

Where I work, we use **Puppet** to manage our growing infrastructure of Linux machines and **Loki** together with **Grafana** to analyse log files.

When the Puppet agent runs on a host it creates something called a „Puppet config report“ which includes all steps taken by the Puppet agent to ensure the required configuration as specified in the Puppet catalog for that host. In our case, these reports are then sent back to **Foreman** as we’re using that on the Puppet server as a frontend.

To analyse these files better, we wanted to push these reports to Loki. Thing is, they are stored in the Foreman database in several tables (as good database design goes).

Here’s a small script to turn those reports into files which in turn can be scraped by **Promtail** or the likes to get them to Loki and analyse them using Grafana assuming you’re using **PostgreSQL** as your Foreman database:

```
#!/usr/bin/env bash

# Get report ids from today
REPORT_IDS=$(echo "select r.id from reports r where date_trunc('day', r.reported_at) = date_trunc('day', now())" | su - postgres -c "psql -A -t foreman")

for REPORT_ID in ${REPORT_IDS}
do
    REPORT_FILENAME=$(echo "select h.name || '-' || to_char(r.reported_at, 'YYYY-MM-DD\T\"HH24:MI:SS') || '.log' from reports r join hosts h on r.host_id=h.id where r.id=${REPORT_ID}" | su - postgres -c "psql -A -t foreman")
    cat <<EOT | su - postgres -c "psql -A -t foreman" > "${REPORT_FILENAME}"
WITH levels (id,level) as (values (2, 'notice'))
SELECT r.created_at || ' ' || '[' || levels.level || ']' || h.name || ' ' ||
s.value || ' ' || m.value
FROM messages m
JOIN logs l on m.id=l.message_id
JOIN sources s on s.id=l.source_id
JOIN levels on levels.id=l.level_id
JOIN reports r on l.report_id = r.id
JOIN hosts h on h.id = r.host_id
WHERE report_id=${REPORT_ID}
ORDER by l.id
EOT
done
```