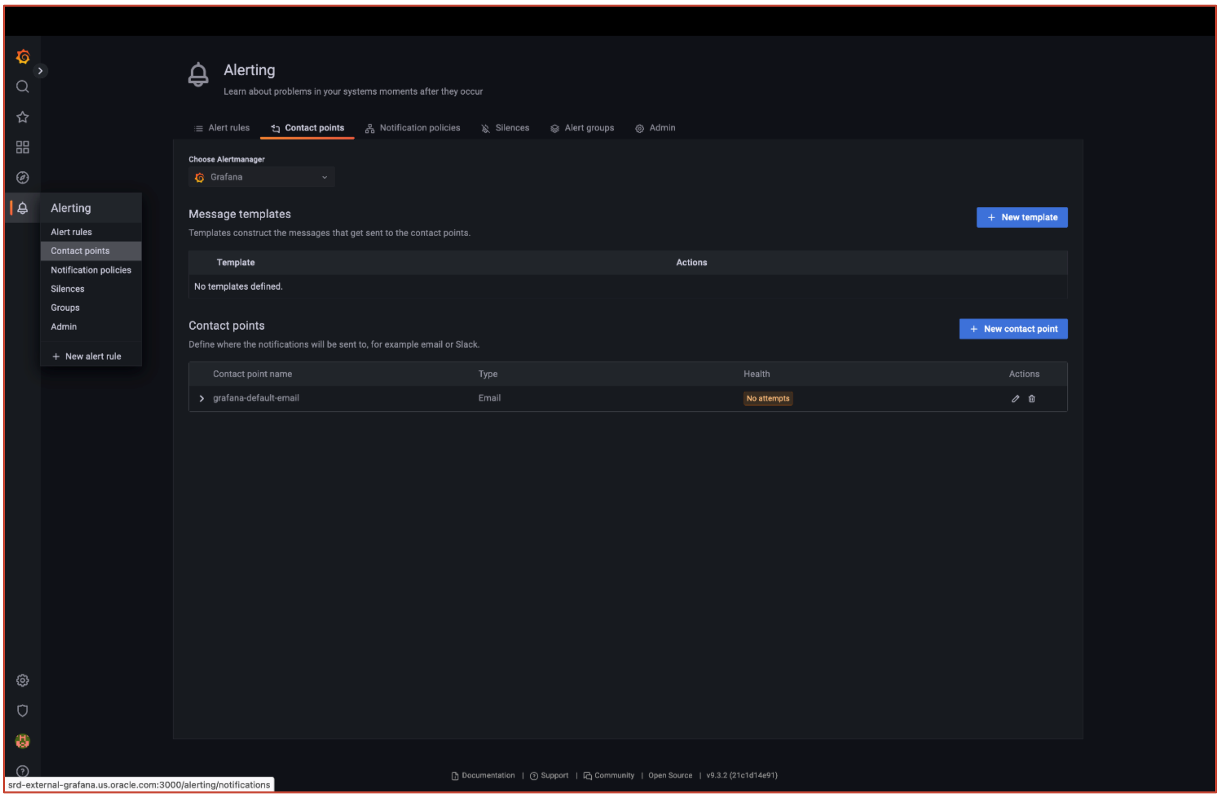
The final configuration change to be made for this new Grafana Server installation is to configure the Contact Points for the Grafana dashboard alerts to use.

For this step-by-step guide, two Contact Points will be created

- Email Alerting
- Webhook (HTTP) Alerting

Email Alerting

By default, a single contact point is provided out-of-the-box. This provides an email alerting mechanism:



Alerting

Learn about problems in your systems moments after they occur

Choose Alertmanager: Grafana

Message templates

Templates construct the messages that get sent to the contact points.

Template	Actions
No templates defined.	

Contact points

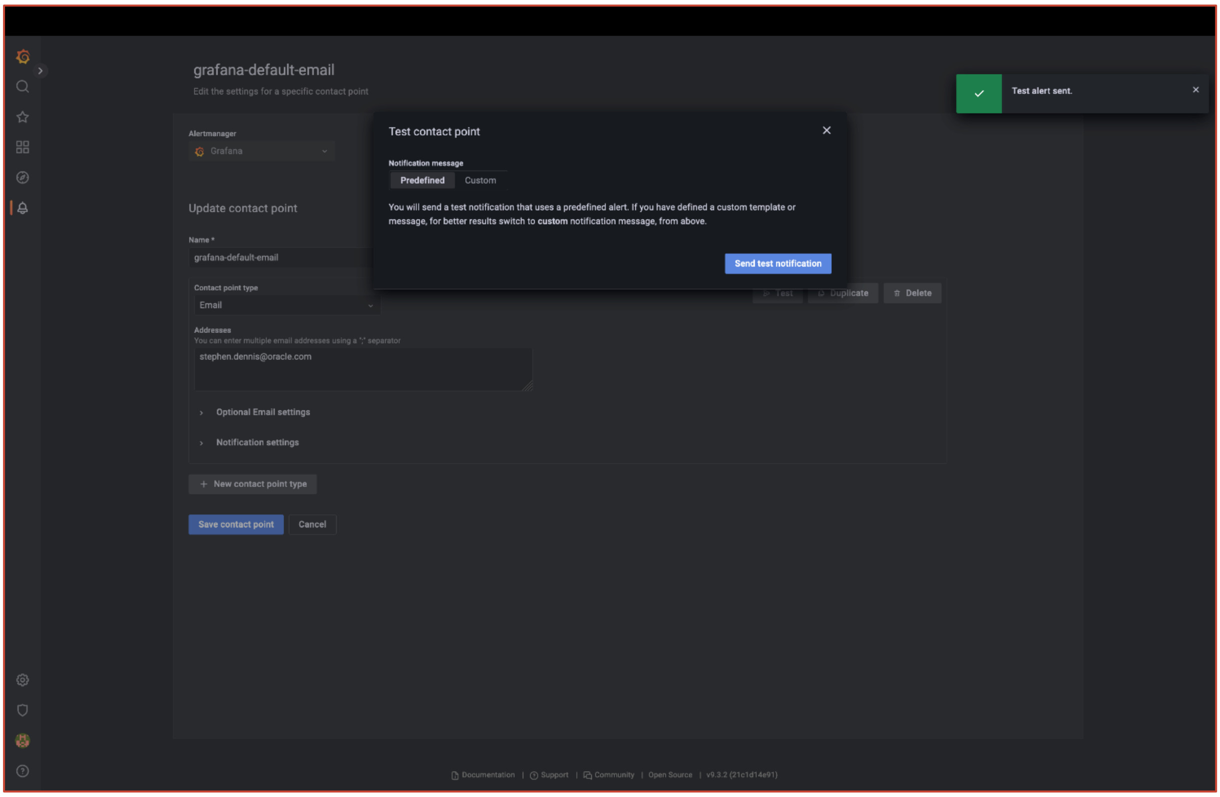
Define where the notifications will be sent to, for example email or Slack.

Contact point name	Type	Health	Actions
grafana-default-email	Email	No attempts	

srd-external-grafana.us.oracle.com:3000/alerting/notifications

Grafana Server – Default Contact Point

Since this Grafana Server instance has an email forwarding capability enabled, a simple edit of the default 'grafana-default-email' contact point to provide a recipient email address and a test alert is all that is required:



grafana-default-email

Edit the settings for a specific contact point

Alertmanager: Grafana

Update contact point

Name *: grafana-default-email

Contact point type: Email

Addresses

You can enter multiple email addresses using a ";" separator

stephen.dennis@oracle.com

Test contact point

Notification message: Predefined | Custom

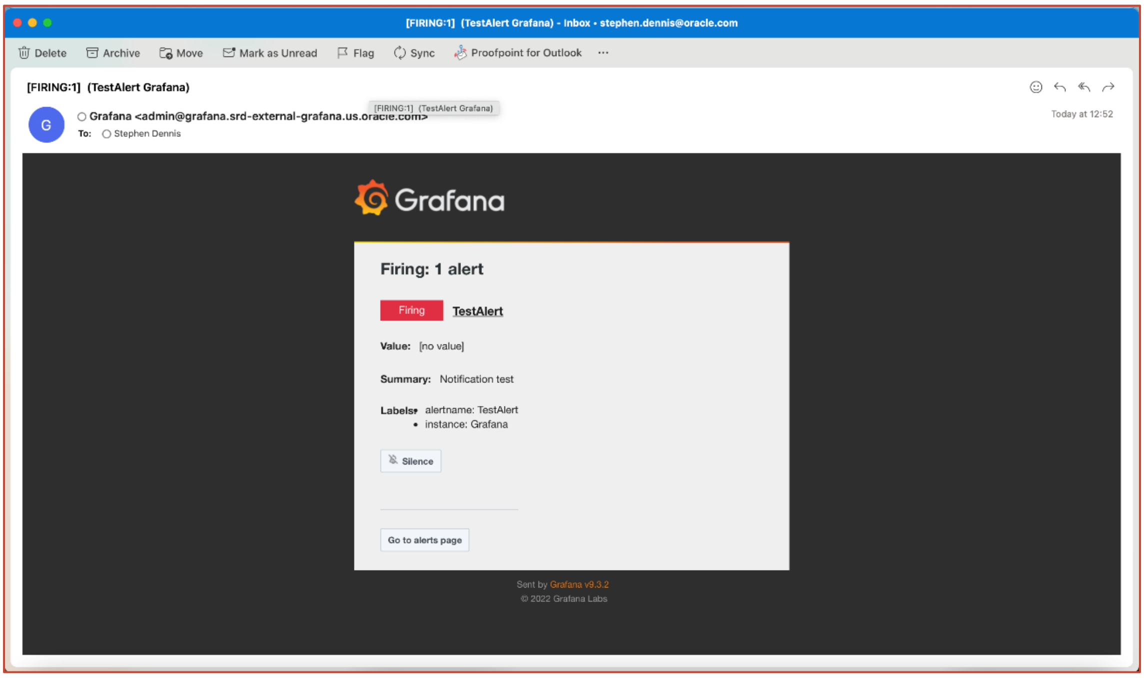
You will send a test notification that uses a predefined alert. If you have defined a custom template or message, for better results switch to **custom** notification message, from above.

Test alert sent.

Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)

Grafana Server – Email Contact Point – Configure & Test

A single email address has been added to the configuration. This could be a list of named users. or a valid email group. A test notification has been sent and confirmation of the Test Alert received within Grafana.



[FIRING:1] (TestAlert Grafana) - Inbox • stephen.dennis@oracle.com

[FIRING:1] (TestAlert Grafana)

Grafana <admin@grafana.srd-external-grafana.us.oracle.com>

To: Stephen Dennis

Today at 12:52

Grafana

Firing: 1 alert

Firing TestAlert

Value: [no value]

Summary: Notification test

Labels: alertname: TestAlert
- instance: Grafana

Sent by Grafana v9.3.2
© 2022 Grafana Labs

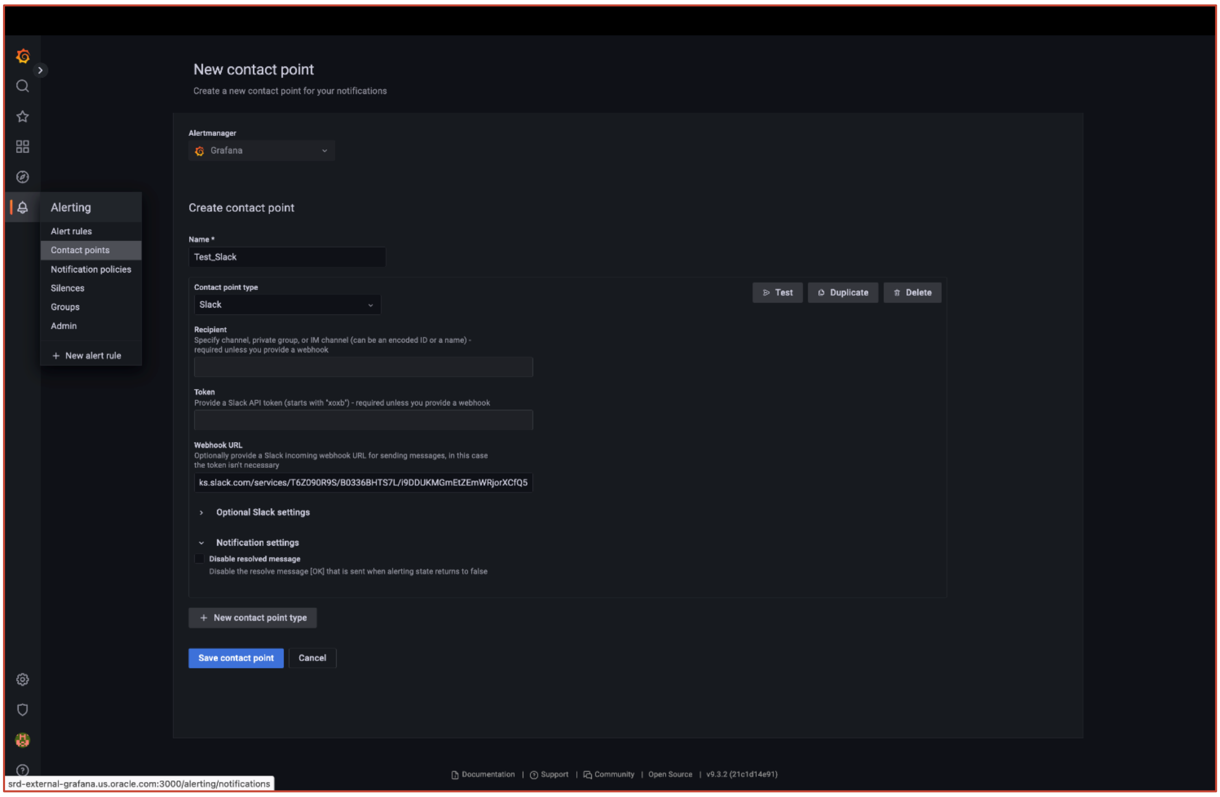
Grafana Server – Email Contact Point – Test Alert

The above email test alert was successfully received.

Slack (Webhook) HTTP Alerting

A new Contact Point needs to be created to permit HTTP based Webhook alerts to be sent.

In this case, a Slack Bot has been previously created to act as the recipient of the Grafana Server alerts. Please use the correct site-specific settings to enable this functionality. First, create a new Contact Point:



New contact point

Create a new contact point for your notifications

Alertmanager: Grafana

Create contact point

Name *: Test_Slack

Contact point type: Slack

Recipient
Specify channel, private group, or IM channel (can be an encoded ID or a name) - required unless you provide a webhook

Token
Provide a Slack API token (starts with "xoxb") - required unless you provide a webhook

Webhook URL
Optionally provide a Slack incoming webhook URL for sending messages, in this case the token isn't necessary

ks.slack.com/services/T6Z090R9S/B0336BHTS7L/i9DDUKMGmEtZEmWRjorXCfQ5

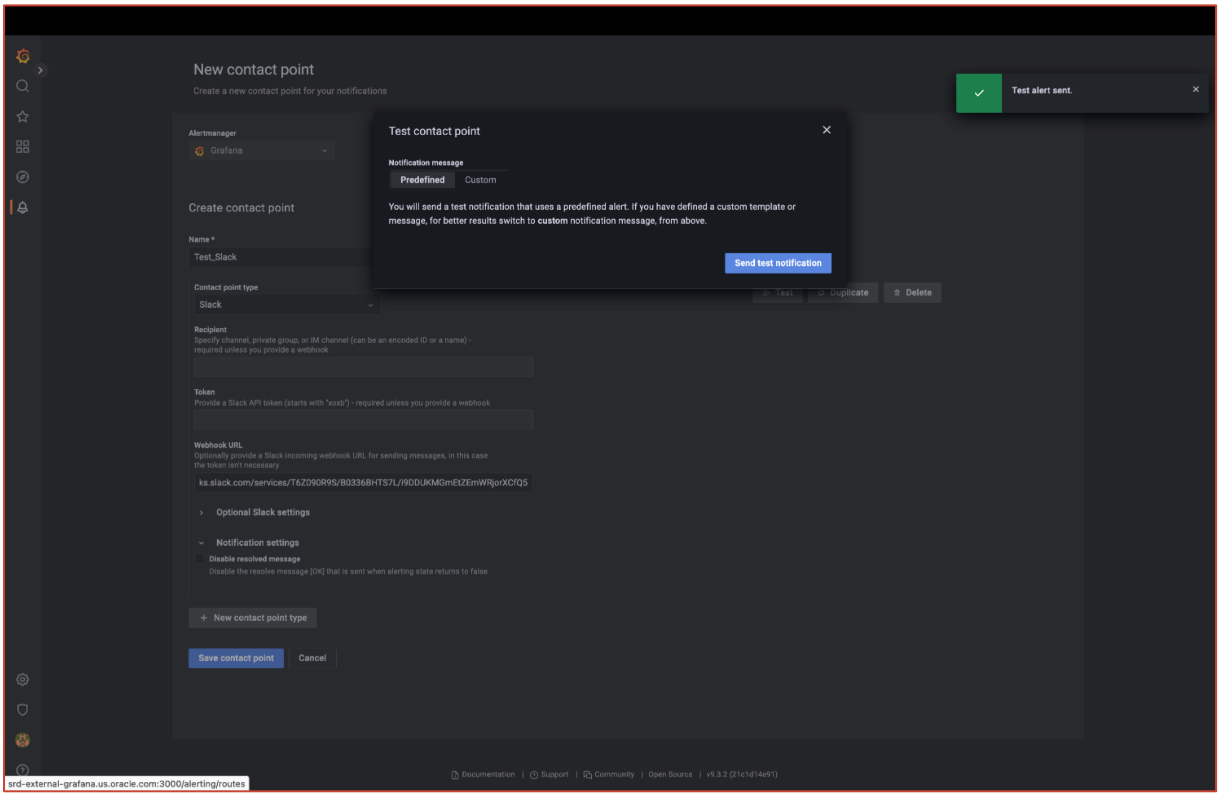
› Optional Slack settings

⌄ Notification settings

□ Disable resolved message
Disable the resolve message [OK] that is sent when alerting state returns to false

Grafana Server – Slack Contact Point – Create

Test fire an alert:



New contact point

Create a new contact point for your notifications

Alertmanager: Grafana

Create contact point

Name *: Test_Slack

Contact point type: Slack

Recipient
Specify channel, private group, or IM channel (can be an encoded ID or a name) - required unless you provide a webhook

Token
Provide a Slack API token (starts with "xoxb") - required unless you provide a webhook

Webhook URL
Optionally provide a Slack incoming webhook URL for sending messages, in this case the token isn't necessary

ks.slack.com/services/T6Z090R9S/B0336BHTS7L/i9DDUKMGmEtZEmWRjorXCfQ5

Optional Slack settings

Notification settings

□ Disable resolved message
Disable the resolve message [OK] that is sent when alerting state returns to false

Test contact point

Notification message: Predefined | Custom

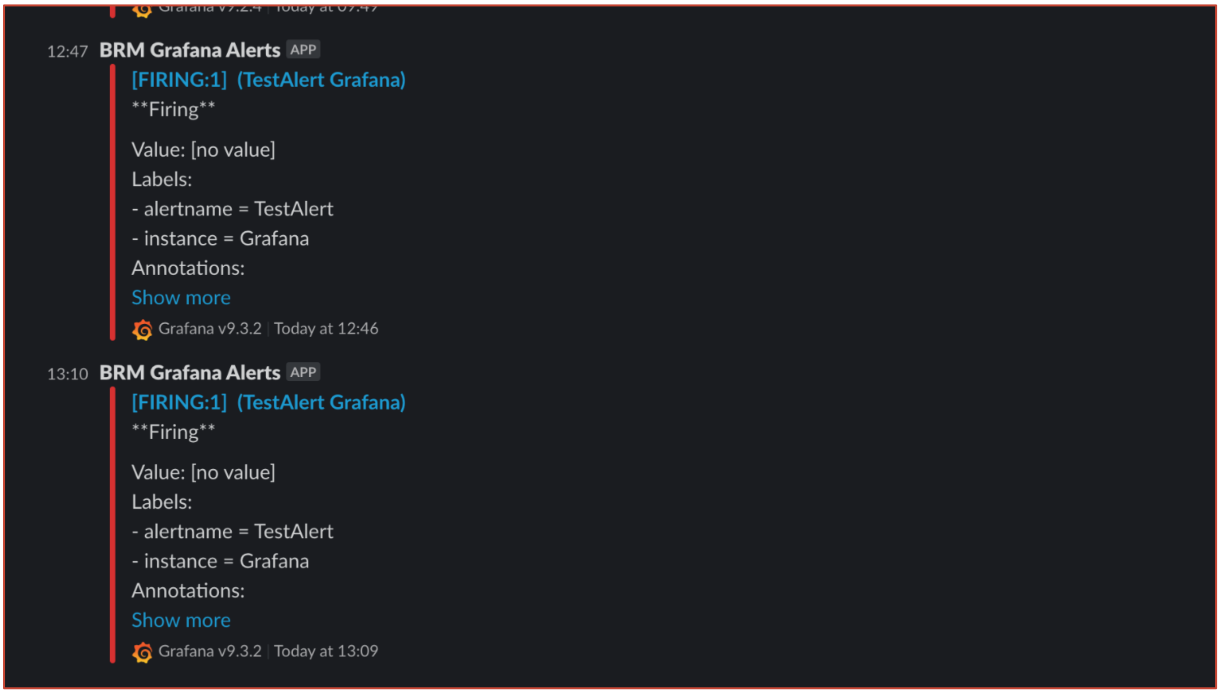
You will send a test notification that uses a predefined alert. If you have defined a custom template or message, for better results switch to **custom** notification message, from above.

✓ Test alert sent.

srd-external-grafana.us.oracle.com:3000/alerting/routes

Grafana Server – Slack Contact Point – Test Alert

Check in Slack for receipt of the test alert:



Grafana v9.2.4 | Today at 09:49

12:47 BRM Grafana Alerts APP

[FIRING:1] (TestAlert Grafana)

Firing

Value: [no value]
Labels:
- alertname = TestAlert
- instance = Grafana
Annotations:
Show more

Grafana v9.3.2 | Today at 12:46

13:10 BRM Grafana Alerts APP

[FIRING:1] (TestAlert Grafana)

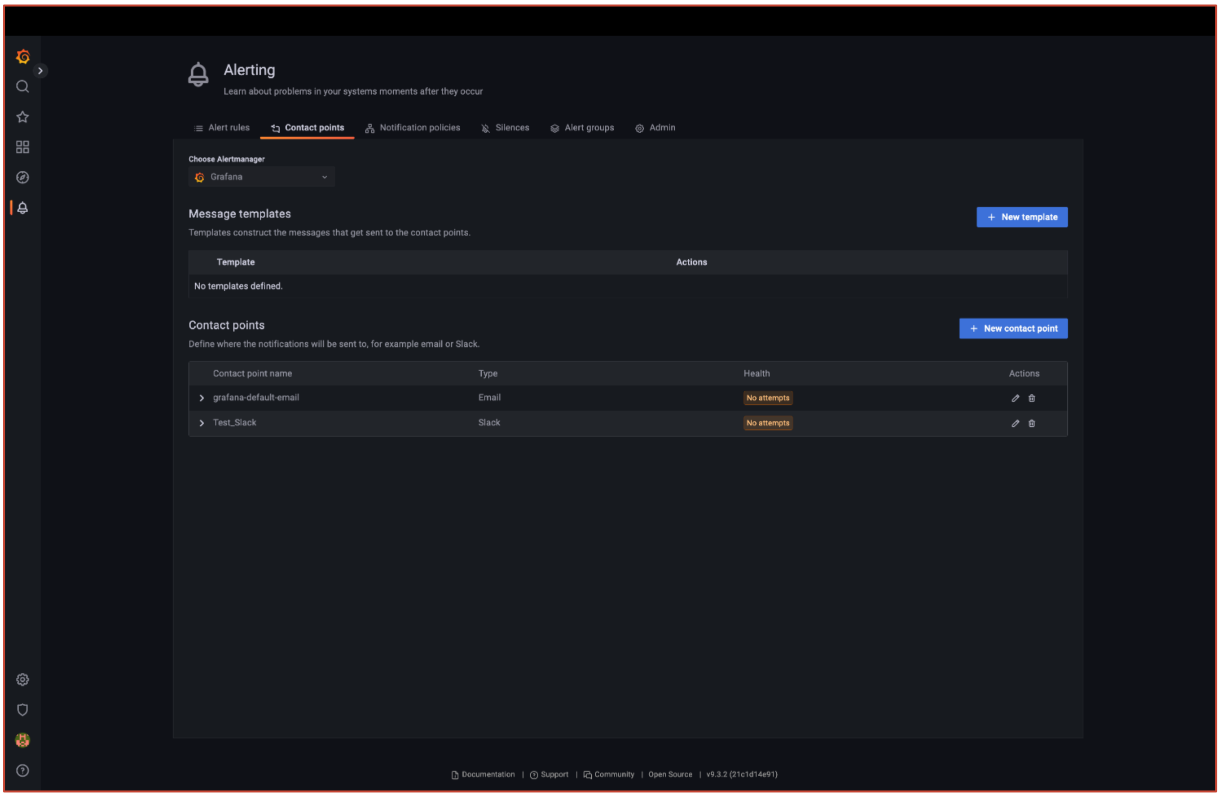
Firing

Value: [no value]
Labels:
- alertname = TestAlert
- instance = Grafana
Annotations:
Show more

Grafana v9.3.2 | Today at 13:09

Grafana Server – Slack Contact Point – Test Received

Display the update list of valid (and tested) Contact Points:



Alerting

Learn about problems in your systems moments after they occur

Choose Alertmanager: Grafana

Message templates

Templates construct the messages that get sent to the contact points.

Template	Actions
No templates defined.	

Contact points

Define where the notifications will be sent to, for example email or Slack.

Contact point name	Type	Health	Actions
grafana-default-email	Email	No attempts	
Test_Slack	Slack	No attempts	

Grafana Server – Available Contact Points

This completes this section of the step-by-step guide.

Section References

The following URL's provide links to additional documentation:

- Grafana Setup – https://grafana.com/docs/grafana/latest/setup-grafana/
- Grafana Administration – https://grafana.com/docs/grafana/latest/administration/
- Grafana Security – https://grafana.com/docs/grafana/latest/setup-grafana/configure-security/
- Grafana Alerting – https://grafana.com/docs/grafana/latest/alerting/

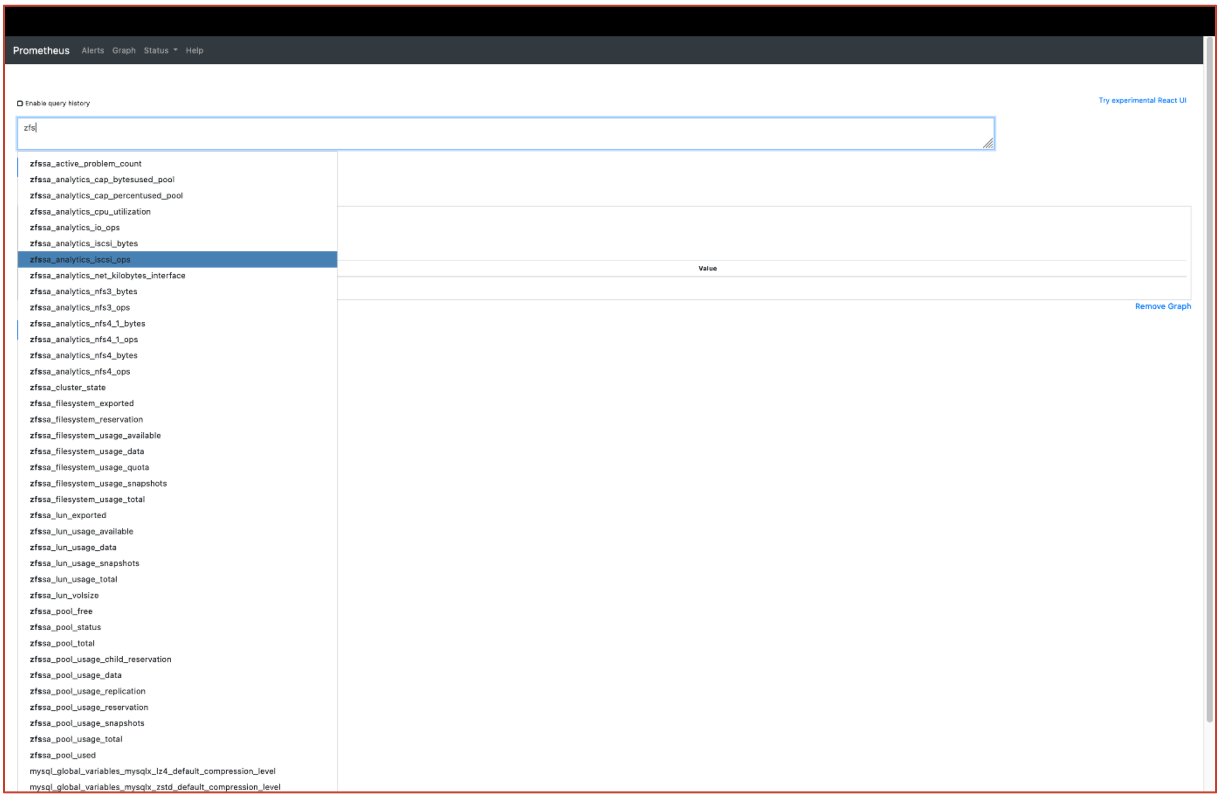
Adding Oracle Private Cloud Appliance X9-2 Systems to an External Grafana Server Service

This section provides a step-by-step guide to adding multiple Oracle Private Cloud Appliance X9-2 systems to an external Grafana Server service.

Oracle Private Cloud Appliance X9-2 Prometheus Service

In the same manner as the internal PCA Grafana Server service is directly addressable using the URL `'grafana.<FQDN>'` for each PCA system, each Oracle Private Cloud Appliance X9-2 system's internal Prometheus service can be externally addressed using the URL `'prometheus.<FQDN>'`.

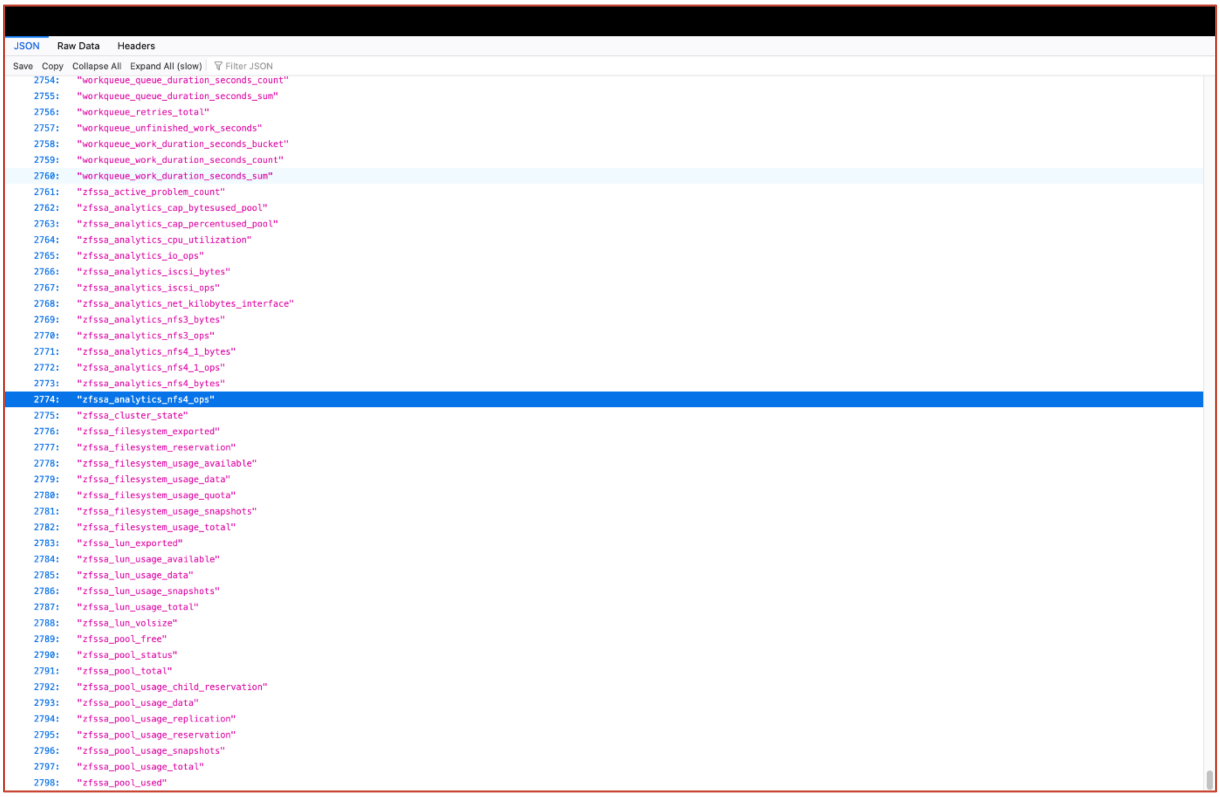
The URL `'prometheus.<FQDN>/graph'` will display the Prometheus graphing capability:



```
zfs
zfssa_active_problem_count
zfssa_analytics_cap_bytesused_pool
zfssa_analytics_cap_percentused_pool
zfssa_analytics_cpu_utilization
zfssa_analytics_io_ops
zfssa_analytics_iscsi_bytes
zfssa_analytics_iscsi_ops
zfssa_analytics_net_kilobytes_interface
zfssa_analytics_nfs3_bytes
zfssa_analytics_nfs3_ops
zfssa_analytics_nfs4_1_bytes
zfssa_analytics_nfs4_1_ops
zfssa_analytics_nfs4_bytes
zfssa_analytics_nfs4_ops
zfssa_cluster_state
zfssa_filesystem_exported
zfssa_filesystem_reservation
zfssa_filesystem_usage_available
zfssa_filesystem_usage_data
zfssa_filesystem_usage_quota
zfssa_filesystem_usage_snapshots
zfssa_filesystem_usage_total
zfssa_lun_exported
zfssa_lun_usage_available
zfssa_lun_usage_data
zfssa_lun_usage_snapshots
zfssa_lun_usage_total
zfssa_lun_volsize
zfssa_pool_free
zfssa_pool_status
zfssa_pool_total
zfssa_pool_usage_child_reservation
zfssa_pool_usage_data
zfssa_pool_usage_replication
zfssa_pool_usage_reservation
zfssa_pool_usage_snapshots
zfssa_pool_usage_total
zfssa_pool_used
mysql_global_variables_mysqlx_lz4_default_compression_level
mysql_global_variables_mysqlx_zstd_default_compression_level
```

External Grafana – Oracle Private Cloud Appliance X9-2 Prometheus service – Graphing

And the URL `'prometheus.<FQDN>/api/v1/label/__name__/values'` will display a list of the available metrics being collected internally within the Oracle Private Cloud Appliance X9-2 by the Prometheus service.



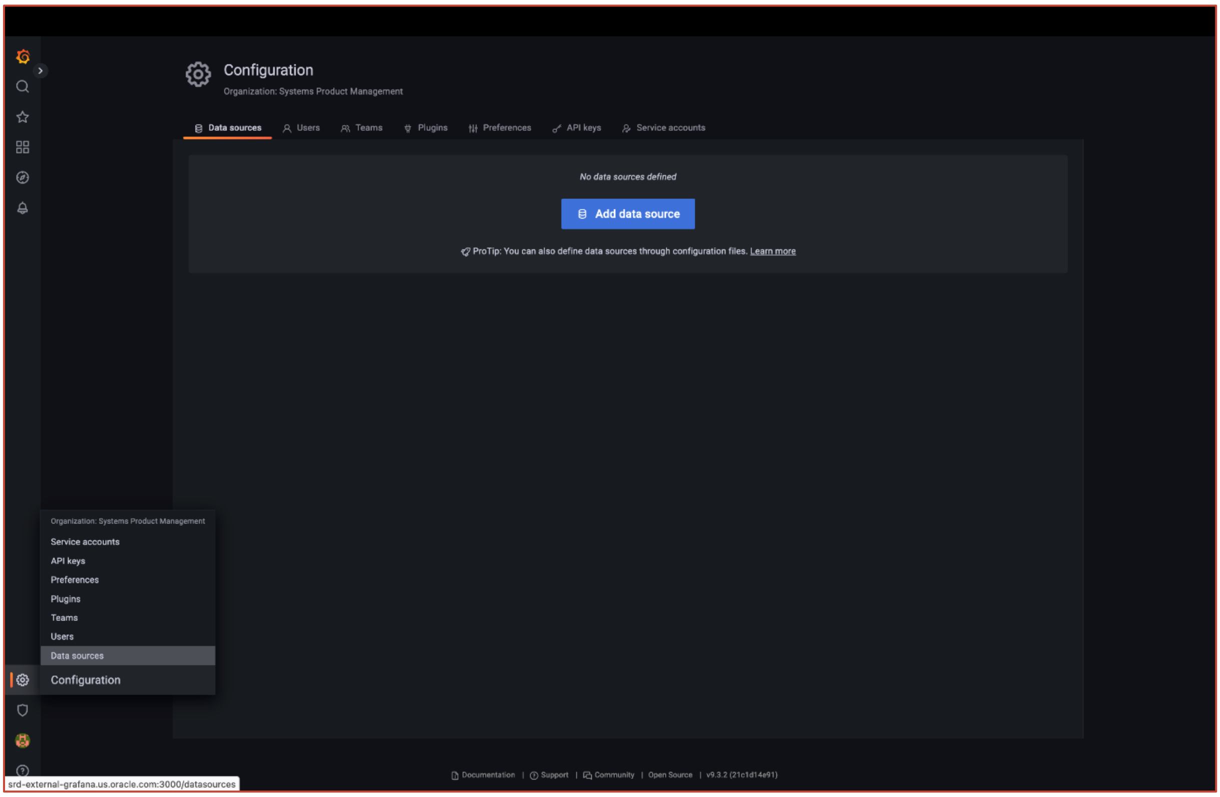
```
2754:  "workqueue_queue_duration_seconds_count"
2755:  "workqueue_queue_duration_seconds_sum"
2756:  "workqueue_retries_total"
2757:  "workqueue_unfinished_work_seconds"
2758:  "workqueue_work_duration_seconds_bucket"
2759:  "workqueue_work_duration_seconds_count"
2760:  "workqueue_work_duration_seconds_sum"
2761:  "zfssa_active_problem_count"
2762:  "zfssa_analytics_cap_bytesused_pool"
2763:  "zfssa_analytics_cap_percentused_pool"
2764:  "zfssa_analytics_cpu_utilization"
2765:  "zfssa_analytics_io_ops"
2766:  "zfssa_analytics_iscsi_bytes"
2767:  "zfssa_analytics_iscsi_ops"
2768:  "zfssa_analytics_net_kilobytes_interface"
2769:  "zfssa_analytics_nfs3_bytes"
2770:  "zfssa_analytics_nfs3_ops"
2771:  "zfssa_analytics_nfs4_1_bytes"
2772:  "zfssa_analytics_nfs4_1_ops"
2773:  "zfssa_analytics_nfs4_bytes"
2774:  "zfssa_analytics_nfs4_ops"
2775:  "zfssa_cluster_state"
2776:  "zfssa_filesystem_exported"
2777:  "zfssa_filesystem_reservation"
2778:  "zfssa_filesystem_usage_available"
2779:  "zfssa_filesystem_usage_data"
2780:  "zfssa_filesystem_usage_quota"
2781:  "zfssa_filesystem_usage_snapshots"
2782:  "zfssa_filesystem_usage_total"
2783:  "zfssa_lun_exported"
2784:  "zfssa_lun_usage_available"
2785:  "zfssa_lun_usage_data"
2786:  "zfssa_lun_usage_snapshots"
2787:  "zfssa_lun_usage_total"
2788:  "zfssa_lun_volsize"
2789:  "zfssa_pool_free"
2790:  "zfssa_pool_status"
2791:  "zfssa_pool_total"
2792:  "zfssa_pool_usage_child_reservation"
2793:  "zfssa_pool_usage_data"
2794:  "zfssa_pool_usage_replication"
2795:  "zfssa_pool_usage_reservation"
2796:  "zfssa_pool_usage_snapshots"
2797:  "zfssa_pool_usage_total"
2798:  "zfssa_pool_used"
```

External Grafana – Oracle Private Cloud Appliance X9-2 Prometheus service - Metrics

Almost 2,800 Prometheus metrics are available from each Oracle Private Cloud Appliance X9-2 system.

Adding Grafana Data Sources

A new Grafana data source can be created using the externally exposed Prometheus service from within each Oracle Private Cloud Appliance X9-2 system. By default, the new Grafana Server instance will have NO configured data sources:



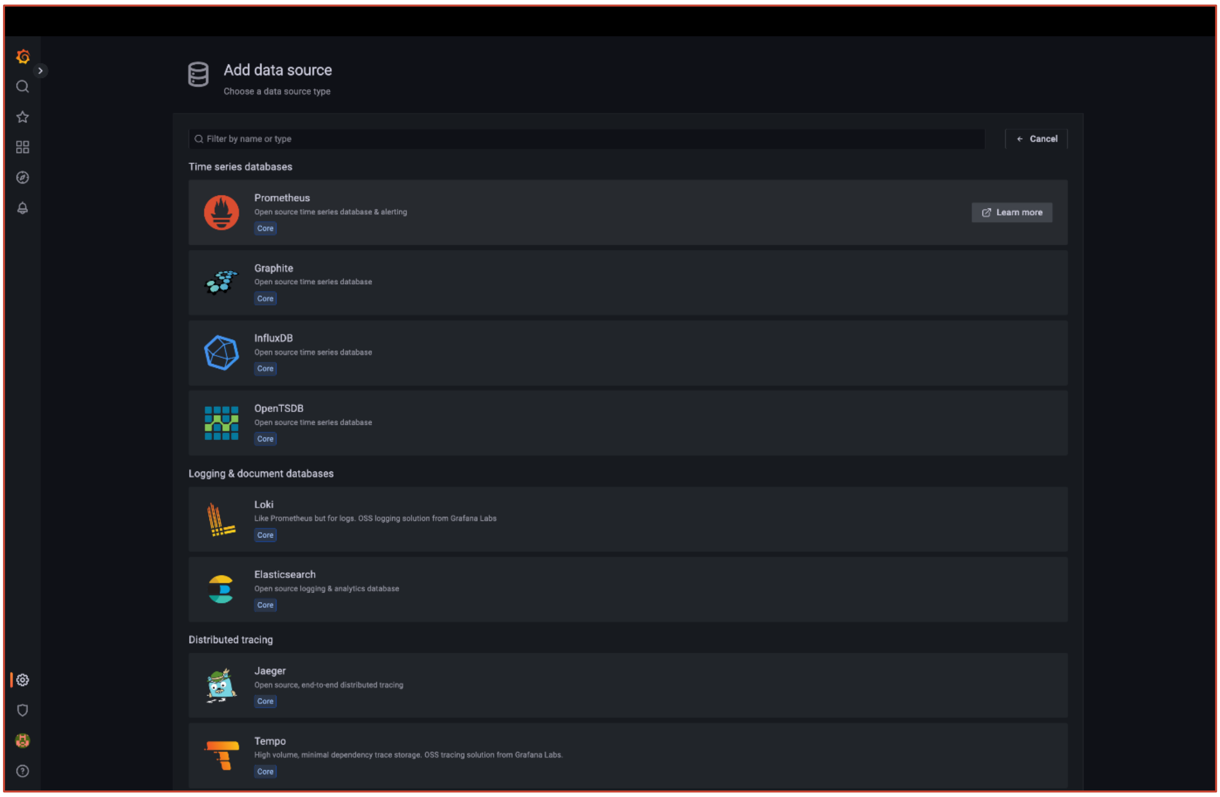
No data sources defined

ProTip: You can also define data sources through configuration files. Learn more

srd-external-grafana.us.oracle.com:3000/datasources

External Grafana – Empty Data source

So, let's add the first Oracle Private Cloud Appliance X9-2 Prometheus data source service. Create a new Grafana data source using the Prometheus template:



Add data source

Choose a data source type

Filter by name or type

Time series databases

- Prometheus — Open source time series database & alerting — Core
- Graphite — Open source time series database — Core
- InfluxDB — Open source time series database — Core
- OpenTSDB — Open source time series database — Core

Logging & document databases

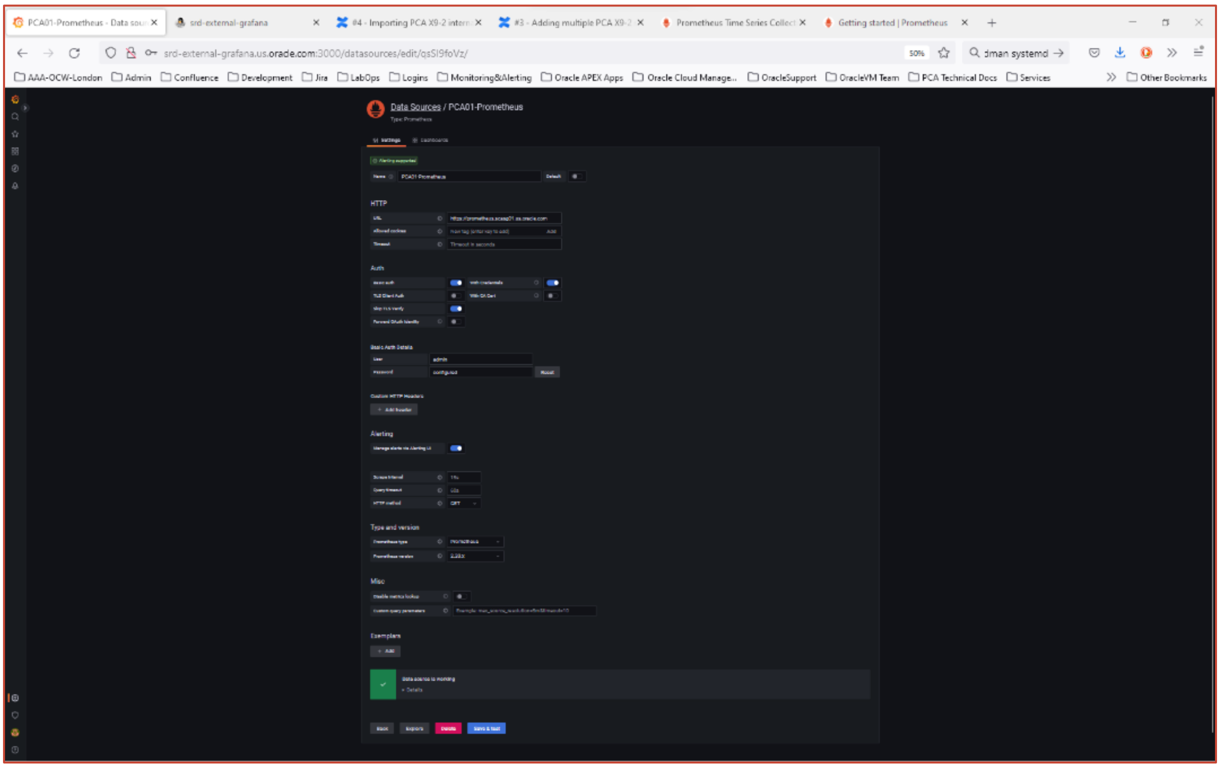
- Loki — Like Prometheus but for logs. OSS logging solution from Grafana Labs — Core
- Elasticsearch — Open source logging & analytics database — Core

Distributed tracing

- Jaeger — Open source, end-to-end distributed tracing — Core
- Tempo — High volume, minimal dependency trace storage. OSS tracing solution from Grafana Labs. — Core

External Grafana – Add Prometheus Data source

Then configure for the first PCA (PCA01) system:



srd-external-grafana.us.oracle.com:3000/datasources/edit/qsSI9foVz/

Data Sources / PCA01-Prometheus

Type: Prometheus

Settings | Dashboards

Alerting supported

Name: PCA01-Prometheus

HTTP

URL: https://prometheus.scasg01.us.oracle.com

Auth

Basic auth: on | With credentials: on

TLS Client Auth: off | With CA Cert: off

Skip TLS Verify: on

Forward OAuth Identity: off

Basic Auth Details

User: admin

Password: configured

Alerting

Manage alerts via Alerting UI: on

Scrape interval: 15s

Query timeout: 60s

HTTP method: GET

Type and version

Prometheus type: Prometheus

Prometheus version: 2.28.x

Misc

Disable metrics lookup: off

Exemplars

Data source is working

External Grafana – Create first PCA Data source

For simplicities' sake, the following options have been used:

- Basic Authentication
- With Credentials
- Skip TLS Verify
- Basic Authentication Details (username & password)

Obviously, a more secure connection model may be required.

In which case, follow the Grafana specific documentation to enable TLS Client Auth and/or CA Certificate exchange. (https://grafana.com/docs/grafana/latest/datasources/prometheus/)

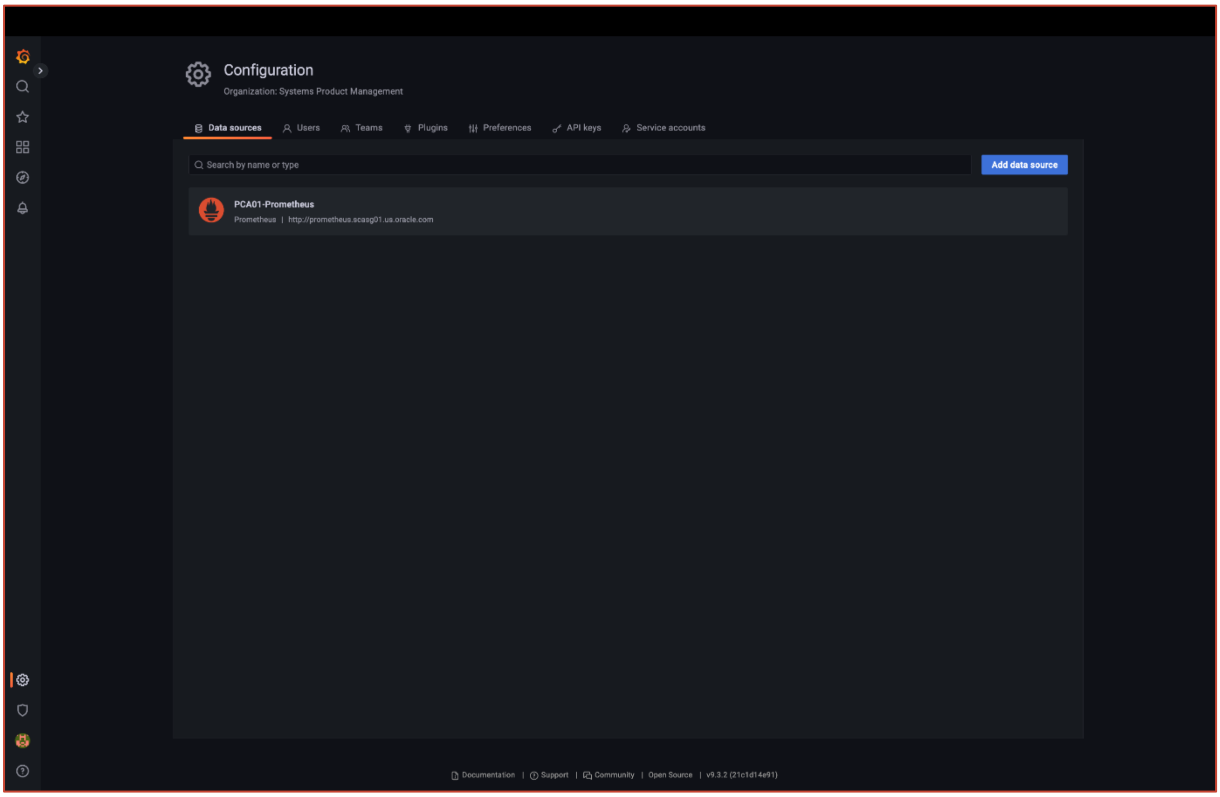
There are further mandatory option settings required to for a working data source:

- HTTP Method should be set to 'GET'
- Data source Type should be set to 'Prometheus'
- Prometheus Version should be set to '2.20.x'

Save and test the new data source to make sure it is working.

The test connection has worked, and the data source is registered as valid (working).

This new data source can now be seen in the Configuration → Data Sources screen within the external Grafana Server instance:



Configuration

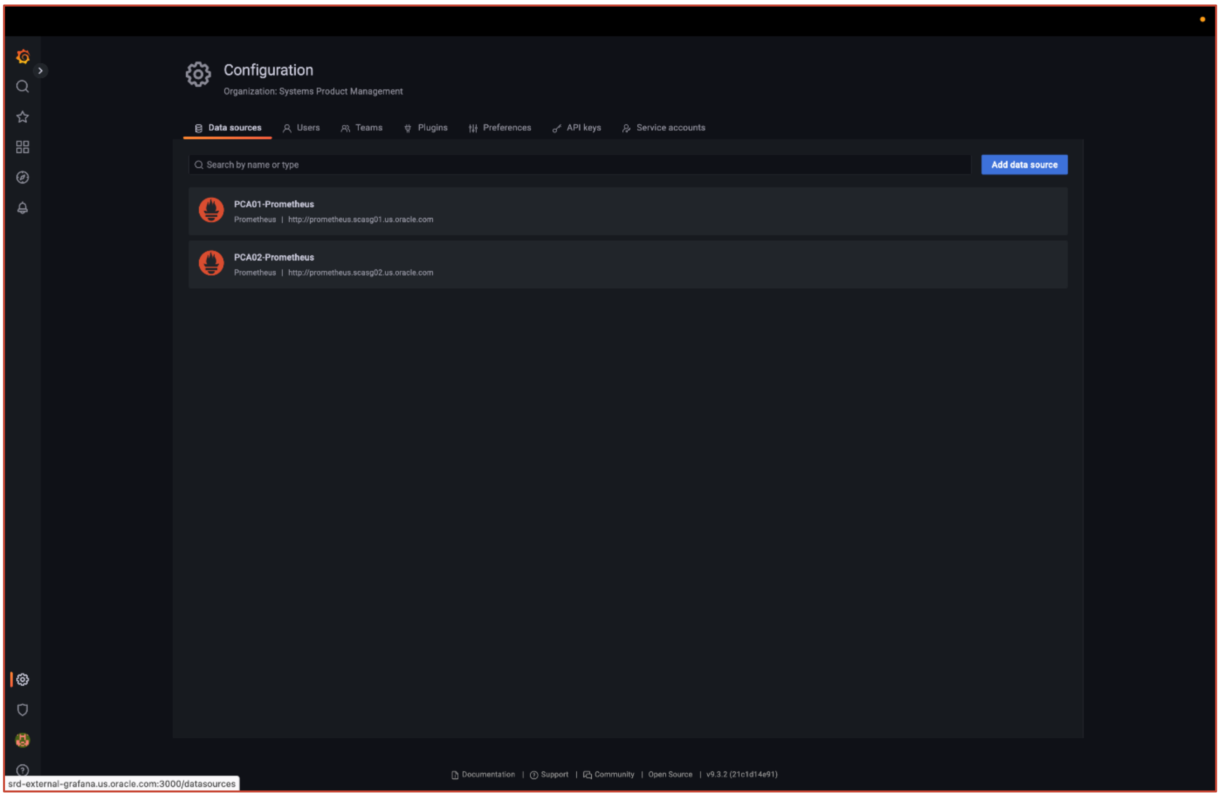
Organization: Systems Product Management

PCA01-Prometheus

Prometheus | http://prometheus.scasg01.us.oracle.com

External Grafana – Validated PCA data source

Now repeat to add further Oracle Private Cloud Appliance X9-2 systems as additional data sources.


Configuration
Organization: Systems Product Management
Data sources
Users
Teams
Plugins
Preferences
API keys
Service accounts
Search by name or type
Add data source
Documentation | Support | Community | Open Source | v9.3.2 (21c1d14e91)
srd-external-grafana.us.oracle.com:3000/datasources

PCA01-Prometheus
Prometheus | http://prometheus.scasg01.us.oracle.com

PCA02-Prometheus
Prometheus | http://prometheus.scasg02.us.oracle.com

External Grafana – Multiple PCA Data sources

As can be seen in the screenshot above, there are now two working Prometheus data sources defined and working:

- PCA01-Prometheus - (http://prometheus.scasg01.us.oracle.com)
- PCA02-Prometheus - (http://prometheus.scasg02.us.oracle.com)

This completes this section on adding multiple Grafana data sources to an external Grafana Server service.

Section References

As always, the definitive source for information and instruction for configuring Grafana Server remains the Grafana documentation itself:

- Main Grafana Document Library – https://grafana.com/docs/grafana/latest/
- Grafana data source documentation – https://grafana.com/docs/grafana/latest/datasources/

Importing Oracle Private Cloud Appliance X9-2 Internal Grafana Dashboards

This section provides a step-by-step guide for importing existing Grafana Dashboards from a single,existing Oracle Private Cloud Appliance X9-2 to be used to monitor all the systems via the external Grafana server service.

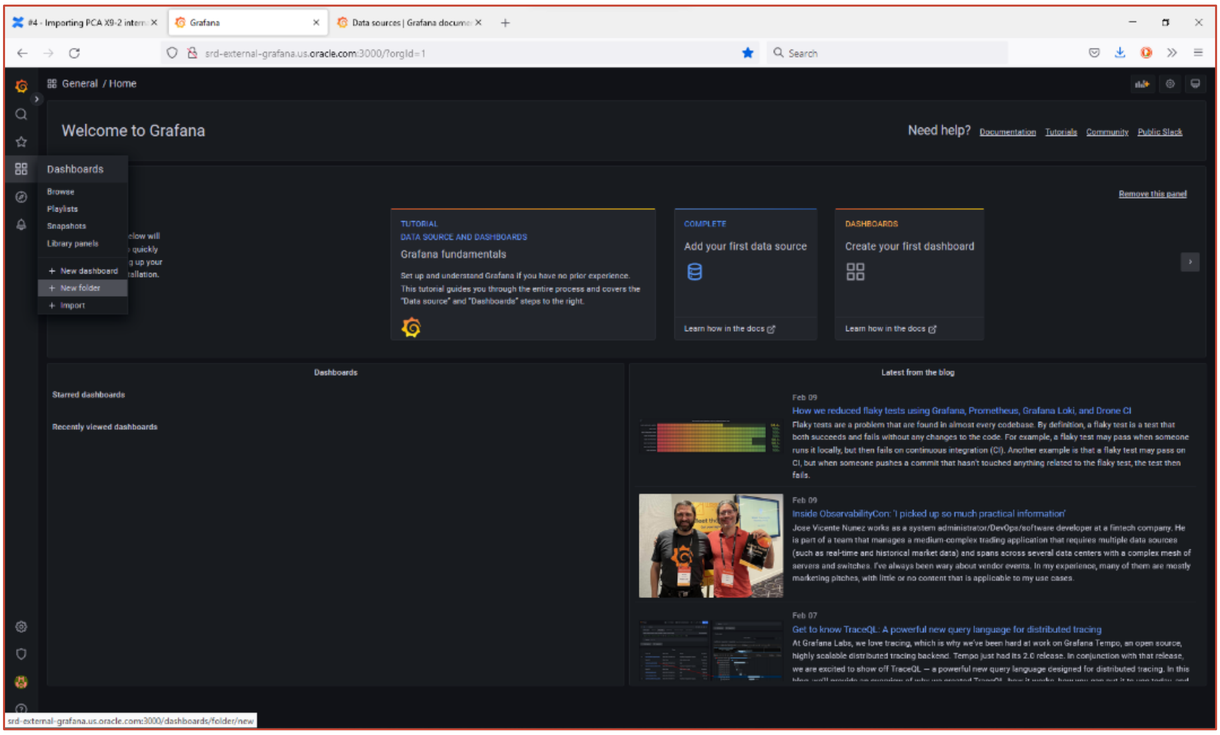
Create Oracle Private Cloud Appliance Specific Folders

Initially, we will import one existing PCA Grafana dashboard from a single Oracle Private Cloud Appliance X9-2 to use for all PCA's being monitored through this external Grafana Server.

Since each imported Grafana dashboard will only be able to have a single Grafana data source, any dashboards will need to be imported multiple times and create PCA-specific versions of each dashboard.

To separate these imported dashboards, a PCA-specific folder will be created to hold each imported dashboard.

From the external Grafana Server home page dashboard icon, select add new folder:



Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

How we reduced flaky tests using Grafana, Prometheus, Grafana Loki, and Drone CI

Flaky tests are a problem that are found in almost every codebase. By definition, a flaky test is a test that both succeeds and fails without any changes to the code. For example, a flaky test may pass when someone runs it locally, but then fails on continuous integration (CI). Another example is that a flaky test may pass on CI, but when someone pushes a commit that hasn't touched anything related to the flaky test, the test then fails.

Inside ObservabilityCon: 'I picked up so much practical information'

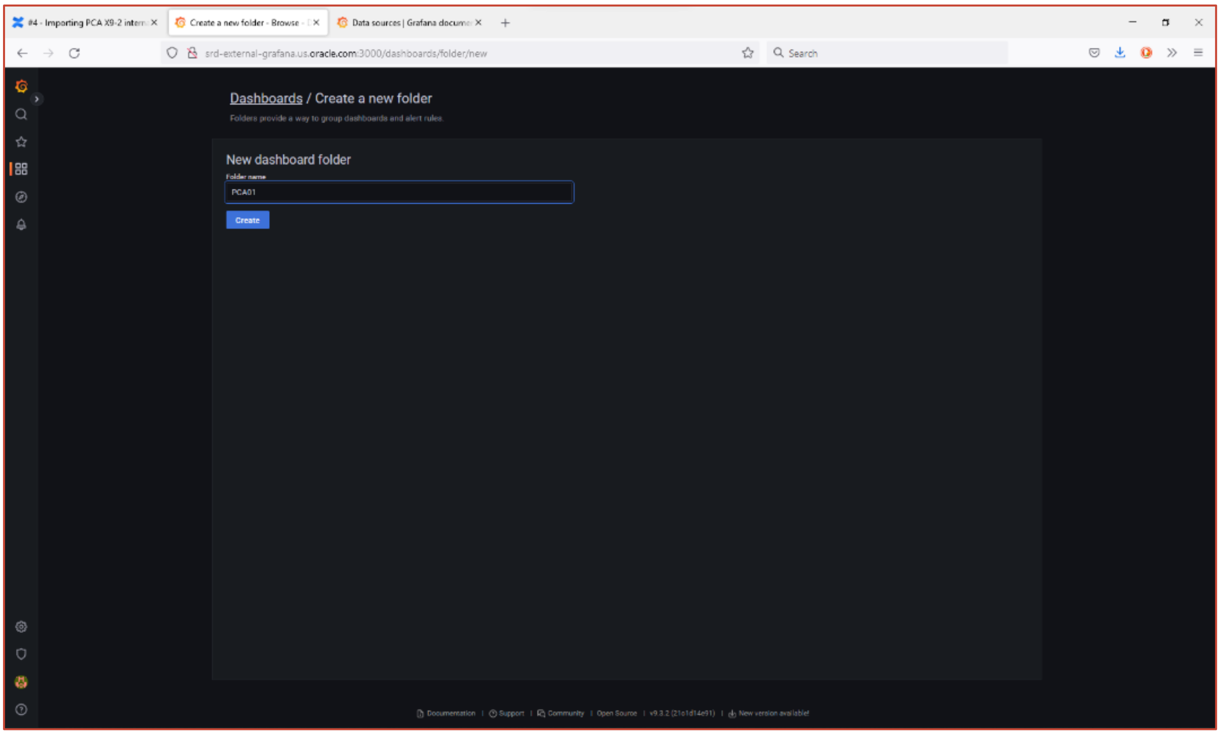
Jose Vicente Nunez works as a system administrator/DevOps/software developer at a fintech company. He is part of a team that manages a medium-complex trading application that requires multiple data sources (such as real-time and historical market data) and spans across several data centers with a complex mesh of servers and switches. I've always been wary about vendor events. In my experience, many of them are mostly marketing pitches, with little or no content that is applicable to my use cases.

Get to know TraceQL: A powerful new query language for distributed tracing

At Grafana Labs, we love tracing, which is why we've been hard at work on Grafana Tempo, an open source, highly scalable distributed tracing backend. Tempo just had its 2.0 release. In conjunction with that release, we are excited to show off TraceQL — a powerful new query language designed for distributed tracing. In this

Import Dashboards – Create Folder

This will display the following screen:


srd-external-grafana.us.oracle.com:3000/dashboards/folder/new

Dashboards / Create a new folder

Folders provide a way to group dashboards and alert rules.

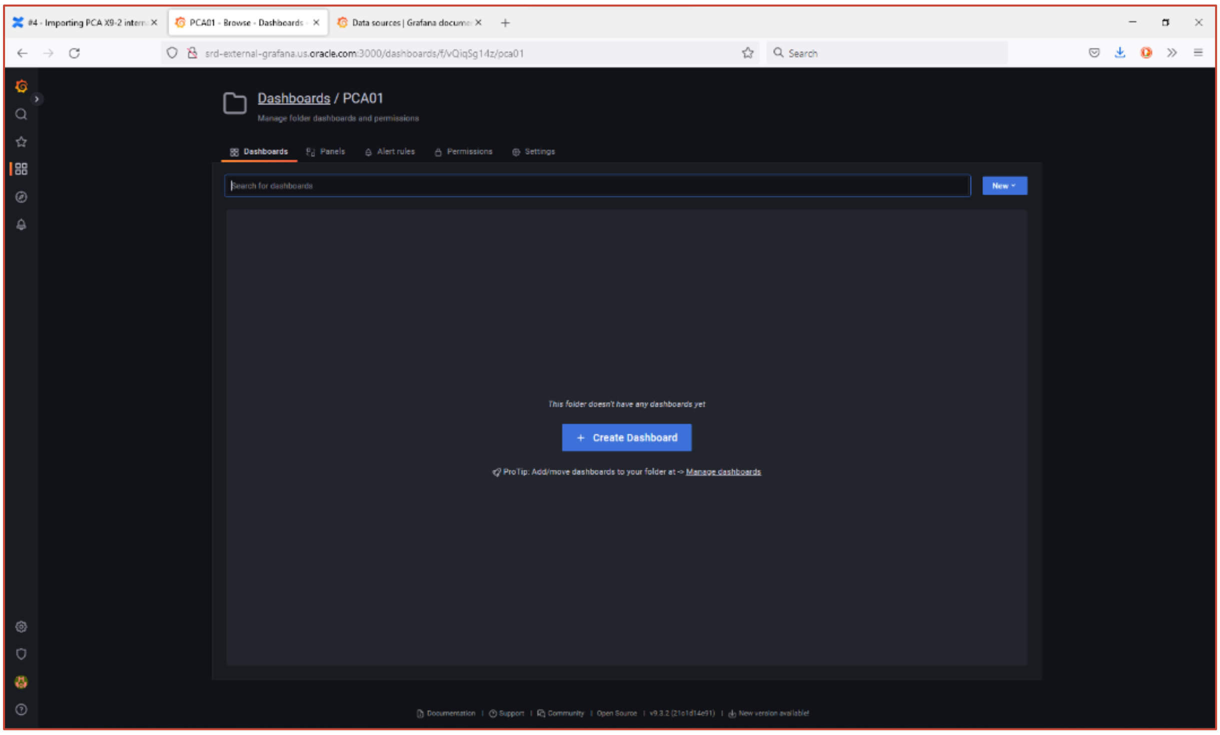
New dashboard folder

Folder name: PCA01

#4 - Importing PCA X9-2 intern
Create a new folder - Browse - D
Data sources | Grafana docume
Search
Create
Documentation | Support | Community | Open Source | v9.3.2 (21o1d14e91) | New version available!

Import Dashboards – New Folder

Create a folder for the first PCA, in this case, PCA01. Upon creation the empty folder is displayed:


srd-external-grafana.us.oracle.com:3000/dashboards/f/vQiqSg14z/pca01

Dashboards / PCA01

Manage folder dashboards and permissions

This folder doesn't have any dashboards yet

ProTip: Add/move dashboards to your folder at -> Manage dashboards

#4 - Importing PCA X9-2 intern
PCA01 - Browse - Dashboards
Data sources | Grafana docume
Search
Dashboards
Panels
Alert rules
Permissions
Settings
Search for dashboards
New
+ Create Dashboard
Documentation | Support | Community | Open Source | v9.3.2 (21o1d14e91) | New version available!

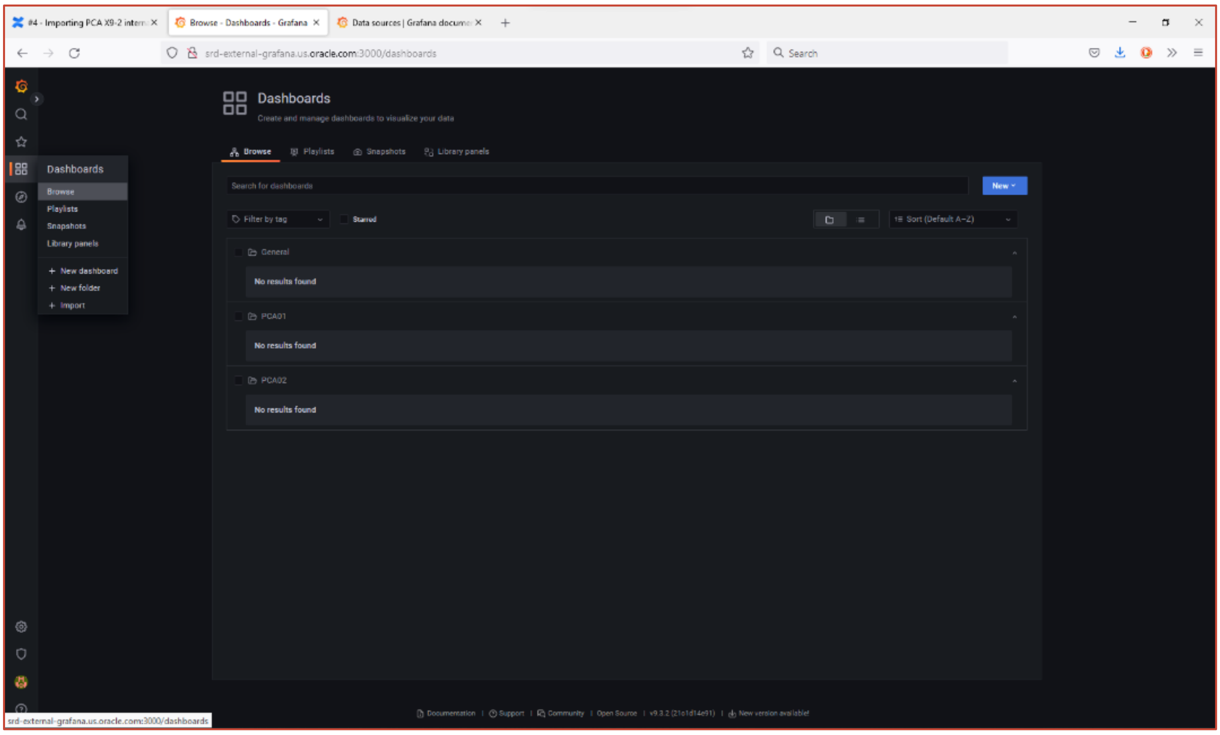
Import Dashboards – PCA01 Folder

Now repeat for the additional Oracle Private Cloud Appliance X9-2 systems for which Prometheus data sources have been created in the previous section.

Once complete, browsing the Grafana dashboards will show the following empty folders:

- General - Default Folder
- PCA01 - First PCA Dashboard Folder

- PCA02 - Second PCA Dashboard Folder



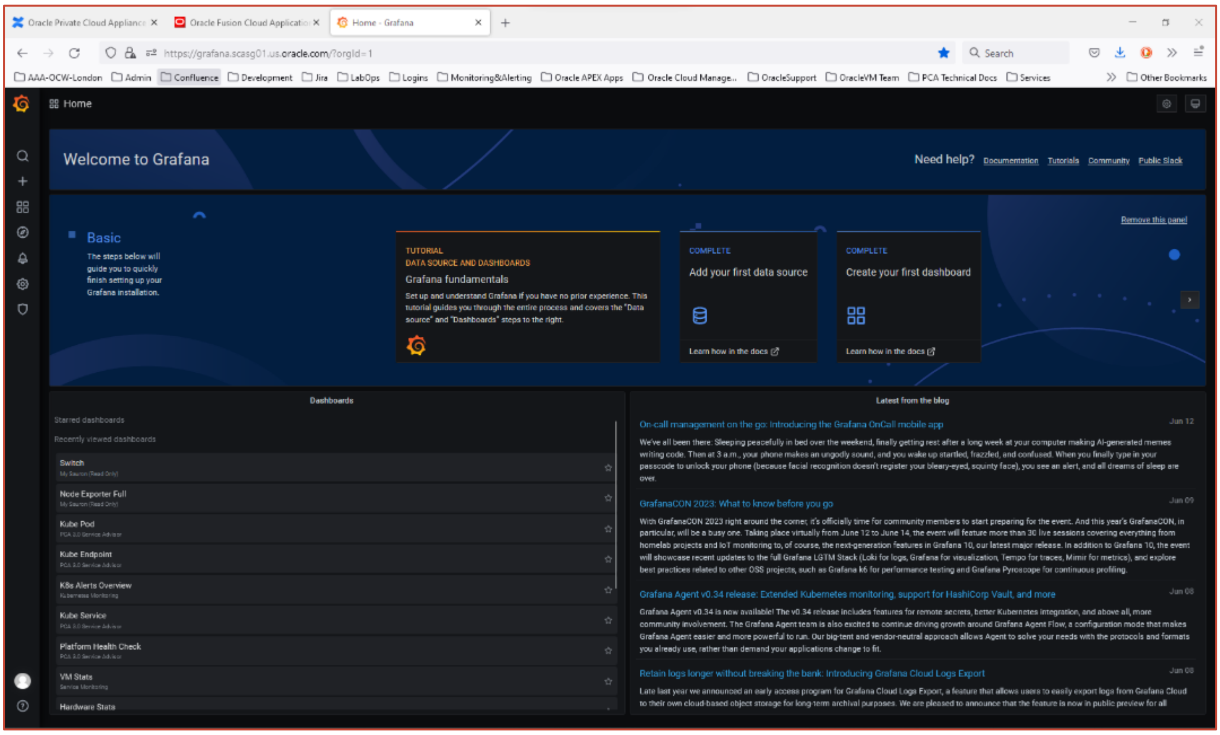
Import Dashboards – Completed Folders

Now the external Grafana Server is ready for Grafana dashboards to be imported or created.

Export Oracle Private Cloud Appliance X9-2 Grafana Dashboards from a Oracle Private Cloud Appliance X9-2

For the purposes of this paper, one of the standard Oracle Private Cloud Appliance X9-2 Grafana dashboards will be displayed and then exported from the first PCA system defined.

Connecting to the *INTERNAL* Grafana service provided by PCA01 displays the standard Grafana home page:


https://grafana.scasg01.us.oracle.com/?orgId=1
Home
Welcome to Grafana
Need help? Documentation Tutorials Community Public Slack
Remove this panel
Basic
TUTORIAL
DATA SOURCE AND DASHBOARDS
Grafana fundamentals
COMPLETE
Add your first data source
Learn how in the docs
COMPLETE
Create your first dashboard
Learn how in the docs
Dashboards
Starred dashboards
Recently viewed dashboards
Latest from the blog

The steps below will guide you to quickly finish setting up your Grafana installation.

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

Switch – My Sauron (Read Only)
Node Exporter Full – My Sauron (Read Only)
Kube Pod – PCA 3.0 Service Advisor
Kube Endpoint – PCA 3.0 Service Advisor
K8s Alerts Overview – Kubernetes Monitoring
Kube Service – PCA 3.0 Service Advisor
Platform Health Check – PCA 3.0 Service Advisor
VM Stats – Service Monitoring
Hardware Stats

On-call management on the go: Introducing the Grafana OnCall mobile app — Jun 12
We've all been there: Sleeping peacefully in bed over the weekend, finally getting rest after a long week at your computer making AI-generated memes writing code. Then at 3 a.m., your phone makes an ungodly sound, and you wake up startled, frazzled, and confused. When you finally type in your passcode to unlock your phone (because facial recognition doesn't register your bleary-eyed, squinty face), you see an alert, and all dreams of sleep are over.

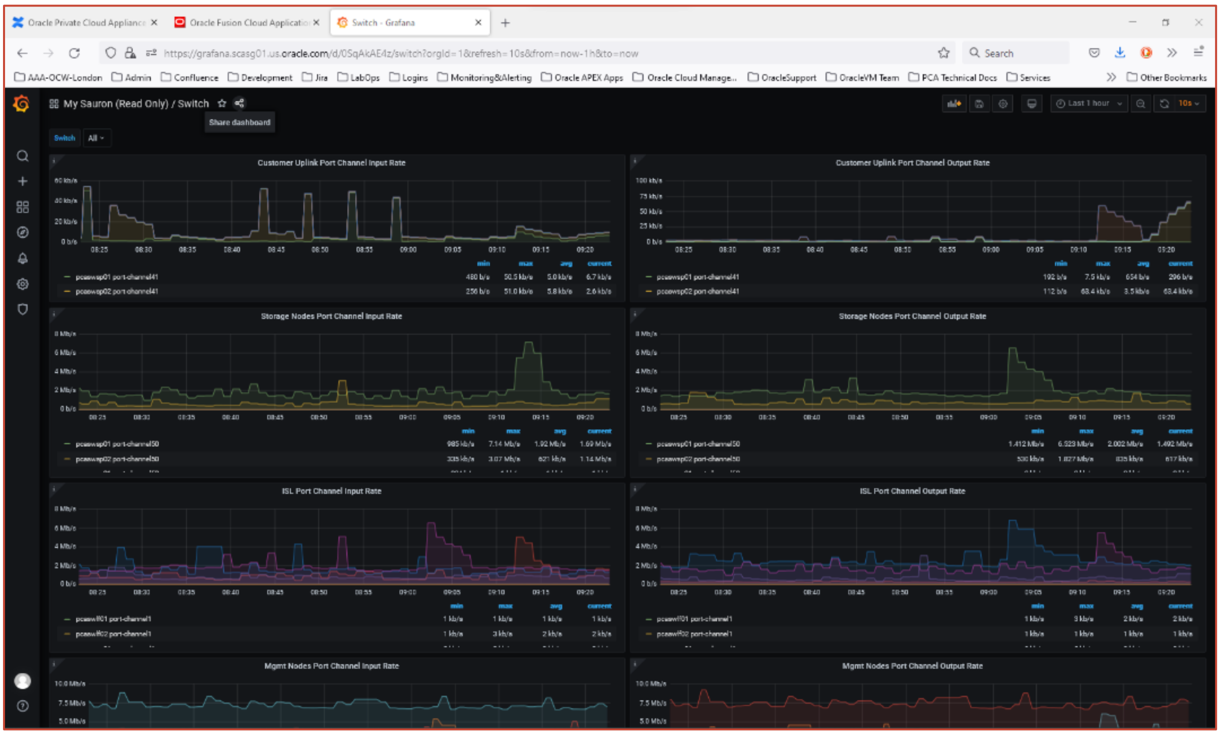
GrafanaCON 2023: What to know before you go — Jun 09
With GrafanaCON 2023 right around the corner, it's officially time for community members to start preparing for the event. And this year's GrafanaCON, in particular, will be a busy one. Taking place virtually from June 12 to June 14, the event will feature more than 30 live sessions covering everything from homelab projects and IoT monitoring to, of course, the next-generation features in Grafana 10, our latest major release. In addition to Grafana 10, the event will showcase recent updates to the full Grafana LGTM Stack (Loki for logs, Grafana for visualization, Tempo for traces, Mimir for metrics), and explore best practices related to other OSS projects, such as Grafana k6 for performance testing and Grafana Pyroscope for continuous profiling.

Grafana Agent v0.34 release: Extended Kubernetes monitoring, support for HashiCorp Vault, and more — Jun 08
Grafana Agent v0.34 is now available! The v0.34 release includes features for remote secrets, better Kubernetes integration, and above all, more community involvement. The Grafana Agent team is also excited to continue driving growth around Grafana Agent Flow, a configuration mode that makes Grafana Agent easier and more powerful to run. Our big-tent and vendor-neutral approach allows Agent to solve your needs with the protocols and formats you already use, rather than demand your applications change to fit.

Retain logs longer without breaking the bank: Introducing Grafana Cloud Logs Export — Jun 08
Late last year we announced an early access program for Grafana Cloud Logs Export, a feature that allows users to easily export logs from Grafana Cloud to their own cloud-based object storage for long-term archival purposes. We are pleased to announce that the feature is now in public preview for all

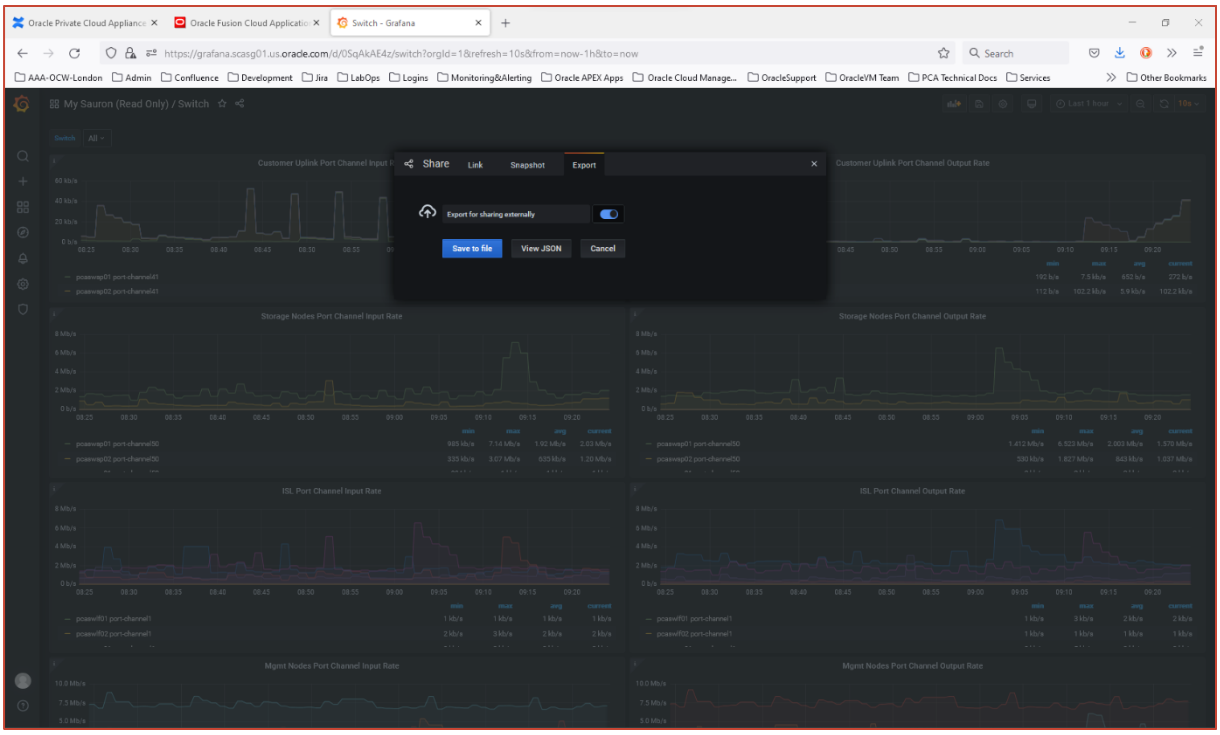
Import Dashboards – Export from Oracle Private Cloud Appliance X9-2

The Switch Grafana dashboard provided as part of the base PCA3 build will be selected. Click on the 'Switch' entry in the recently viewed dashboards list:


https://grafana.scasg01.us.oracle.com/d/0SqAkAE4z/switch?orgId=1&refresh=10s&from=now-1h&to=now
My Sauron (Read Only) / Switch
Share dashboard
Switch
All
Last 1 hour
10s
Customer Uplink Port Channel Input Rate
Customer Uplink Port Channel Output Rate
Storage Nodes Port Channel Input Rate
Storage Nodes Port Channel Output Rate
ISL Port Channel Input Rate
ISL Port Channel Output Rate
Mgmt Nodes Port Channel Input Rate
Mgmt Nodes Port Channel Output Rate

Import Dashboards – Oracle Private Cloud Appliance X9-2 Switch Dashboard

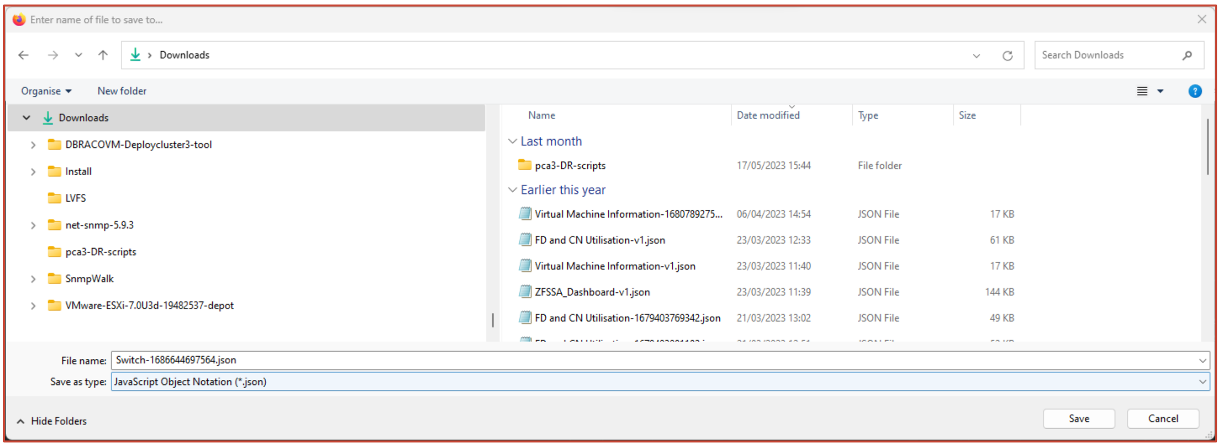
Alongside the dashboard Title (top left of the screen) is the share link (as highlighted above). Click through on this and the share windows pops up:


https://grafana.scasg01.us.oracle.com/d/0SqAkAE4z/switch?orgId=1&refresh=10s&from=now-1h&to=now
My Sauron (Read Only) / Switch
Share
Link
Snapshot
Export
Save to file
View JSON
Cancel

Export for sharing externally: on

Import Dashboards – Export Switch Dashboard

Select the Export tab, enable the 'Export for sharing externally' option and then save to file:



Name	Date modified	Type	Size
Last month			
pca3-DR-scripts	17/05/2023 15:44	File folder	
Earlier this year			
Virtual Machine Information-1680789275...	06/04/2023 14:54	JSON File	17 KB
FD and CN Utilisation-v1.json	23/03/2023 12:33	JSON File	61 KB
Virtual Machine Information-v1.json	23/03/2023 11:40	JSON File	17 KB
ZFSSA_Dashboard-v1.json	23/03/2023 11:39	JSON File	144 KB
FD and CN Utilisation-1679403769342.json	21/03/2023 13:02	JSON File	49 KB

File name: Switch-1686644697564.json

Save as type: JavaScript Object Notation (*.json)

Import Dashboards – Save exported Switch Dashboard

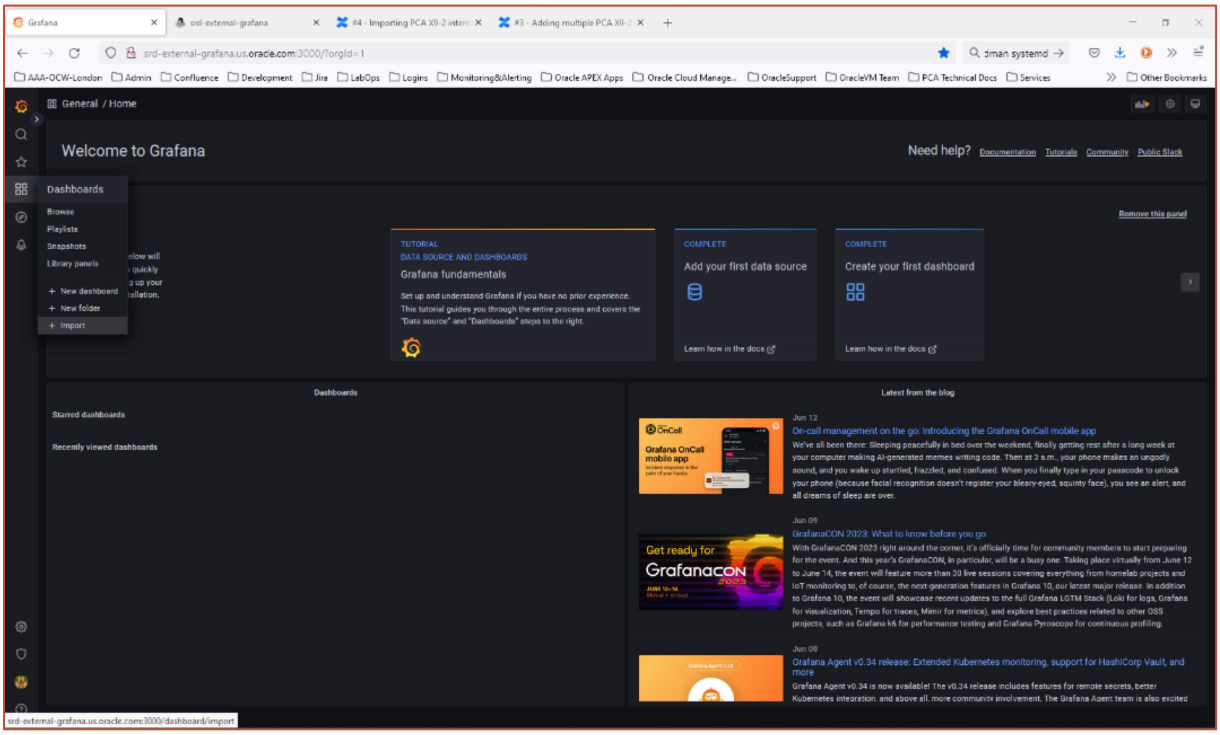
This saves the complete switch dashboard as a JSON file.

Take note of both the filename and its location.

This file can then be imported into the external Grafana Server service.

Import Oracle Private Cloud Appliance X9-2 Grafana Dashboards to PCA-specific folders

Returning to the external Grafana Server instance, select the 'Import' option from the dashboards menu:



Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

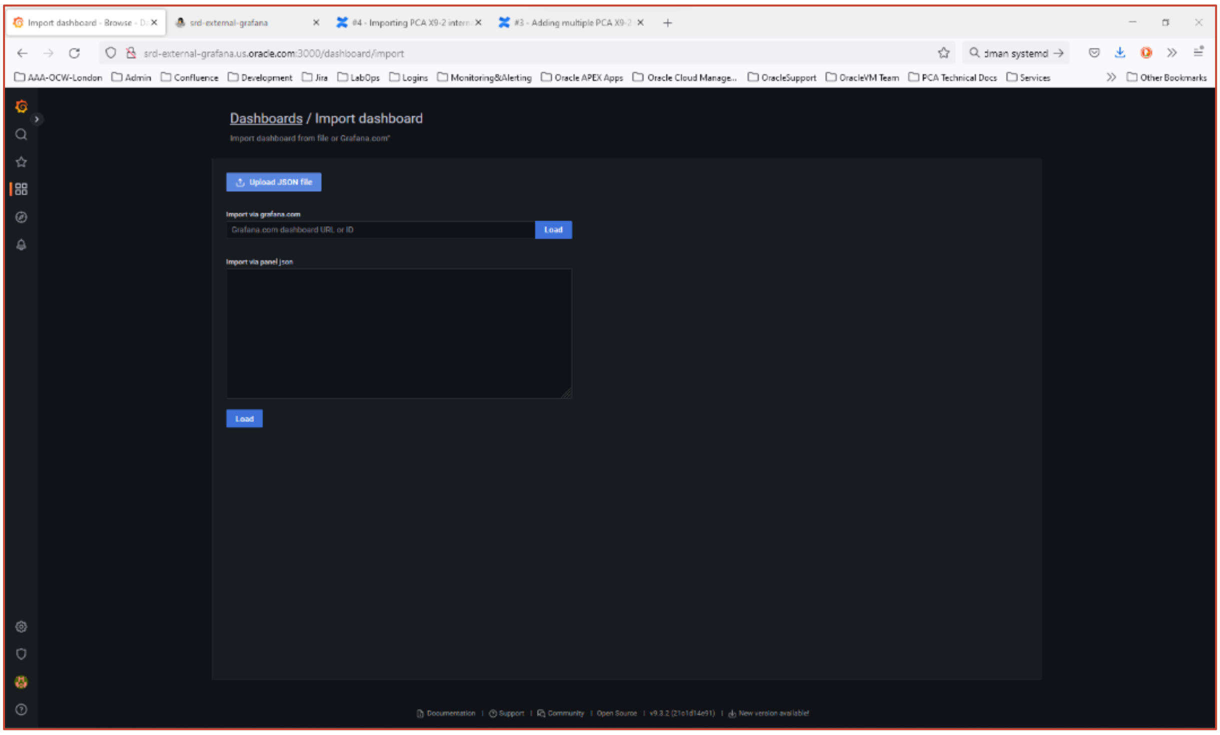
Jun 12
On-call management on the go: Introducing the Grafana OnCall mobile app
We've all been there: Sleeping peacefully in bed over the weekend, finally getting rest after a long week at your computer making AI-generated memes writing code. Then at 3 a.m., your phone makes an ungodly sound, and you wake up startled, frazzled, and confused. When you finally type in your passcode to unlock your phone (because facial recognition doesn't register your bleary-eyed, squinty face), you see an alert, and all dreams of sleep are over.

Jun 09
GrafanaCON 2023: What to know before you go
With GrafanaCON 2023 right around the corner, it's officially time for community members to start preparing for the event. And this year's GrafanaCON, in particular, will be a busy one. Taking place virtually from June 12 to June 14, the event will feature more than 30 live sessions covering everything from homelab projects and IoT monitoring to, of course, the next-generation features in Grafana 10, our latest major release. In addition to Grafana 10, the event will showcase recent updates to the full Grafana LGTM Stack (Loki for logs, Grafana for visualization, Tempo for traces, Mimir for metrics), and explore best practices related to other OSS projects, such as Grafana k6 for performance testing and Grafana Pyroscope for continuous profiling.

Jun 08
Grafana Agent v0.34 release: Extended Kubernetes monitoring, support for HashiCorp Vault, and more
Grafana Agent v0.34 is now available! The v0.34 release includes features for remote secrets, better Kubernetes integration, and above all, more community involvement. The Grafana Agent team is also excited

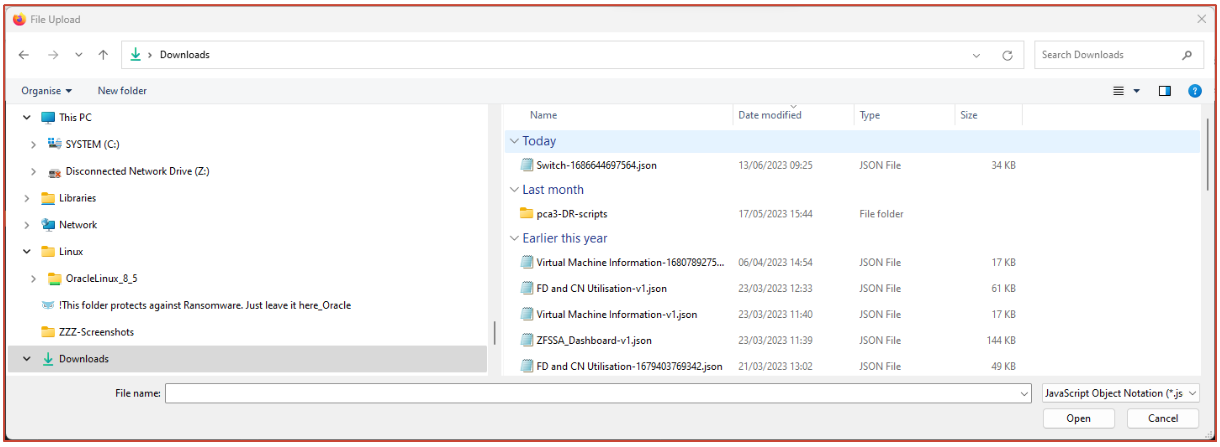
Import Dashboards - Importing

Then click on the 'Upload JSON file' button:


Dashboards / Import dashboard
Import dashboard from file or Grafana.com
Upload JSON file
Import via grafana.com
Grafana.com dashboard URL or ID
Load
Import via panel json
Load

Import Dashboards – Import Switch Dashboard

An import window will open. Select the JSON file saved previously:

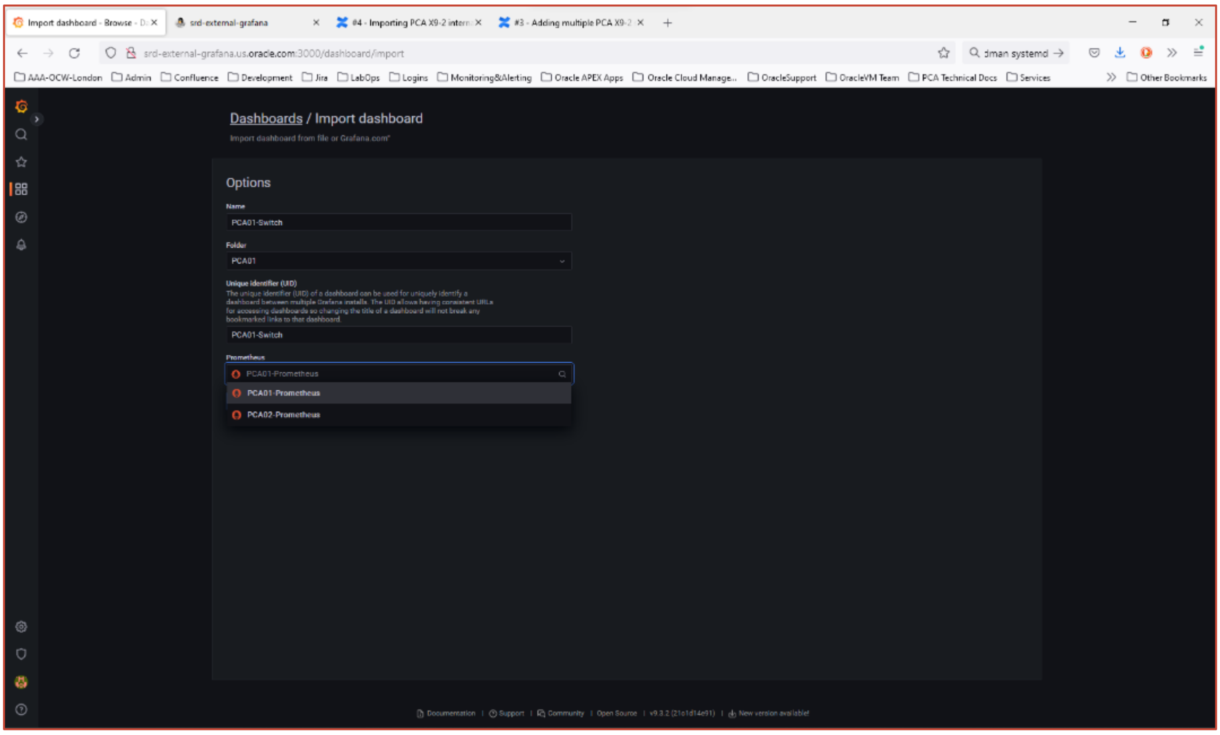


Name	Date modified	Type	Size
Today			
Switch-1686644697564.json	13/06/2023 09:25	JSON File	34 KB
Last month			
pca3-DR-scripts	17/05/2023 15:44	File folder	
Earlier this year			
Virtual Machine Information-1680789275...	06/04/2023 14:54	JSON File	17 KB
FD and CN Utilisation-v1.json	23/03/2023 12:33	JSON File	61 KB
Virtual Machine Information-v1.json	23/03/2023 11:40	JSON File	17 KB
ZFSSA_Dashboard-v1.json	23/03/2023 11:39	JSON File	144 KB
FD and CN Utilisation-1679403769342.json	21/03/2023 13:02	JSON File	49 KB

File name:

Import Dashboards – Select exported JSON file

An import window is then displayed:


Import dashboard - Browse
srd-external-grafana
#4 - Importing PCA X9-2 intern
#3 - Adding multiple PCA X9-2
srd-external-grafana.us.oracle.com:3000/dashboard/import
AAA-OCW-London
Admin
Confluence
Development
Jira
LabOps
Logins
Monitoring&Alerting
Oracle APEX Apps
Oracle Cloud Manage...
OracleSupport
OracleVM Team
PCA Technical Docs
Services
Other Bookmarks
Documentation | Support | Community | Open Source | v9.3.2 (21a1d14e91) | New version available!

Dashboards / Import dashboard

Import dashboard from file or Grafana.com"

Options

Name

PCA01-Switch

Folder

PCA01

Unique identifier (UID)

The unique identifier (UID) of a dashboard can be used for uniquely identify a dashboard between multiple Grafana installs. The UID allows having consistent URLs for accessing dashboards so changing the title of a dashboard will not break any bookmarked links to that dashboard.

PCA01-Switch

Prometheus

PCA01-Prometheus

PCA01-Prometheus

PCA02-Prometheus

Import Dashboard – Importing JSON file

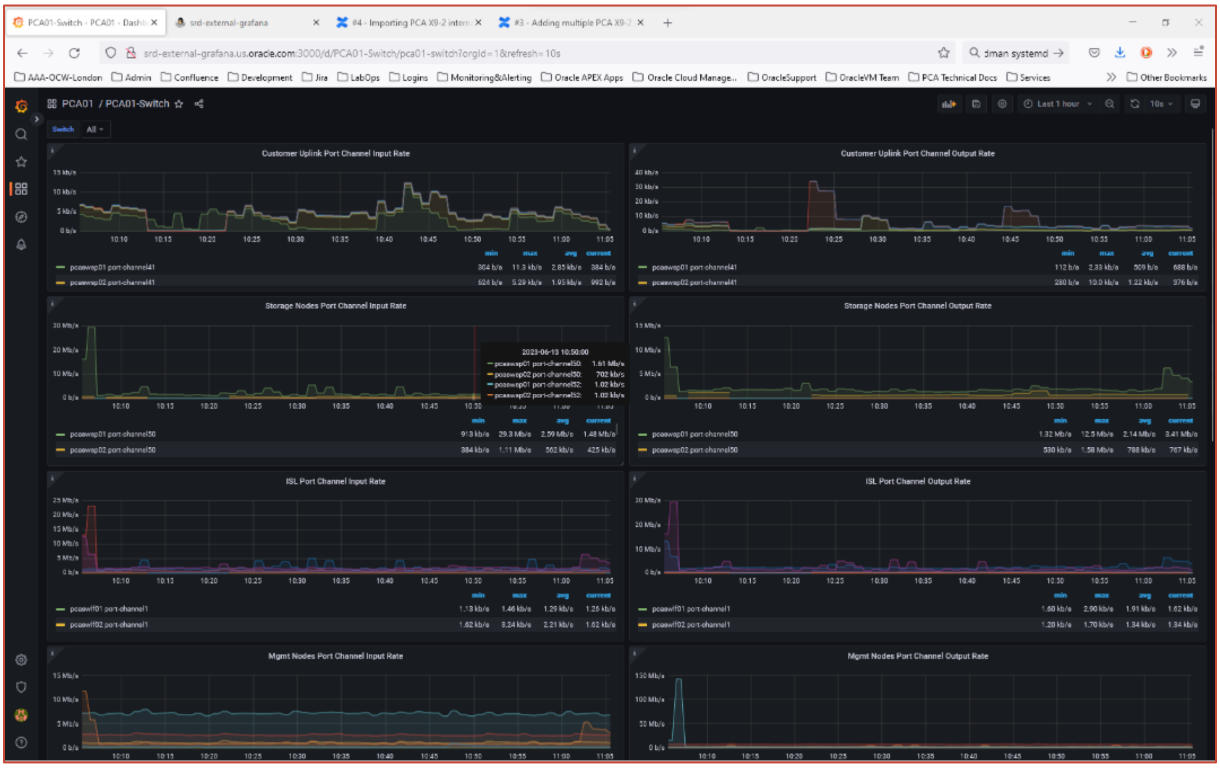
Because this same JSON file is going to be imported several times, there are edits to each of the default values prepopulated within this window.

These are:

- Name:
  - By default, the imported dashboard will retain the name of the original (e.g. Switch)
  - Change the name of the dashboard to reflect the Oracle Private Cloud Appliance X9-2 system it will be displaying the Prometheus data for - in this case; 'PCA01-Switch'
- Folder:
  - The default folder for all dashboard imports is the 'General' folder
  - Change the import folder to the correct PCA specific name - in this case 'PCA01'
- Unique Identifier:
  - Each Grafana dashboard has an alphanumeric UUID. This cannot be duplicated within the Grafana Server service
  - Change the UUID to a PCA specific name - the simplest option here is to repeat the name given to the dashboard above
  - This has an added benefit when Grafana alerting is enabled, since the alerts use the UUID of the dashboard to identify where the alerts originate from
- Prometheus (data source):
  - Select the correct Prometheus data source for the Oracle Private Cloud Appliance X9-2 system - in this case 'PCA01-Prometheus'

Then click on the 'Import' button.

The dashboard will then load:

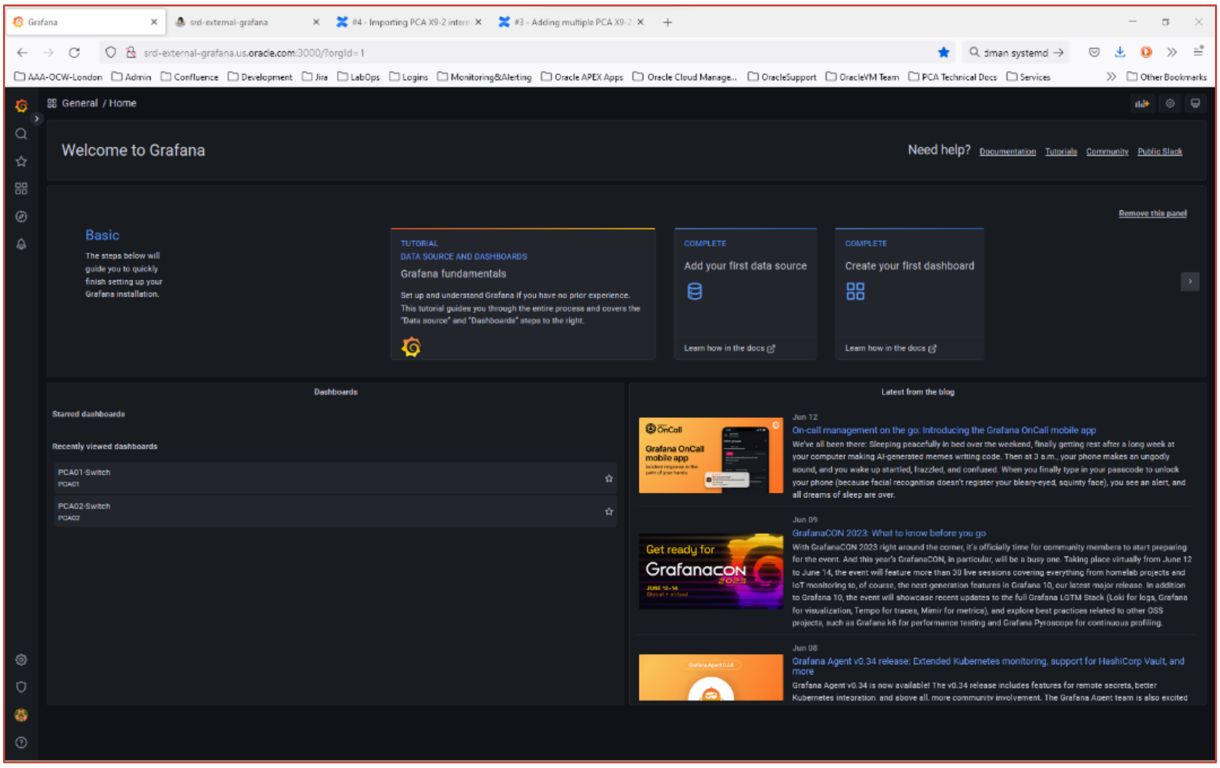

PCA01-Switch - PCA01 - Dashb
srd-external-grafana
#4 - Importing PCA X9-2 intern
#3 - Adding multiple PCA X9-2
srd-external-grafana.us.oracle.com:3000/d/PCA01-Switch/pca01-switch?orgId=1&refresh=10s
PCA01 / PCA01-Switch
Switch All
Last 1 hour
10s
Customer Uplink Port Channel Input Rate
Customer Uplink Port Channel Output Rate
Storage Nodes Port Channel Input Rate
Storage Nodes Port Channel Output Rate
ISL Port Channel Input Rate
ISL Port Channel Output Rate
Mgmt Nodes Port Channel Input Rate
Mgmt Nodes Port Channel Output Rate
min max avg current
pcaswsp01 port-channel41
pcaswsp02 port-channel41
pcaswsp01 port-channel50
pcaswsp02 port-channel50
pcaswlf01 port-channel1
pcaswlf02 port-channel1

Import Dashboards – Imported Switch Dashboard for PCA01

As can be seen in the above screenshot, the folder (PCA01) and dashboard name (PCA01-Switch) are displayed in the top left of the screen and data is already being displayed.

Now repeat again for the second PCA.

Returning to the external Grafana home page now shows the two switch dashboards within the recently viewed dashboards panel:



Welcome to Grafana

Basic

The steps below will guide you to quickly finish setting up your Grafana installation.

TUTORIAL
DATA SOURCE AND DASHBOARDS
Grafana fundamentals

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

COMPLETE
Add your first data source

COMPLETE
Create your first dashboard

Dashboards

Starred dashboards

Recently viewed dashboards

PCA01-Switch
PCA01

PCA02-Switch
PCA02

Latest from the blog

Jun 12
On-call management on the go: Introducing the Grafana OnCall mobile app
We've all been there: Sleeping peacefully in bed over the weekend, finally getting rest after a long week at your computer making AI-generated memes writing code. Then at 3 a.m., your phone makes an ungodly sound, and you wake up startled, frazzled, and confused. When you finally type in your passcode to unlock your phone (because facial recognition doesn't register your bleary-eyed, squinty face), you see an alert, and all dreams of sleep are over.

Jun 09
GrafanaCON 2023: What to know before you go
With GrafanaCON 2023 right around the corner, it's officially time for community members to start preparing for the event. And this year's GrafanaCON, in particular, will be a busy one. Taking place virtually from June 12 to June 14, the event will feature more than 30 live sessions covering everything from homelab projects and IoT monitoring to, of course, the next-generation features in Grafana 10, our latest major release. In addition to Grafana 10, the event will showcase recent updates to the full Grafana LGTM Stack (Loki for logs, Grafana for visualization, Tempo for traces, Mimir for metrics), and explore best practices related to other OSS projects, such as Grafana k6 for performance testing and Grafana Pyroscope for continuous profiling.

Jun 08
Grafana Agent v0.34 release: Extended Kubernetes monitoring, support for HashiCorp Vault, and more
Grafana Agent v0.34 is now available! The v0.34 release includes features for remote secrets, better Kubernetes integration, and above all, more community involvement. The Grafana Agent team is also excited

Import Dashboards – Completed Imports

This same process of exporting & importing Grafana dashboards can then be repeated to populate the external Grafana Server instance with dedicated dashboards for each Oracle Private Cloud Appliance X9-2 system with a valid Prometheus data source defined.

This completes this section on importing Grafana dashboards into the external Grafana Server instance.

For some deployments, just having a single Grafana Server service which can provide a centralized monitoring and alerting reference point may be sufficient.

Further documents will show how this external Grafana Server service can be extended to:

- Combine multiple Oracle Private Cloud Appliance X9-2 systems within a single Grafana dashboard
- Provide in-depth monitoring and alerting capabilities at an operating systems level for:
  - VM instances running within the Oracle Private Cloud Appliance X9-2 systems
  - Physical and/or virtualized systems external to the Oracle Private Cloud Appliance X9-2 systems

Section References

The definitive source for information and instruction for configuring Grafana Server remains the Grafana documentation itself:

- Main Grafana Document Library – https://grafana.com/docs/grafana/latest/
- Grafana Dashboard Documentation – https://grafana.com/docs/grafana/latest/dashboards/

Reference Materials

The following reference URLs provide a consolidated summary of the various section references provided elsewhere within this document:

Oracle References

- Oracle Linux – Reference Library – https://docs.oracle.com/en/operating-systems/oracle-linux/9/
- Oracle Linux – Podman User Guide – https://docs.oracle.com/en/operating-systems/oracle-linux/podman/
- Oracle Linux – Using the Cockpit Web Console – https://docs.oracle.com/en/operating-systems/oracle-linux/cockpit/

Grafana References

- Grafana Document Library – https://grafana.com/docs/grafana/latest/
- Grafana Setup – https://grafana.com/docs/grafana/latest/setup-grafana/
- Grafana Administration – https://grafana.com/docs/grafana/latest/administration/
- Grafana Security – https://grafana.com/docs/grafana/latest/setup-grafana/configure-security/
- Grafana Alerting – https://grafana.com/docs/grafana/latest/alerting/
- Grafana Data Source documentation – https://grafana.com/docs/grafana/latest/datasources/
- Grafana Dashboard Documentation – https://grafana.com/docs/grafana/latest/dashboards/

Other References

- Cockpit documentation – https://cockpit-project.org/documentation.html
- Performance Co-Pilot documentation – https://pcp.readthedocs.io/en/latest/
- Performance Co-Pilot documentation – Quick Reference Guide – https://pcp.readthedocs.io/en/latest/QG/QuickReferenceGuide.html

Connect with us

Call +**1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

blogs.oracle.com facebook.com/oracle twitter.com/oracle

Disclaimer: If you are unsure whether your data sheet needs a disclaimer, read the revenue recognition policy. If you have further questions about your content and the disclaimer requirements, e-mail REVREC_US@oracle.com.