

Install Promtail Binary and Start as a Service

Video Lecture



Download and Install Promtail Binary

Now we will create the Promtail service that will act as the collector for Loki.

We can also get the Promtail binary from the same place as Loki.

To check the latest version of Promtail, visit the Loki releases page. <https://github.com/grafana/loki/releases/>

```
cd /usr/local/bin
```

```
curl -O -L "https://github.com/grafana/loki/releases/download/v2.4.1/promtail-linux-amd64.zip"
```

```
unzip "promtail-linux-amd64.zip"
```

And allow the `execute` permission on the Promtail binary

```
sudo chmod a+x "promtail-linux-amd64"
```

Create the Promtail config

Now we will create the Promtail config file.

```
sudo nano config-promtail.yml
```

And add this script,

```
server:
```

```
http_listen_port: 9080
grpc_listen_port: 0

positions:
  filename: /tmp/positions.yaml

clients:
  - url: 'http://localhost:3100/loki/api/v1/push'

scrape_configs:
  - job_name: system
    static_configs:
      - targets:
          - localhost
        labels:
          job: varlogs
          __path__: /var/log/*log
```

This default configuration was copied from <https://raw.githubusercontent.com/grafana/loki/master/clients/cmd/promtail/promtail-local-config.yaml>. There may be changes to this config depending on any future updates to Loki.

Configure Promtail as a Service

Now we will configure Promtail as a service so that we can keep it running in the background.

Create user specifically for the Promtail service

```
sudo useradd --system promtail
```

Create a file called *promtail.service*

```
sudo nano /etc/systemd/system/promtail.service
```

And add this script,

```
[Unit]
Description=Promtail service
After=network.target

[Service]
Type=simple
User=promtail
ExecStart=/usr/local/bin/promtail-linux-amd64 -config.file /usr/local/bin/config-promtail.yml
```

```
[Install]
WantedBy=multi-user.target
```

Now start and check the service is running.

```
sudo service promtail start
sudo service promtail status
```

We can now leave the new Promtail service running.

Now, since this example uses Promtail to read system log files, the `promtail` user won't yet have permissions to read them.

So add the user `promtail` to the `adm` group

```
usermod -a -G adm promtail
```

Verify that the user is now in the `adm` group

```
id promtail
```

Restart Promtail and check status

```
sudo service promtail restart
sudo service promtail status
```

If you ever need to stop the new Promtail service, then type

```
sudo service promtail stop
sudo service promtail status
```

Note it may take a minute to stop.



Warning

If you reboot your server, the Promtail Service may not restart automatically.

You can set the Promtail service to auto restart after reboot by entering,

```
sudo systemctl enable promtail.service
```

Configure Firewall

Note

IPTables is a very precise science, to avoid problems setting up the exercises in the course, you can ignore these steps. However, if working in a production environment, ensure your firewalls are managed.

When your Promtail server is running, it will be accessible remotely. If you only want localhost to be able to connect, then type

```
iptables -A INPUT -p tcp -s localhost --dport 9080 -j ACCEPT
iptables -A INPUT -p tcp --dport 9080 -j DROP
iptables -L
```

Also, I have hard coded my Promtail service to use port 9097 for gRPC communications. This port may also be accessible across the internet. To close it using `iptables`, then use,

```
iptables -A INPUT -p tcp -s <your servers domain name or ip address> --dport 9097
-j ACCEPT
iptables -A INPUT -p tcp -s localhost --dport 9097 -j ACCEPT
iptables -A INPUT -p tcp --dport 9097 -j DROP
iptables -L
```

Note

If configuring the firewall to block port 9097 like I show in the video, be sure to run all 4 lines above. In the 1st line, you will use your own Grafana servers external IP address.

Be sure to back up your **iptables** rules if you installed **iptables-persistent** in the last section.

```
iptables-save > /etc/iptables/rules.v4
iptables-save > /etc/iptables/rules.v6
```

After blocking port 9080 for external requests, you can verify that local request still work using

```
curl "127.0.0.1:9080/metrics"
```

Warning

iptables settings will be lost in case of system reboot. You will need to reapply them manually, or

install **iptables-persistent**

```
sudo apt install iptables-persistent
```

This will save your settings into two files called,

```
/etc/iptables/rules.v4
```

```
/etc/iptables/rules.v6
```

Any changes you make to the **iptables** configuration won't be auto saved to these persistent files, so if you want to update these files with any changes, then use the commands,

```
iptables-save > /etc/iptables/rules.v4
```

```
iptables-save > /etc/iptables/rules.v6
```

Troubleshooting

Permission Denied

```
msg="error creating promtail" error="open /tmp/positions.yaml: permission denied"
```

You should check the owner of the file configured in the **positions** section of the **config-promtail.yml** matches the name of the user configured in the **promtail.service** script above.

My user is **promtail**, and the positions is set as **/tmp/positions.yaml**, so I set the owner.

```
chown promtail:promtail /tmp/positions.yaml
```

If you set up Promtail service with to run as a specific user, and you are using Promtail to view **adm** and you don't see any data in Grafana, but you can see the job name, then possibly you need to add the user Promtail to the **adm** group.

```
usermod -a -G adm promtail
```

You may need to restart the Promtail service and checking again its status.

Spaces versus Tabs

If you see the error "found character that cannot start any token" than that is likely to mean you have a tab somewhere in the YML indenting one of the tokens. Replace it with spaces.

Tail Promtail

On Linux, you can check the `syslog` for any Promtail related entries by using the command,

```
tail -f /var/log/syslog | grep promtail
```

Useful Links

[IPTables Cheat Sheet](#)

Grafana 10 and Ubuntu 22.04 Notes

Promtail v2.8.2

Since recording this video, Promtail is now at version 2.8.2. You can install this version if you wish.

```
cd /usr/local/bin
```

```
curl -O -L "https://github.com/grafana/loki/releases/download/v2.8.2/promtail-linux-amd64.zip"
```

```
unzip "promtail-linux-amd64.zip"
```

Now go back and continue from [Create The Promtail Config](#)

Daemons using outdated libraries

Since Ubuntu 22.04, installing new packages may give you a full screen prompt to restart other dependent services. Press `Tab` to highlight the `OK` option, and press `Enter`.

Unable to connect with Loki (Failed to call resource). Please check the server logs for more details.

If you have chosen to set up IPTables rules, then your problems may be related to that.

Ensure that the status of Loki and Promtail is running. If so, then try removing the firewall rules for port 9097.

Type

```
iptables -L --line-numbers
```

Then delete all rules, by line number, related to port 9097. I.e., this line below deletes line 5.

```
iptables -D INPUT 5
```

Run the above line several times to delete from the modified list.

Verify using

```
iptables -L
```

IPTables is a very precise science and while learning Grafana, you may find it a lot easier to ignore it. But be sure to manage your firewalls in Production using a dedicated firewall, and only use IPTables if you don't have a dedicated firewall solution already in place.

Below is a copy of my IPTables rules that you can use for reference. See that the `DROP` statements for each port are after the `ACCEPTs` for the same port.

```
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source                destination           tcp dpt:3000
ACCEPT     tcp  -- localhost             anywhere              tcp dpt:3000
DROP       tcp  -- anywhere              anywhere              tcp dpt:3000
ACCEPT     tcp  -- localhost             anywhere              tcp dpt:3100
DROP       tcp  -- anywhere              anywhere              tcp dpt:3100
ACCEPT     tcp  -- grafana               anywhere              tcp dpt:9097
ACCEPT     tcp  -- localhost             anywhere              tcp dpt:9097
DROP       tcp  -- anywhere              anywhere              tcp dpt:9097
```

Log Browser renamed to Label browser

The `Log browser` has now been renamed to the `[Label browser]`, and it is a button that opens a modal window.

