To implement Grafana on an EC2 instance, follow these steps:

### **1. Set up the EC2 Instance**

1. **Launch an EC2 Instance:**
   1. Choose an appropriate instance type (e.g., t2.micro for testing, or larger for production).
   2. Select an OS (Amazon Linux 2, Ubuntu, or other Linux distributions are common choices).
   3. Configure security groups to allow access to Grafana (default port: 3000).
2. **SSH into the Instance:**
   1. Use the key pair you created during the EC2 launch.

bash

Copy code

ssh -i "your-key.pem" ec2-user@your-ec2-ip

### **2. Install Grafana**

#### **For Amazon Linux 2:**

1. Add Grafana's repository:

bash

Copy code

sudo tee /etc/yum.repos.d/grafana.repo <<EOF  
[grafana]  
name=Grafana  
baseurl=https://packages.grafana.com/oss/rpm  
repo\_gpgcheck=1  
enabled=1  
gpgcheck=1  
gpgkey=https://packages.grafana.com/gpg.key  
EOF

1. Install Grafana:

bash

Copy code

sudo yum install -y grafana

#### **For Ubuntu/Debian:**

1. Add Grafana's APT repository:

bash

Copy code

sudo apt-get install -y software-properties-common  
sudo add-apt-repository "deb <https://packages.grafana.com/oss/deb> stable main"  
wget -q -O - <https://packages.grafana.com/gpg.key> | sudo apt-key add -

1. Update and install Grafana:

bash

Copy code

sudo apt-get update  
sudo apt-get install -y grafana

### **3. Start Grafana**

1. Enable Grafana to start on boot:

bash

Copy code

sudo systemctl enable grafana-server

1. Start the Grafana service:

bash

Copy code

sudo systemctl start grafana-server

1. Check the status:

bash

Copy code

sudo systemctl status grafana-server

### **4. Configure Security Group**

1. Edit the security group of your EC2 instance.
2. Add an inbound rule to allow traffic on port 3000 (Grafana's default port):
   1. Protocol: TCP
   2. Port Range: 3000
   3. Source: 0.0.0.0/0 (for global access) or restrict to specific IPs.

### **5. Access Grafana**

1. Open your web browser and navigate to:

vbnet

Copy code

http://<EC2-public-IP>:3000

1. Log in with the default credentials:
   1. Username: admin
   2. Password: admin (you will be prompted to change it after logging in).

### **6. Add Data Sources**

1. Log into the Grafana UI.
2. Go to **Configuration > Data Sources**.
3. Add and configure data sources like Prometheus, MySQL, PostgreSQL, etc.

### **7. Create Dashboards**

1. Navigate to **+ (Create) > Dashboard**.
2. Add panels to visualize metrics from the configured data sources.

### **8. Optional: Enable Public Access (Reverse Proxy or Domain)**

* Use **Nginx** or **Apache** as a reverse proxy to make Grafana accessible via a custom domain.
* Secure Grafana with **TLS/SSL** using tools like Let’s Encrypt.

To configure **Grafana**, **Prometheus**, **Alertmanager**, and **Node Exporter** to monitor metrics on AWS, follow this comprehensive guide:

### **1. Set Up EC2 Instances**

1. **Launch EC2 Instances:**
   1. Create an instance for running Prometheus, Grafana, and Alertmanager.
   2. Create additional EC2 instances for Node Exporter (target instances).
2. **Configure Security Groups:**
   1. Allow inbound traffic for:
      1. Grafana: Port **3000**
      2. Prometheus: Port **9090**
      3. Alertmanager: Port **9093**
      4. Node Exporter: Port **9100**
   2. Enable SSH access to all instances.
3. **Install Required Tools:**
   1. Install wget, curl, and tar on all instances.

### **2. Install and Configure Prometheus**

1. **Download Prometheus:** On the Prometheus server instance:

bash

Copy code

wget <https://github.com/prometheus/prometheus/releases/download/v2.47.0/prometheus-2.47.0.linux-amd64.tar.gz>tar -xvzf prometheus-2.47.0.linux-amd64.tar.gz  
cd prometheus-2.47.0.linux-amd64

1. **Configure Prometheus:** Edit the prometheus.yml file to scrape metrics from Node Exporter and Alertmanager:

yaml

Copy code

scrape\_configs:  
 - job\_name: 'node\_exporter'  
 static\_configs:  
 - targets: ['<node1-ip>:9100', '<node2-ip>:9100']  
  
 - job\_name: 'alertmanager'  
 static\_configs:  
 - targets: ['localhost:9093']

1. **Run Prometheus:**

bash

Copy code

./prometheus --config.file=prometheus.yml

1. **Access Prometheus UI:** Open http://<Prometheus-Server-IP>:9090.

### **3. Install and Configure Node Exporter**

1. **Download Node Exporter:** On all target EC2 instances:

bash

Copy code

wget <https://github.com/prometheus/node_exporter/releases/download/v1.6.1/node_exporter-1.6.1.linux-amd64.tar.gz>tar -xvzf node\_exporter-1.6.1.linux-amd64.tar.gz  
cd node\_exporter-1.6.1.linux-amd64

1. **Run Node Exporter:**

bash

Copy code

./node\_exporter

1. **Verify Node Exporter:** Open http://<Node-Exporter-Instance-IP>:9100/metrics.

### **4. Install and Configure Alertmanager**

1. **Download Alertmanager:** On the Prometheus server instance:

bash

Copy code

wget <https://github.com/prometheus/alertmanager/releases/download/v0.26.0/alertmanager-0.26.0.linux-amd64.tar.gz>tar -xvzf alertmanager-0.26.0.linux-amd64.tar.gz  
cd alertmanager-0.26.0.linux-amd64

1. **Create Alertmanager Configuration:** Edit alertmanager.yml:

yaml

Copy code

global:  
 resolve\_timeout: 5m  
  
route:  
 receiver: 'email'  
  
receivers:  
 - name: 'email'  
 email\_configs:  
 - to: '[your-email@example.com](mailto:your-email@example.com)'  
 from: '[alertmanager@example.com](mailto:alertmanager@example.com)'  
 smarthost: 'smtp.example.com:587'  
 auth\_username: 'smtp-username'  
 auth\_password: 'smtp-password'

1. **Run Alertmanager:**

bash

Copy code

./alertmanager --config.file=alertmanager.yml

1. **Verify Alertmanager:** Open http://<Alertmanager-Server-IP>:9093.

### **5. Install and Configure Grafana**

1. **Download and Install Grafana:** Follow the [Grafana setup steps](https://inc-word-edit.officeapps.live.com/we/wordeditorframe.aspx?ui=en-US&rs=en-US&wopisrc=https%3A%2F%2Fcreditmantrifinservepvtltd-my.sharepoint.com%2Fpersonal%2Fsanthosh_d_creditmantri_com%2F_vti_bin%2Fwopi.ashx%2Ffiles%2F112ff7ecac92436da11b9358f6d2672d&wdenableroaming=1&mscc=1&wdodb=1&hid=332369A1-A09D-4000-2E5D-894B27B0B2D7.0&uih=sharepointcom&wdlcid=en-US&jsapi=1&jsapiver=v2&corrid=b3088dac-d58b-a3cd-fa50-7807820828e2&usid=b3088dac-d58b-a3cd-fa50-7807820828e2&newsession=1&sftc=1&uihit=docaspx&muv=1&cac=1&sams=1&mtf=1&sfp=1&sdp=1&hch=1&hwfh=1&dchat=1&sc=%7B%22pmo%22%3A%22https%3A%2F%2Fcreditmantrifinservepvtltd-my.sharepoint.com%22%2C%22pmshare%22%3Atrue%7D&ctp=LeastProtected&rct=Normal&wdorigin=Other&csc=1&instantedit=1&wopicomplete=1&wdredirectionreason=Unified_SingleFlush#how-to-implement-grafana-on-ec2) to install Grafana on an EC2 instance.
2. **Add Prometheus as a Data Source:**
   1. Open Grafana at http://<Grafana-Server-IP>:3000.
   2. Go to **Configuration > Data Sources**.
   3. Add Prometheus:
      1. URL: http://<Prometheus-Server-IP>:9090
3. **Import Dashboards:**
   1. Use pre-built Grafana dashboards for Node Exporter.
   2. Import dashboards from Grafana.com.

### **6. Create Alerts and Notifications**

1. **Configure Alerts in Prometheus:** Edit the prometheus.yml file:

yaml

Copy code

alerting:  
 alertmanagers:  
 - static\_configs:  
 - targets: ['localhost:9093']  
  
rule\_files:  
 - "alert.rules.yml"

1. **Create Alert Rules:** In alert.rules.yml:

yaml

Copy code

groups:  
 - name: instance\_health  
 rules:  
 - alert: HighCPUUsage  
 expr: 100 - (avg by (instance) (rate(node\_cpu\_seconds\_total{mode="idle"}[1m])) \* 100) > 80  
 for: 1m  
 labels:  
 severity: critical  
 annotations:  
 summary: "High CPU Usage"  
 description: "CPU usage is above 80% on {{ $labels.instance }}"

1. **Reload Prometheus Configuration:** Reload Prometheus by sending a SIGHUP or restarting the service.

### **7. Verify Metrics and Alerts**

* Access Grafana dashboards to see real-time metrics from Node Exporter.
* Test Alertmanager by triggering alerts and checking notifications.

### **8. Optional Enhancements**

* **Automate Startup:** Use systemd service files for Prometheus, Alertmanager, and Node Exporter.
* **TLS/SSL:** Secure your Grafana and Prometheus endpoints with HTTPS.
* **Scalability:** Use EC2 Auto Scaling Groups and Load Balancers for high availability.

**Important Note:- Example document**

**Grafana configuration file :-**

**Config file : grafana.ini**

**##################### Grafana Configuration Example #####################**

**#**

**# Everything has defaults so you only need to uncomment things you want to**

**# change**

**# possible values : production, development**

**;app\_mode = production**

**# instance name, defaults to HOSTNAME environment variable value or hostname if HOSTNAME var is empty**

**;instance\_name = ${HOSTNAME}**

**#################################### Paths ####################################**

**[paths]**

**# Path to where grafana can store temp files, sessions, and the sqlite3 db (if that is used)**

**;data = /var/lib/grafana**

**# Temporary files in `data` directory older than given duration will be removed**

**;temp\_data\_lifetime = 24h**

**# Directory where grafana can store logs**

**;logs = /var/log/grafana**

**# Directory where grafana will automatically scan and look for plugins**

**;plugins = /var/lib/grafana/plugins**

**# folder that contains provisioning config files that grafana will apply on startup and while running.**

**;provisioning = conf/provisioning**

**#################################### Server ####################################**

**[server]**

**# Protocol (http, https, h2, socket)**

**;protocol = http**

**# The ip address to bind to, empty will bind to all interfaces**

**;http\_addr =**

**# The http port to use**

**;http\_port =**

**# The public facing domain name used to access grafana from a browser**

**;domain = https://monitoring.creditmantri.com**

**# Redirect to correct domain if host header does not match domain**

**# Prevents DNS rebinding attacks**

**;enforce\_domain = false**

**# The full public facing url you use in browser, used for redirects and emails**

**# If you use reverse proxy and sub path specify full url (with sub path)**

**root\_url = https://monitoring.creditmantri.com/**

**# Serve Grafana from subpath specified in `root\_url` setting. By default it is set to `false` for compatibility reasons.**

**;serve\_from\_sub\_path = false**

**# Log web requests**

**;router\_logging = false**

**# the path relative working path**

**;static\_root\_path = public**

**# enable gzip**

**;enable\_gzip = false**

**# https certs & key file**

**;cert\_file =**

**;cert\_key =**

**# Unix socket path**

**;socket =**

**# CDN Url**

**;cdn\_url =**

**# Sets the maximum time using a duration format (5s/5m/5ms) before timing out read of an incoming request and closing idle connections.**

**# `0` means there is no timeout for reading the request.**

**;read\_timeout = 0**

**#################################### Database ####################################**

**[database]**

**# You can configure the database connection by specifying type, host, name, user and password**

**# as separate properties or as on string using the url properties.**

**# Either "mysql", "postgres" or "sqlite3", it's your choice**

**;type = sqlite3**

**;host = 127.0.0.1:3306**

**;name = grafana**

**;user = root**

**# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""**

**;password =**

**# Use either URL or the previous fields to configure the database**

**# Example: mysql://user:secret@host:port/database**

**;url =**

**# For "postgres" only, either "disable", "require" or "verify-full"**

**;ssl\_mode = disable**

**# Database drivers may support different transaction isolation levels.**

**# Currently, only "mysql" driver supports isolation levels.**

**# If the value is empty - driver's default isolation level is applied.**

**# For "mysql" use "READ-UNCOMMITTED", "READ-COMMITTED", "REPEATABLE-READ" or "SERIALIZABLE".**

**;isolation\_level =**

**;ca\_cert\_path =**

**;client\_key\_path =**

**;client\_cert\_path =**

**;server\_cert\_name =**

**# For "sqlite3" only, path relative to data\_path setting**

**;path = grafana.db**

**# Max idle conn setting default is 2**

**;max\_idle\_conn = 2**

**# Max conn setting default is 0 (mean not set)**

**;max\_open\_conn =**

**# Connection Max Lifetime default is 14400 (means 14400 seconds or 4 hours)**

**;conn\_max\_lifetime = 14400**

**# Set to true to log the sql calls and execution times.**

**;log\_queries =**

**# For "sqlite3" only. cache mode setting used for connecting to the database. (private, shared)**

**;cache\_mode = private**

**################################### Data sources #########################**

**[datasources]**

**# Upper limit of data sources that Grafana will return. This limit is a temporary configuration and it will be deprecated when pagination will be introduced on the list data sources API.**

**;datasource\_limit = 5000**

**#################################### Cache server #############################**

**[remote\_cache]**

**# Either "redis", "memcached" or "database" default is "database"**

**;type = database**

**# cache connectionstring options**

**# database: will use Grafana primary database.**

**# redis: config like redis server e.g. `addr=127.0.0.1:6379,pool\_size=100,db=0,ssl=false`. Only addr is required. ssl may be 'true', 'false', or 'insecure'.**

**# memcache: 127.0.0.1:11211**

**;connstr =**

**#################################### Data proxy ###########################**

**[dataproxy]**

**# This enables data proxy logging, default is false**

**;logging = false**

**# How long the data proxy waits before timing out, default is 30 seconds.**

**# This setting also applies to core backend HTTP data sources where query requests use an HTTP client with timeout set.**

**;timeout = 30**

**# How many seconds the data proxy waits before sending a keepalive probe request.**

**;keep\_alive\_seconds = 30**

**# How many seconds the data proxy waits for a successful TLS Handshake before timing out.**

**;tls\_handshake\_timeout\_seconds = 10**

**# How many seconds the data proxy will wait for a server's first response headers after**

**# fully writing the request headers if the request has an "Expect: 100-continue"**

**# header. A value of 0 will result in the body being sent immediately, without**

**# waiting for the server to approve.**

**;expect\_continue\_timeout\_seconds = 1**

**# The maximum number of idle connections that Grafana will keep alive.**

**;max\_idle\_connections = 100**

**# How many seconds the data proxy keeps an idle connection open before timing out.**

**;idle\_conn\_timeout\_seconds = 90**

**# If enabled and user is not anonymous, data proxy will add X-Grafana-User header with username into the request, default is false.**

**;send\_user\_header = false**

**#################################### Analytics ####################################**

**[analytics]**

**# Server reporting, sends usage counters to stats.grafana.org every 24 hours.**

**# No ip addresses are being tracked, only simple counters to track**

**# running instances, dashboard and error counts. It is very helpful to us.**

**# Change this option to false to disable reporting.**

**;reporting\_enabled = true**

**# The name of the distributor of the Grafana instance. Ex hosted-grafana, grafana-labs**

**;reporting\_distributor = grafana-labs**

**# Set to false to disable all checks to https://grafana.net**

**# for new versions (grafana itself and plugins), check is used**

**# in some UI views to notify that grafana or plugin update exists**

**# This option does not cause any auto updates, nor send any information**

**# only a GET request to http://grafana.com to get latest versions**

**;check\_for\_updates = true**

**# Google Analytics universal tracking code, only enabled if you specify an id here**

**;google\_analytics\_ua\_id =**

**# Google Tag Manager ID, only enabled if you specify an id here**

**;google\_tag\_manager\_id =**

**#################################### Security ####################################**

**[security]**

**# disable creation of admin user on first start of grafana**

**;disable\_initial\_admin\_creation = false**

**# default admin user, created on startup**

**;admin\_user = admin**

**# default admin password, can be changed before first start of grafana, or in profile settings**

**;admin\_password = admin**

**# used for signing**

**;secret\_key = SW2YcwTIb9zpOOhoPsMm**

**# disable gravatar profile images**

**;disable\_gravatar = false**

**# data source proxy whitelist (ip\_or\_domain:port separated by spaces)**

**;data\_source\_proxy\_whitelist =**

**# disable protection against brute force login attempts**

**;disable\_brute\_force\_login\_protection = false**

**# set to true if you host Grafana behind HTTPS. default is false.**

**;cookie\_secure = false**

**# set cookie SameSite attribute. defaults to `lax`. can be set to "lax", "strict", "none" and "disabled"**

**;cookie\_samesite = lax**

**# set to true if you want to allow browsers to render Grafana in a <frame>, <iframe>, <embed> or <object>. default is false.**

**;allow\_embedding = false**

**# Set to true if you want to enable http strict transport security (HSTS) response header.**

**# This is only sent when HTTPS is enabled in this configuration.**

**# HSTS tells browsers that the site should only be accessed using HTTPS.**

**;strict\_transport\_security = false**

**# Sets how long a browser should cache HSTS. Only applied if strict\_transport\_security is enabled.**

**;strict\_transport\_security\_max\_age\_seconds = 86400**

**# Set to true if to enable HSTS preloading option. Only applied if strict\_transport\_security is enabled.**

**;strict\_transport\_security\_preload = false**

**# Set to true if to enable the HSTS includeSubDomains option. Only applied if strict\_transport\_security is enabled.**

**;strict\_transport\_security\_subdomains = false**

**# Set to true to enable the X-Content-Type-Options response header.**

**# The X-Content-Type-Options response HTTP header is a marker used by the server to indicate that the MIME types advertised**

**# in the Content-Type headers should not be changed and be followed.**

**;x\_content\_type\_options = true**

**# Set to true to enable the X-XSS-Protection header, which tells browsers to stop pages from loading**

**# when they detect reflected cross-site scripting (XSS) attacks.**

**;x\_xss\_protection = true**

**# Enable adding the Content-Security-Policy header to your requests.**

**# CSP allows to control resources the user agent is allowed to load and helps prevent XSS attacks.**

**;content\_security\_policy = false**

**# Set Content Security Policy template used when adding the Content-Security-Policy header to your requests.**

**# $NONCE in the template includes a random nonce.**

**;content\_security\_policy\_template = """script-src 'unsafe-eval' 'strict-dynamic' $NONCE;object-src 'none';font-src 'self';style-src 'self' 'unsafe-inline';img-src 'self' data:;base-uri 'self';connect-src 'self' grafana.com;manifest-src 'self';media-src 'none';form-action 'self';"""**

**#################################### Snapshots ###########################**

**[snapshots]**

**# snapshot sharing options**

**;external\_enabled = true**

**;external\_snapshot\_url = https://snapshots-origin.raintank.io**

**;external\_snapshot\_name = Publish to snapshot.raintank.io**

**# Set to true to enable this Grafana instance act as an external snapshot server and allow unauthenticated requests for**

**# creating and deleting snapshots.**

**;public\_mode = false**

**# remove expired snapshot**

**;snapshot\_remove\_expired = true**

**#################################### Dashboards History ##################**

**[dashboards]**

**# Number dashboard versions to keep (per dashboard). Default: 20, Minimum: 1**

**;versions\_to\_keep = 20**

**# Minimum dashboard refresh interval. When set, this will restrict users to set the refresh interval of a dashboard lower than given interval. Per default this is 5 seconds.**

**# The interval string is a possibly signed sequence of decimal numbers, followed by a unit suffix (ms, s, m, h, d), e.g. 30s or 1m.**

**;min\_refresh\_interval = 5s**

**# Path to the default home dashboard. If this value is empty, then Grafana uses StaticRootPath + "dashboards/home.json"**

**;default\_home\_dashboard\_path =**

**#################################### Users ###############################**

**[users]**

**# disable user signup / registration**

**;allow\_sign\_up = true**

**# Allow non admin users to create organizations**

**;allow\_org\_create = true**

**# Set to true to automatically assign new users to the default organization (id 1)**

**;auto\_assign\_org = true**

**# Set this value to automatically add new users to the provided organization (if auto\_assign\_org above is set to true)**

**;auto\_assign\_org\_id = 1**

**# Default role new users will be automatically assigned (if disabled above is set to true)**

**;auto\_assign\_org\_role = Viewer**

**# Require email validation before sign up completes**

**;verify\_email\_enabled = false**

**# Background text for the user field on the login page**

**;login\_hint = email or username**

**;password\_hint = password**

**# Default UI theme ("dark" or "light")**

**;default\_theme = dark**

**# Path to a custom home page. Users are only redirected to this if the default home dashboard is used. It should match a frontend route and contain a leading slash.**

**; home\_page =**

**# External user management, these options affect the organization users view**

**;external\_manage\_link\_url =**

**;external\_manage\_link\_name =**

**;external\_manage\_info =**

**# Viewers can edit/inspect dashboard settings in the browser. But not save the dashboard.**

**;viewers\_can\_edit = false**

**# Editors can administrate dashboard, folders and teams they create**

**;editors\_can\_admin = false**

**# The duration in time a user invitation remains valid before expiring. This setting should be expressed as a duration. Examples: 6h (hours), 2d (days), 1w (week). Default is 24h (24 hours). The minimum supported duration is 15m (15 minutes).**

**;user\_invite\_max\_lifetime\_duration = 24h**

**# Enter a comma-separated list of users login to hide them in the Grafana UI. These users are shown to Grafana admins and themselves.**

**; hidden\_users =**

**[auth]**

**# Login cookie name**

**;login\_cookie\_name = grafana\_session**

**# The maximum lifetime (duration) an authenticated user can be inactive before being required to login at next visit. Default is 7 days (7d). This setting should be expressed as a duration, e.g. 5m (minutes), 6h (hours), 10d (days), 2w (weeks), 1M (month). The lifetime resets at each successful token rotation.**

**;login\_maximum\_inactive\_lifetime\_duration =**

**# The maximum lifetime (duration) an authenticated user can be logged in since login time before being required to login. Default is 30 days (30d). This setting should be expressed as a duration, e.g. 5m (minutes), 6h (hours), 10d (days), 2w (weeks), 1M (month).**

**;login\_maximum\_lifetime\_duration =**

**# How often should auth tokens be rotated for authenticated users when being active. The default is each 10 minutes.**

**;token\_rotation\_interval\_minutes = 10**

**# Set to true to disable (hide) the login form, useful if you use OAuth, defaults to false**

**;disable\_login\_form = false**

**# Set to true to disable the signout link in the side menu. useful if you use auth.proxy, defaults to false**

**;disable\_signout\_menu = false**

**# URL to redirect the user to after sign out**

**;signout\_redirect\_url =**

**# Set to true to attempt login with OAuth automatically, skipping the login screen.**

**# This setting is ignored if multiple OAuth providers are configured.**

**;oauth\_auto\_login = false**

**# OAuth state max age cookie duration in seconds. Defaults to 600 seconds.**

**;oauth\_state\_cookie\_max\_age = 600**

**# limit of api\_key seconds to live before expiration**

**;api\_key\_max\_seconds\_to\_live = -1**

**# Set to true to enable SigV4 authentication option for HTTP-based datasources.**

**;sigv4\_auth\_enabled = false**

**#################################### Anonymous Auth ######################**

**[auth.anonymous]**

**# enable anonymous access**

**;enabled = false**

**# specify organization name that should be used for unauthenticated users**

**;org\_name = Main Org.**

**# specify role for unauthenticated users**

**;org\_role = Viewer**

**# mask the Grafana version number for unauthenticated users**

**;hide\_version = false**

**#################################### GitHub Auth ##########################**

**[auth.github]**

**;enabled = false**

**;allow\_sign\_up = true**

**;client\_id = some\_id**

**;client\_secret = some\_secret**

**;scopes = user:email,read:org**

**;auth\_url = https://github.com/login/oauth/authorize**

**;token\_url = https://github.com/login/oauth/access\_token**

**;api\_url = https://api.github.com/user**

**;allowed\_domains =**

**;team\_ids =**

**;allowed\_organizations =**

**#################################### GitLab Auth #########################**

**[auth.gitlab]**

**;enabled = false**

**;allow\_sign\_up = true**

**;client\_id = some\_id**

**;client\_secret = some\_secret**

**;scopes = api**

**;auth\_url = https://gitlab.com/oauth/authorize**

**;token\_url = https://gitlab.com/oauth/token**

**;api\_url = https://gitlab.com/api/v4**

**;allowed\_domains =**

**;allowed\_groups =**

**#################################### Google Auth ##########################**

**[auth.google]**

**;enabled = false**

**;allow\_sign\_up = true**

**;client\_id = some\_client\_id**

**;client\_secret = some\_client\_secret**

**;scopes = https://www.googleapis.com/auth/userinfo.profile https://www.googleapis.com/auth/userinfo.email**

**;auth\_url = https://accounts.google.com/o/oauth2/auth**

**;token\_url = https://accounts.google.com/o/oauth2/token**

**;api\_url = https://www.googleapis.com/oauth2/v1/userinfo**

**;allowed\_domains =**

**;hosted\_domain =**

**#################################### Grafana.com Auth ####################**

**[auth.grafana\_com]**

**;enabled = false**

**;allow\_sign\_up = true**

**;client\_id = some\_id**

**;client\_secret = some\_secret**

**;scopes = user:email**

**;allowed\_organizations =**

**#################################### Azure AD OAuth #######################**

**[auth.azuread]**

**;name = Azure AD**

**;enabled = false**

**;allow\_sign\_up = true**

**;client\_id = some\_client\_id**

**;client\_secret = some\_client\_secret**

**;scopes = openid email profile**

**;auth\_url = https://login.microsoftonline.com/<tenant-id>/oauth2/v2.0/authorize**

**;token\_url = https://login.microsoftonline.com/<tenant-id>/oauth2/v2.0/token**

**;allowed\_domains =**

**;allowed\_groups =**

**#################################### Okta OAuth #######################**

**[auth.okta]**

**;name = Okta**

**;enabled = false**

**;allow\_sign\_up = true**

**;client\_id = some\_id**

**;client\_secret = some\_secret**

**;scopes = openid profile email groups**

**;auth\_url = https://<tenant-id>.okta.com/oauth2/v1/authorize**

**;token\_url = https://<tenant-id>.okta.com/oauth2/v1/token**

**;api\_url = https://<tenant-id>.okta.com/oauth2/v1/userinfo**

**;allowed\_domains =**

**;allowed\_groups =**

**;role\_attribute\_path =**

**#################################### Generic OAuth ##########################**

**[auth.generic\_oauth]**

**;enabled = false**

**;name = OAuth**

**;allow\_sign\_up = true**

**;client\_id = some\_id**

**;client\_secret = some\_secret**

**;scopes = user:email,read:org**

**;email\_attribute\_name = email:primary**

**;email\_attribute\_path =**

**;login\_attribute\_path =**

**;name\_attribute\_path =**

**;id\_token\_attribute\_name =**

**;auth\_url = https://foo.bar/login/oauth/authorize**

**;token\_url = https://foo.bar/login/oauth/access\_token**

**;api\_url = https://foo.bar/user**

**;allowed\_domains =**

**;team\_ids =**

**;allowed\_organizations =**

**;role\_attribute\_path =**

**;tls\_skip\_verify\_insecure = false**

**;tls\_client\_cert =**

**;tls\_client\_key =**

**;tls\_client\_ca =**

**#################################### Basic Auth ##########################**

**[auth.basic]**

**;enabled = true**

**#################################### Auth Proxy ##########################**

**[auth.proxy]**

**;enabled = false**

**;header\_name = X-WEBAUTH-USER**

**;header\_property = username**

**;auto\_sign\_up = true**

**;sync\_ttl = 60**

**;whitelist = 192.168.1.1, 192.168.2.1**

**;headers = Email:X-User-Email, Name:X-User-Name**

**# Read the auth proxy docs for details on what the setting below enables**

**;enable\_login\_token = false**

**#################################### Auth LDAP ##########################**

**[auth.ldap]**

**;enabled = false**

**;config\_file = /etc/grafana/ldap.toml**

**;allow\_sign\_up = true**

**# LDAP background sync (Enterprise only)**

**# At 1 am every day**

**;sync\_cron = "0 0 1 \* \* \*"**

**;active\_sync\_enabled = true**

**#################################### AWS ###########################**

**[aws]**

**# Enter a comma-separated list of allowed AWS authentication providers.**

**# Options are: default (AWS SDK Default), keys (Access && secret key), credentials (Credentials field), ec2\_iam\_role (EC2 IAM Role)**

**; allowed\_auth\_providers = default,keys,credentials**

**# Allow AWS users to assume a role using temporary security credentials.**

**# If true, assume role will be enabled for all AWS authentication providers that are specified in aws\_auth\_providers**

**; assume\_role\_enabled = true**

**#################################### SMTP / Emailing ##########################**

**[smtp]**

**enabled = true**

**host = email-smtp.ap-south-1.amazonaws.com:587**

**user = AKIA6IP74RHP2KRYOW5M**

**# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""**

**password = BBZyWsE6wP9AVMng59FVQWBMVJ6i7ToDie4ppC0iVpRl**

**;cert\_file =**

**;key\_file =**

**skip\_verify = true**

**from\_address = curldom@creditmantri.com**

**;from\_name = Grafana**

**# EHLO identity in SMTP dialog (defaults to instance\_name)**

**;ehlo\_identity = dashboard.example.com**

**# SMTP startTLS policy (defaults to 'OpportunisticStartTLS')**

**;startTLS\_policy = NoStartTLS**

**[emails]**

**;welcome\_email\_on\_sign\_up = false**

**;templates\_pattern = emails/\*.html**

**#################################### Logging ##########################**

**[log]**

**# Either "console", "file", "syslog". Default is console and file**

**# Use space to separate multiple modes, e.g. "console file"**

**;mode = console file**

**# Either "debug", "info", "warn", "error", "critical", default is "info"**

**;level = info**

**# optional settings to set different levels for specific loggers. Ex filters = sqlstore:debug**

**;filters =**

**# For "console" mode only**

**[log.console]**

**;level =**

**# log line format, valid options are text, console and json**

**;format = console**

**# For "file" mode only**

**[log.file]**

**;level =**

**# log line format, valid options are text, console and json**

**;format = text**

**# This enables automated log rotate(switch of following options), default is true**

**;log\_rotate = true**

**# Max line number of single file, default is 1000000**

**;max\_lines = 1000000**

**# Max size shift of single file, default is 28 means 1 << 28, 256MB**

**;max\_size\_shift = 28**

**# Segment log daily, default is true**

**;daily\_rotate = true**

**# Expired days of log file(delete after max days), default is 7**

**;max\_days = 7**

**[log.syslog]**

**;level =**

**# log line format, valid options are text, console and json**

**;format = text**

**# Syslog network type and address. This can be udp, tcp, or unix. If left blank, the default unix endpoints will be used.**

**;network =**

**;address =**

**# Syslog facility. user, daemon and local0 through local7 are valid.**

**;facility =**

**# Syslog tag. By default, the process' argv[0] is used.**

**;tag =**

**[log.frontend]**

**# Should Sentry javascript agent be initialized**

**;enabled = false**

**# Sentry DSN if you want to send events to Sentry.**

**;sentry\_dsn =**

**# Custom HTTP endpoint to send events captured by the Sentry agent to. Default will log the events to stdout.**

**;custom\_endpoint = /log**

**# Rate of events to be reported between 0 (none) and 1 (all), float**

**;sample\_rate = 1.0**

**# Requests per second limit enforced an extended period, for Grafana backend log ingestion endpoint (/log).**

**;log\_endpoint\_requests\_per\_second\_limit = 3**

**# Max requests accepted per short interval of time for Grafana backend log ingestion endpoint (/log).**

**;log\_endpoint\_burst\_limit = 15**

**#################################### Usage Quotas ########################**

**[quota]**

**; enabled = false**

**#### set quotas to -1 to make unlimited. ####**

**# limit number of users per Org.**

**; org\_user = 10**

**# limit number of dashboards per Org.**

**; org\_dashboard = 100**

**# limit number of data\_sources per Org.**

**; org\_data\_source = 10**

**# limit number of api\_keys per Org.**

**; org\_api\_key = 10**

**# limit number of orgs a user can create.**

**; user\_org = 10**

**# Global limit of users.**

**; global\_user = -1**

**# global limit of orgs.**

**; global\_org = -1**

**# global limit of dashboards**

**; global\_dashboard = -1**

**# global limit of api\_keys**

**; global\_api\_key = -1**

**# global limit on number of logged in users.**

**; global\_session = -1**

**#################################### Alerting ############################**

**[alerting]**

**# Disable alerting engine & UI features**

**;enabled = true**

**# Makes it possible to turn off alert rule execution but alerting UI is visible**

**;execute\_alerts = true**

**# Default setting for new alert rules. Defaults to categorize error and timeouts as alerting. (alerting, keep\_state)**

**;error\_or\_timeout = alerting**

**# Default setting for how Grafana handles nodata or null values in alerting. (alerting, no\_data, keep\_state, ok)**

**;nodata\_or\_nullvalues = no\_data**

**# Alert notifications can include images, but rendering many images at the same time can overload the server**

**# This limit will protect the server from render overloading and make sure notifications are sent out quickly**

**;concurrent\_render\_limit = 5**

**# Default setting for alert calculation timeout. Default value is 30**

**;evaluation\_timeout\_seconds = 30**

**# Default setting for alert notification timeout. Default value is 30**

**;notification\_timeout\_seconds = 30**

**# Default setting for max attempts to sending alert notifications. Default value is 3**

**;max\_attempts = 3**

**# Makes it possible to enforce a minimal interval between evaluations, to reduce load on the backend**

**;min\_interval\_seconds = 1**

**# Configures for how long alert annotations are stored. Default is 0, which keeps them forever.**

**# This setting should be expressed as a duration. Examples: 6h (hours), 10d (days), 2w (weeks), 1M (month).**

**;max\_annotation\_age =**

**# Configures max number of alert annotations that Grafana stores. Default value is 0, which keeps all alert annotations.**

**;max\_annotations\_to\_keep =**

**#################################### Annotations #########################**

**[annotations]**

**# Configures the batch size for the annotation clean-up job. This setting is used for dashboard, API, and alert annotations.**

**;cleanupjob\_batchsize = 100**

**[annotations.dashboard]**

**# Dashboard annotations means that annotations are associated with the dashboard they are created on.**

**# Configures how long dashboard annotations are stored. Default is 0, which keeps them forever.**

**# This setting should be expressed as a duration. Examples: 6h (hours), 10d (days), 2w (weeks), 1M (month).**

**;max\_age =**

**# Configures max number of dashboard annotations that Grafana stores. Default value is 0, which keeps all dashboard annotations.**

**;max\_annotations\_to\_keep =**

**[annotations.api]**

**# API annotations means that the annotations have been created using the API without any**

**# association with a dashboard.**

**# Configures how long Grafana stores API annotations. Default is 0, which keeps them forever.**

**# This setting should be expressed as a duration. Examples: 6h (hours), 10d (days), 2w (weeks), 1M (month).**

**;max\_age =**

**# Configures max number of API annotations that Grafana keeps. Default value is 0, which keeps all API annotations.**

**;max\_annotations\_to\_keep =**

**#################################### Explore #############################**

**[explore]**

**# Enable the Explore section**

**;enabled = true**

**#################################### Internal Grafana Metrics ##########################**

**# Metrics available at HTTP API Url /metrics**

**[metrics]**

**# Disable / Enable internal metrics**

**;enabled = true**

**# Graphite Publish interval**

**;interval\_seconds = 10**

**# Disable total stats (stat\_totals\_\*) metrics to be generated**

**;disable\_total\_stats = false**

**#If both are set, basic auth will be required for the metrics endpoint.**

**; basic\_auth\_username =**

**; basic\_auth\_password =**

**# Metrics environment info adds dimensions to the `grafana\_environment\_info` metric, which**

**# can expose more information about the Grafana instance.**

**[metrics.environment\_info]**

**#exampleLabel1 = exampleValue1**

**#exampleLabel2 = exampleValue2**

**# Send internal metrics to Graphite**

**[metrics.graphite]**

**# Enable by setting the address setting (ex localhost:2003)**

**;address =**

**;prefix = prod.grafana.%(instance\_name)s.**

**#################################### Grafana.com integration ##########################**

**# Url used to import dashboards directly from Grafana.com**

**[grafana\_com]**

**;url = https://grafana.com**

**#################################### Distributed tracing ############**

**[tracing.jaeger]**

**# Enable by setting the address sending traces to jaeger (ex localhost:6831)**

**;address = localhost:6831**

**# Tag that will always be included in when creating new spans. ex (tag1:value1,tag2:value2)**

**;always\_included\_tag = tag1:value1**

**# Type specifies the type of the sampler: const, probabilistic, rateLimiting, or remote**

**;sampler\_type = const**

**# jaeger samplerconfig param**

**# for "const" sampler, 0 or 1 for always false/true respectively**

**# for "probabilistic" sampler, a probability between 0 and 1**

**# for "rateLimiting" sampler, the number of spans per second**

**# for "remote" sampler, param is the same as for "probabilistic"**

**# and indicates the initial sampling rate before the actual one**

**# is received from the mothership**

**;sampler\_param = 1**

**# sampling\_server\_url is the URL of a sampling manager providing a sampling strategy.**

**;sampling\_server\_url =**

**# Whether or not to use Zipkin propagation (x-b3- HTTP headers).**

**;zipkin\_propagation = false**

**# Setting this to true disables shared RPC spans.**

**# Not disabling is the most common setting when using Zipkin elsewhere in your infrastructure.**

**;disable\_shared\_zipkin\_spans = false**

**#################################### External image storage ##########################**

**[external\_image\_storage]**

**# Used for uploading images to public servers so they can be included in slack/email messages.**

**# you can choose between (s3, webdav, gcs, azure\_blob, local)**

**;provider =**

**[external\_image\_storage.s3]**

**;endpoint =**

**;path\_style\_access =**

**;bucket =**

**;region =**

**;path =**

**;access\_key =**

**;secret\_key =**

**[external\_image\_storage.webdav]**

**;url =**

**;public\_url =**

**;username =**

**;password =**

**[external\_image\_storage.gcs]**

**;key\_file =**

**;bucket =**

**;path =**

**[external\_image\_storage.azure\_blob]**

**;account\_name =**

**;account\_key =**

**;container\_name =**

**[external\_image\_storage.local]**

**# does not require any configuration**

**[rendering]**

**# Options to configure a remote HTTP image rendering service, e.g. using https://github.com/grafana/grafana-image-renderer.**

**# URL to a remote HTTP image renderer service, e.g. http://localhost:8081/render, will enable Grafana to render panels and dashboards to PNG-images using HTTP requests to an external service.**

**;server\_url =**

**# If the remote HTTP image renderer service runs on a different server than the Grafana server you may have to configure this to a URL where Grafana is reachable, e.g. http://grafana.domain/.**

**;callback\_url =**

**# Concurrent render request limit affects when the /render HTTP endpoint is used. Rendering many images at the same time can overload the server,**

**# which this setting can help protect against by only allowing a certain amount of concurrent requests.**

**;concurrent\_render\_request\_limit = 30**

**[panels]**

**# If set to true Grafana will allow script tags in text panels. Not recommended as it enable XSS vulnerabilities.**

**;disable\_sanitize\_html = false**

**[plugins]**

**;enable\_alpha = false**

**;app\_tls\_skip\_verify\_insecure = false**

**# Enter a comma-separated list of plugin identifiers to identify plugins that are allowed to be loaded even if they lack a valid signature.**

**;allow\_loading\_unsigned\_plugins =**

**;marketplace\_url = https://grafana.com/grafana/plugins/**

**#################################### Grafana Image Renderer Plugin ##########################**

**[plugin.grafana-image-renderer]**

**# Instruct headless browser instance to use a default timezone when not provided by Grafana, e.g. when rendering panel image of alert.**

**# See ICU’s metaZones.txt (https://cs.chromium.org/chromium/src/third\_party/icu/source/data/misc/metaZones.txt) for a list of supported**

**# timezone IDs. Fallbacks to TZ environment variable if not set.**

**;rendering\_timezone =**

**# Instruct headless browser instance to use a default language when not provided by Grafana, e.g. when rendering panel image of alert.**

**# Please refer to the HTTP header Accept-Language to understand how to format this value, e.g. 'fr-CH, fr;q=0.9, en;q=0.8, de;q=0.7, \*;q=0.5'.**

**;rendering\_language =**

**# Instruct headless browser instance to use a default device scale factor when not provided by Grafana, e.g. when rendering panel image of alert.**

**# Default is 1. Using a higher value will produce more detailed images (higher DPI), but will require more disk space to store an image.**

**;rendering\_viewport\_device\_scale\_factor =**

**# Instruct headless browser instance whether to ignore HTTPS errors during navigation. Per default HTTPS errors are not ignored. Due to**

**# the security risk it's not recommended to ignore HTTPS errors.**

**;rendering\_ignore\_https\_errors =**

**# Instruct headless browser instance whether to capture and log verbose information when rendering an image. Default is false and will**

**# only capture and log error messages. When enabled, debug messages are captured and logged as well.**

**# For the verbose information to be included in the Grafana server log you have to adjust the rendering log level to debug, configure**

**# [log].filter = rendering:debug.**

**;rendering\_verbose\_logging =**

**# Instruct headless browser instance whether to output its debug and error messages into running process of remote rendering service.**

**# Default is false. This can be useful to enable (true) when troubleshooting.**

**;rendering\_dumpio =**

**# Additional arguments to pass to the headless browser instance. Default is --no-sandbox. The list of Chromium flags can be found**

**# here (https://peter.sh/experiments/chromium-command-line-switches/). Multiple arguments is separated with comma-character.**

**;rendering\_args =**

**# You can configure the plugin to use a different browser binary instead of the pre-packaged version of Chromium.**

**# Please note that this is not recommended, since you may encounter problems if the installed version of Chrome/Chromium is not**

**# compatible with the plugin.**

**;rendering\_chrome\_bin =**

**# Instruct how headless browser instances are created. Default is 'default' and will create a new browser instance on each request.**

**# Mode 'clustered' will make sure that only a maximum of browsers/incognito pages can execute concurrently.**

**# Mode 'reusable' will have one browser instance and will create a new incognito page on each request.**

**;rendering\_mode =**

**# When rendering\_mode = clustered you can instruct how many browsers or incognito pages can execute concurrently. Default is 'browser'**

**# and will cluster using browser instances.**

**# Mode 'context' will cluster using incognito pages.**

**;rendering\_clustering\_mode =**

**# When rendering\_mode = clustered you can define maximum number of browser instances/incognito pages that can execute concurrently..**

**;rendering\_clustering\_max\_concurrency =**

**# Limit the maximum viewport width, height and device scale factor that can be requested.**

**;rendering\_viewport\_max\_width =**

**;rendering\_viewport\_max\_height =**

**;rendering\_viewport\_max\_device\_scale\_factor =**

**# Change the listening host and port of the gRPC server. Default host is 127.0.0.1 and default port is 0 and will automatically assign**

**# a port not in use.**

**;grpc\_host =**

**;grpc\_port =**

**[enterprise]**

**# Path to a valid Grafana Enterprise license.jwt file**

**;license\_path =**

**[feature\_toggles]**

**# enable features, separated by spaces**

**;enable =**

**[date\_formats]**

**# For information on what formatting patterns that are supported https://momentjs.com/docs/#/displaying/**

**# Default system date format used in time range picker and other places where full time is displayed**

**;full\_date = YYYY-MM-DD HH:mm:ss**

**# Used by graph and other places where we only show small intervals**

**;interval\_second = HH:mm:ss**

**;interval\_minute = HH:mm**

**;interval\_hour = MM/DD HH:mm**

**;interval\_day = MM/DD**

**;interval\_month = YYYY-MM**

**;interval\_year = YYYY**

**# Experimental feature**

**;use\_browser\_locale = false**

**# Default timezone for user preferences. Options are 'browser' for the browser local timezone or a timezone name from IANA Time Zone database, e.g. 'UTC' or 'Europe/Amsterdam' etc.**

**;default\_timezone = browser**

**[expressions]**

**# Enable or disable the expressions functionality.**

**;enabled = true**

**Alertmanager.yml Alert Manager configuration :-**

**global:**

**resolve\_timeout: 5m**

**route:**

**group\_by: ['alertname']**

**group\_wait: 5m**

**group\_interval: 30m**

**repeat\_interval: 30m**

**receiver: 'hari.v@creditmantri.com'**

**receivers:**

**- name: 'hari.v@creditmantri.com'**

**email\_configs:**

**- to: 'cmalerting@creditmantri.com'**

**from: 'curldom@creditmantri.com'**

**smarthost: email-smtp.ap-south-1.amazonaws.com:587**

**auth\_username: 'AKIA6IP74RHP2KRYOW5M'**

**auth\_identity: 'AKIA6IP74RHP2KRYOW5M'**

**auth\_password: 'BBZyWsE6wP9AVMng59FVQWBMVJ6i7ToDie4ppC0iVpRl'**

**inhibit\_rules:**

**- source\_match:**

**severity: 'critical'**

**target\_match:**

**severity: 'warning'**

**equal: ['alertname', 'dev', 'instance']**

**Prometheus configuration :-- prometheus.yml**

**global:**

**scrape\_interval: 1s**

**evaluation\_interval: 1s**

**rule\_files:**

**- alert.rules.yml**

**# - "second.rules"**

**alerting:**

**alertmanagers:**

**- static\_configs:**

**- targets:**

**- localhost:9093**

**scrape\_configs:**

**- job\_name: prometheus**

**scrape\_interval: 5s**

**static\_configs:**

**- targets: [localhost:9090]**

**- job\_name: node\_exporter**

**scrape\_interval: 5s**

**static\_configs:**

**- targets: ['172.31.48.21:9100']**

**labels:**

**name: 'BuildMyCreditScore'**

**# - targets: ['172.31.48.11:9100']**

**# labels:**

**# name: 'Pre-Login-Frontend-Private'**

**#- targets: ['172.31.48.196:9100']**

**#labels:**

**#name: 'CMOL-Dashboard-Private'**

**# - targets: ['172.31.48.72:9100']**

**# labels:**

**#name: 'Clone-ekam-private'**

**#- targets: ['172.31.48.93:9100']**

**#labels:**

**# name: 'Prelogin-API-Private'**

**#- targets: ['172.31.48.228:9100']**

**#labels:**

**# name: 'Postlogin-offer-Private'**

**- targets: ['172.31.48.41:9100']**

**labels:**

**name: 'Regen-Utility'**

**# - targets: ['172.31.48.20:9100']**

**# labels:**

**# name: 'ProdMonES - Kibana'**

**- targets: ['172.31.49.62:9100']**

**labels:**

**name: 'DSA-Agent'**

**#- targets: ['172.31.0.224:9100']**

**# labels:**

**# name: 'Pass2-CHR-CMOL-Old-DoNOTStop'**

**- targets: ['172.31.0.7:9100']**

**labels:**

**name: 'Credence-Proxy'**

**# - targets: ['172.31.48.76:9100']**

**# labels:**

**#name: 'CRM-NFlow'**

**- targets: ['172.31.48.121:9100']**

**labels:**

**name: 'CMOL-Logger-Utility'**

**- targets: ['172.31.0.227:9100']**

**labels:**

**name: 'BOBSME API - CRMAPI - SMELDB'**

**- job\_name: WMI\_exporter**

**scrape\_interval: 5s**

**static\_configs:**

**- targets: ['172.31.16.131:9182']**

**labels:**

**name: 'windows\_sftp'**

**Aleart call file in promeheus :- alert.rules.yml**

**groups:**

**- name: alert.rules**

**rules:**

**- alert: InstanceDown**

**expr: up == 0**

**for: 15s**

**labels:**

**severity: critical**

**annotations:**

**summary: "Endpoint {{ $labels.name }} down"**

**description: "{{ $labels.name }} has been down for more than 15s seconds."**

**- alert: HostOutOfMemory**

**expr: node\_memory\_MemAvailable\_bytes / node\_memory\_MemTotal\_bytes \* 100 < 20**

**for: 2m**

**labels:**

**severity: warning**

**annotations:**

**summary: Host out of memory (instance {{ $labels.name }})**

**description: "Node memory is filling up (< 20% left)\n VALUE = {{ $value }}\n LABELS = {{ $labels }}"**

**- alert: HostOutOfMemory**

**expr: node\_memory\_MemAvailable\_bytes / node\_memory\_MemTotal\_bytes \* 100 < 10**

**for: 2m**

**labels:**

**severity: critical**

**annotations:**

**summary: Host out of memory (instance {{ $labels.name }})**

**description: "Node memory is filling up (< 10% left)\n VALUE = {{ $value }}\n LABELS = {{ $labels }}"**

**- alert: HostOutOfDiskSpace**

**expr: (node\_filesystem\_avail\_bytes \* 100) / node\_filesystem\_size\_bytes < 20 and ON (instance, device, mountpoint) node\_filesystem\_readonly == 0**

**for: 2m**

**labels:**

**severity: warning**

**annotations:**

**summary: Host out of disk space (instance {{ $labels.name }})**

**description: "Disk is almost full (< 20% left)\n VALUE = {{ $value }}%\n LABELS = {{ $labels }}"**

**- alert: HostHighCpuLoad**

**expr: 100 - (avg by(instance) (rate(node\_cpu\_seconds\_total{mode="idle"}[2m])) \* 100) > 80**

**for: 0m**

**labels:**

**severity: warning**

**annotations:**

**summary: Host out of cpu (instance {{ $labels.instance }})**

**description: "CPU load is > 80%\n VALUE = {{ $value }}\n LABELS = {{ $labels }}"**

**- alert: HostHighCpuLoad**

**expr: 100 - (avg by(instance) (rate(node\_cpu\_seconds\_total{mode="idle"}[2m])) \* 100) > 90**

**for: 0m**

**labels:**

**severity: critical**

**annotations:**

**summary: Host out of cpu (instance {{ $labels.instance }})**

**description: "CPU load is > 90%\n VALUE = {{ $value }}\n LABELS = {{ $labels }}"**