Ansible Workshop - Exercises

Projects

Use your Ansible skills to complete a couple of small projects.

Project - Linux automation

To further enhance your Ansible skills, let's deploy the monitoring tool *Grafana* to one of the nodes in the demo environment.

Objective

Create an Ansible project *from scratch* and automate some basic linux configurations.

Guide

Step 1 - Prepare project

Create a new project folder in your home directory:

```
[student@ansible-1 ~]$ mkdir grafana-deployment
```

Create an inventory file with a *grafana* group definition. You will deploy Grafana to one of the nodes in the lab environment. Copy the *node2* configuration from the default inventory file to your *grafana* group.
Create a small Ansible configuration file (`ansible.cfg`) and instruct Ansible to always use the inventory you just created.

For example, you may check your inventory with the `ansible-inventory` CLI utility. In this case, the host has an alias of *grafana-instance1* which is part of a group *grafana*:

```
[student@ansible-1 ~]$ ansible-inventory --graph --vars
@all:
  |--@ungrouped:
  |--@grafana:
  |  |--grafana-instance1
  |  |  |--{ansible_host = node2.example.com}
```

Hint

As you can see above, no inventory was provided in the CLI call (e.g. with `-i inventory`), but the correct inventory is used.

Achieve the following tasks:

- Inventory file created
- Configuration file created which sets the correct inventory source

Step 2 - Install Grafana

The Grafana package comes from a dedicated repository, you'll need to enable it for the *yum* package manager on *node2*. Use the following file and copy it to `/etc/yum.repos.d/grafana.repo` with an Ansible task:

```
[grafana]
name=grafana
baseurl=https://packages.grafana.com/oss/rpm
repo_gpgcheck=1
enabled=1
gpgcheck=1
gpgkey=https://packages.grafana.com/gpg.key
sslverify=1
sslcacert=/etc/pki/tls/certs/ca-bundle.crt
```

The next task should install the `grafana` package. Another task is needed to start (and enable) the `grafana-server` service.

Achieve the following tasks:

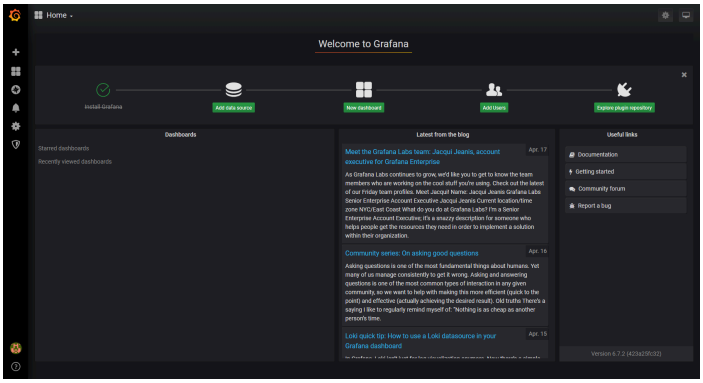
- Running Grafana instance on node2
- Grafana service running and enabled at startup

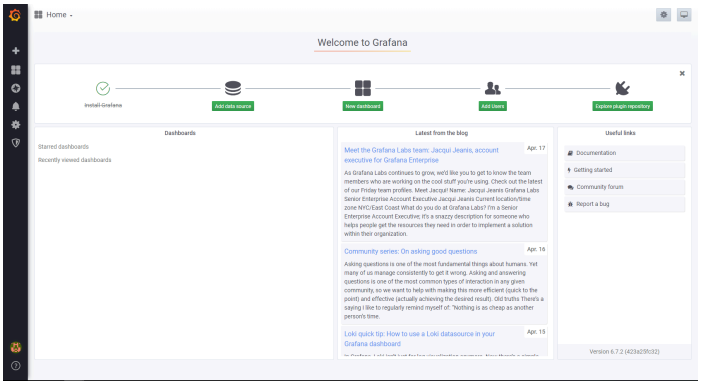
Ensure that Grafana is running with an ad hoc command:

```
[student@ansible-1 grafana-deployment]$ ansible grafana -a "systemctl status grafana-server"
node2 | CHANGED | rc=0 >>
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset:
disabled)
   Active: active (running) since Sun 2022-04-17 10:00:35 UTC; 2min 44s ago
     Docs: http://docs.grafana.org
 Main PID: 20887 (grafana-server)
    Tasks: 7 (limit: 4579)
   Memory: 97.7M
   CGroup: /system.slice/grafana-server.service
           └─20887 /usr/sbin/grafana-server --config=/etc/grafana/grafana.ini --
pidfile=/var/run/grafana/grafana-server.pid --packaging=rpm
cfg:default.paths.logs=/var/log/grafana cfg:default.paths.data=/var/lib/grafana
cfg:default.paths.plugins=/var/lib/grafana/plugins
cfg:default.paths.provisioning=/etc/grafana/provisioning
```

Nice, the Grafana package is installed and the service is running!
Grafana has a nice UI, unfortunately, the UI currently can't be viewed directly in the Red Hat Demo environment!

This is how it would look, on the left in the default **dark** theme, the right screenshot shows the **light** theme.


Home
Welcome to Grafana
Install Grafana
Add data source
New dashboard
Add Users
Explore plugin repository
Dashboards
Starred dashboards
Recently viewed dashboards
Latest from the blog
Meet the Grafana Labs team: Jacqui Jeanis, account executive for Grafana Enterprise
Apr. 17
As Grafana Labs continues to grow, we'd like you to get to know the team members who are working on the cool stuff you're using. Check out the latest of our Friday team profiles. Meet Jacqui Name: Jacqui Jeanis Grafana Labs Senior Enterprise Account Executive Jacqui Jeanis Current location/time zone NYC/East Coast What do you do at Grafana Labs? I'm a Senior Enterprise Account Executive. It's a snazzy description for someone who helps people get the resources they need in order to implement a solution within their organization.
Community series: On asking good questions
Apr. 16
Asking questions is one of the most fundamental things about humans. Yet many of us manage consistently to get it wrong. Asking and answering questions is one of the most common types of interaction in any given community, so we want to help with making this more efficient (quick to the point) and effective (actually achieving the desired result). Old truths There's a saying I like to regularly remind myself of: "Nothing is as cheap as another person's time.
Loki quick tip: How to use a Loki datasource in your Grafana dashboard
Apr. 15
Useful links
Documentation
Getting started
Community forum
Report a bug
Version 6.7.2 (423a29fc32)


Home
Welcome to Grafana
Install Grafana
Add data source
New dashboard
Add Users
Explore plugin repository
Dashboards
Starred dashboards
Recently viewed dashboards
Latest from the blog
Meet the Grafana Labs team: Jacqui Jeanis, account executive for Grafana Enterprise
Apr. 17
As Grafana Labs continues to grow, we'd like you to get to know the team members who are working on the cool stuff you're using. Check out the latest of our Friday team profiles. Meet Jacqui Name: Jacqui Jeanis Grafana Labs Senior Enterprise Account Executive Jacqui Jeanis Current location/time zone NYC/East Coast What do you do at Grafana Labs? I'm a Senior Enterprise Account Executive. It's a snazzy description for someone who helps people get the resources they need in order to implement a solution within their organization.
Community series: On asking good questions
Apr. 16
Asking questions is one of the most fundamental things about humans. Yet many of us manage consistently to get it wrong. Asking and answering questions is one of the most common types of interaction in any given community, so we want to help with making this more efficient (quick to the point) and effective (actually achieving the desired result). Old truths There's a saying I like to regularly remind myself of: "Nothing is as cheap as another person's time.
Loki quick tip: How to use a Loki datasource in your Grafana dashboard
Apr. 15
Useful links
Documentation
Getting started
Community forum
Report a bug
Version 6.7.2 (423a29fc32)

Success

You can use the following playbook to check the current theme setting, create a new file, paste to content and run it:

```
- name: Test Grafana theme setting
  hosts: node2
  vars:
    # By default, Grafana listens on port 3000!
    # Change the value to '8080' if you want to check the target port later on
    grafana_port: 3000
  tasks:
    - name: Get Grafana UI content
      ansible.builtin.uri:
        url: http://node2:{{ grafana_port }}
        return_content: true
      register: grafana_ui_content

    - name: Output current theme setting
      ansible.builtin.debug:
        msg: "HTML body returns '{{ grafana_ui_content.content | replace('\n', '') |
regex_replace('^(.*body class=\\\")(.*)( app-grafana.*)', '\\2') }}' as the current color setting."
```

NOT in the Red Hat Demo environment? Click me.

If you are in a local environment, you can use the default login credentials *admin:admin*, you can skip the password change request.
You can check if the UI is available by using the curl request `curl -L node2:8080`

▼ Details

```
[student@ansible-1 ansible_files]$ curl -L node2:8080
<!DOCTYPE html>
<html lang="en">
  <head>
    <script nonce="">

      !(function () {
        if ('PerformanceLongTaskTiming' in window) {
          var g = (window.__tti = { e: [] });
          g.o = new PerformanceObserver(function (l) {
            g.e = g.e.concat(l.getEntries());
          });
          g.o.observe({ entryTypes: ['longtask'] });
        }
      })();
    </script>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1" />
    <meta name="viewport" content="width=device-width" />
    <meta name="theme-color" content="#000" />

    <title>Grafana</title>

    <base href="/" />

    <link
      rel="preload"
      href="public/fonts/roboto/RxZJdnzeo3R5zSexge8UUVtXRa8TVwTICgirnJhmVJw.woff2"
      as="font"
      crossorigin
    />

    <link rel="icon" type="image/png" href="public/img/fav32.png" />
    <link rel="apple-touch-icon" sizes="180x180" href="public/img/apple-touch-icon.png" />
    <link rel="mask-icon" href="public/img/grafana_mask_icon.svg" color="#F05A28" />
    <link rel="stylesheet" href="public/build/grafana.dark.3b87c7ad03e52dfc5e30.css" />

    <script nonce="">
      performance.mark('frontend_boot_css_time_seconds');
    </script>

    <meta name="apple-mobile-web-app-capable" content="yes" />
    <meta name="apple-mobile-web-app-status-bar-style" content="black" />
    <meta name="msapplication-TileColor" content="#2b5797" />
    <meta name="msapplication-config" content="public/img/browserconfig.xml" />
  </head>

  <body class="theme-light app-grafana">
    <style>
      .preloader {
        height: 100%;
        flex-direction: column;
        display: flex;
        justify-content: center;
        align-items: center;
      }
  ...<cut for readability>...
```

Step 3 - Configure Grafana

By default, Grafana provides the webinterface on **port 3000** and uses a **dark background**.
You need to adjust the Grafana configuration with Ansible to instruct the service to show the webinterface on **port 8080** and to configure a **white background (the so-called `light` theme)**.

The configuration for Grafana is stored in `/etc/grafana/grafana.ini`. You need to adjust the theme configuration in the *users* section, as well as the *http_port* in the *server* section. Take a look at the Grafana documentation on how to change the parameters.
Naturally, you should achieve this with Ansible! Find an appropriate module (there is more than one way to achieve the solution...) and adjust the Grafana configuration file.

Tip

Configuration changes require a service restart!

Tip

Login to `node2` via ssh, take a look at the file. You can download the file from there, copy the content or just take a look at how it looks to be able to find the appropraiate lines to change, there are many ways to achieve the solution!

Yesterday you started an Apache webserver on port 8080 with Ansible. **Two services (Apache/Httpd and Grafana) can not use the same port!**.

Warning

There should be no running Apache webserver on *node2*, if Apache is running, you'll need to stop *httpd* on *node2*! If the port is occupied, Grafana can not be started!
You could (and should!) ensure a stopped Apache easily with an Ansible task...

After adjusting the configuration, run the playbook above (which checks the Grafana UI theme settings), adjust the `grafana_port` variable to use the new port 8080.

Achieve the following tasks:

- Accessible Grafana UI on port 8080
- Grafana UI in `light` theme
- Bonus: Can you manage to control the look of Grafana by just switching a variable?

Step 4 - Re-format project to role structure

All Ansible projects should use the role structure, if your project does not already uses it, now is the time to rearrange your content. Create a `roles` folder and an appropriately named sub-folder for the grafana deployment with all necessary folder and files.
Change your playbook to use your role, e.g.:

```yaml
---
# This is the main Playbook for the 'Grafana Deployment' Project

- name: Deploy Grafana instance
  hosts: grafana
  roles:
    - grafana
```

Make sure everything works by executing your playbook again, you should not see any changes, all tasks should return a green "Ok" status.

Achieve the following tasks:

- Project uses Ansible role structure
- Playbook references role

Step 5 - Bonus: Upload project to Github

Create a new project in your personal Github account and commit your Ansible project.

Step 6 - Bonus: Run your project within AAP

Create a new project in AAP, reference your Grafana project from Github as the code source. Create a template and run your playbook.