



Nastavni predmet	KONFIGURIRANJE RAČUNALNIH MREŽA I SERVISA
Naslov jedinice	Vježba 15: Konfiguracija i upravljanje Prometheus monitoring sustavom na Linux Mintu

## CILJ VJEŽBE

Naučiti kako postaviti monitoring sustav za praćenje stanja servera i mreže koristeći Prometheus i Grafanu . Naučiti kako postaviti Prometheus za prikupljanje metrika i Grafanu za vizualizaciju podataka. Konfigurirati nadzor osnovnih resursa (CPU, memorija, disk, mreža) i postaviti obavijesti putem e-maila.

## IZVOĐENJE VJEŽBE

Pratite oprezno sve korake, pokažite profesoru izvedeno prema koracima u nastavku.

### Potrebni paketi

- **Prometheus:** Sustav za prikupljanje i pohranu metrika.
- **Node Exporter:** Agent za prikupljanje metrika s Linux sustava.
- **Grafana:** Alat za vizualizaciju metrika.
- **Alertmanager:** Alat za upravljanje obavijestima u Prometheusu.

## Koraci za izvođenje vježbe:

### 1. Inicijalne postavke

- 2) Pokrenuti VMware Workstation Pro
- 3) Postaviti snapshot u VMware-u za poslužitelj (Linux Mint) na **cleanInstall** kako bi započeli čistu konfiguraciju SNMP servisa
- 4) Pokrenuti Poslužitelj (Linux Mint)

User : **osboxes.org**

Password : **osboxes.org**

- 5) Na Poslužitelju (Linux Mint) u Network adapter postavkama odaberite **NAT** kako bi mogli pristupiti internetu kod instalacije Linux Mint paketa nadogradnje.
- 6) Na Poslužitelju (Linux Mint) postavite tipkovnicu na **Croatia (Layout)** zbog prilagodbe tipkovnice hrvatskoj verziji (ostale tipkovnice možete obrisati).

## 2. Inicijalne postavke

- 1) Ažurirajte repozitorije:

```
$ sudo apt update
```

- 2) Instaliraj potrebne alate:

```
$ sudo apt install -y wget curl
```

## 3. Instalacija i konfiguracija Prometheusa

- 1) Preuzmi i instaliraj Prometheus:

- Preuzmi najnoviju verziju Prometheusa s githuba:

```
$ wget
```

```
https://github.com/prometheus/prometheus/releases/download/v2.47.0/prometheus-2.47.0.linux-amd64.tar.gz //sve u jednom redu
```

- Raspakiraj arhivu:

```
$ tar xvfz prometheus-2.47.0.linux-amd64.tar.gz
```

- Premjesti Prometheus u `/opt` direktorij:

```
$ sudo mv prometheus-2.47.0.linux-amd64 /opt/prometheus
```

## 2) Konfiguriraj Prometheus

- Uredi konfiguracijsku datoteku:

```
$ sudo nano /opt/prometheus/prometheus.yml
```

- Dodaj sljedeće u datoteku (obriši prvo sve linije s Ctrl + K):

```
global:
```

```
  scrape_interval: 15s
```

```
scrape_configs:
```

```
  - job_name: 'node_exporter'
```

```
    static_configs:
```

```
      - targets: ['localhost:9100']
```

Snimate promjene i zatvorite (Ctrl+X, Y, Enter).

- Pokrenuti Prometheus kao pozadinski proces:

```
$ /opt/prometheus/prometheus --  
config.file=/opt/prometheus/prometheus.yml //u jednom redu
```

- Otvorite drugi Terminal i provjeri da li Prometheus radi:

```
$ ps aux | grep prometheus
```

Odgovor:

```
osboxes      2831  0.0  3.7 1316280 73844 pts/0    Sl   15:03   0:00  
/opt/prometheus/prometheusudo apt --  
config.file=/opt/prometheus/prometheus.yml  
osboxes      3347  0.0  0.1   9144   2176 pts/1    S+   15:19   0:00 grep  
--color=auto prometheus
```

- Provjeri da li Prometheus radi i u Firefox unesite:

```
http://localhost:9090
```

## 4. Instalacija i konfiguracija Node Exportera

### 1) Preuzmi i instaliraj Node Exporter:

- Preuzmi najnoviju verziju Node Exportera:

```
$ wget
```

```
https://github.com/prometheus/node_exporter/releases/download/v1.  
6.1/node_exporter-1.6.1.linux-amd64.tar.gz //sve u jednom redu
```

- Raspakiraj arhivu:

```
$ tar xvfz node_exporter-1.6.1.linux-amd64.tar.gz
```

- Premjesti Node Exporter u `/opt` direktorij:

```
$ sudo mv node_exporter-1.6.1.linux-amd64 /opt/node_exporter
```

## 2) Pokreni Node Exporter:

- Pokreni Node Exporter kao pozadinski proces:

```
$ /opt/node_exporter/node_exporter &
```

- Provjeri da li Node Exporter radi:

```
$ curl http://localhost:9100/metrics
```

Ako dobijete odgovor tipa:

```
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
```

```
# TYPE go_gc_duration_seconds summary
```

```
go_gc_duration_seconds{quantile="0"} 2.7221e-05
```

```
go_gc_duration_seconds{quantile="0.25"} 4.4374e-05
```

```
go_gc_duration_seconds{quantile="0.5"} 5.1167e-05
```

```
go_gc_duration_seconds{quantile="0.75"} 7.1965e-05
```

```
go_gc_duration_seconds{quantile="1"} 0.00157919
```

```
go_gc_duration_seconds_sum 0.016055933
```

```
go_gc_duration_seconds_count 184
```

```
# HELP go_goroutines Number of goroutines that currently exist.
```

```
# TYPE go_goroutines gauge
```

.....

Odgovor je ispravan!

## 5. Instalacija i konfiguracija Grafane

### 1) Instaliraj Grafanu:

- Dodaj Grafana repozitorij:

```
$ sudo apt-get install -y apt-transport-https
```

```
$ sudo apt-get install -y software-properties-common wget
```

```
$ wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-
```

```
key add - //jedan red
```

```
$ echo "deb https://packages.grafana.com/oss/deb stable main" |
```

```
sudo tee -a /etc/apt/sources.list.d/grafana.list //jedan red
```

- Ažurirajte repozitorije i popis paketa:

```
$ sudo apt update
```

- Instaliraj Grafanu:

```
$ sudo apt install grafana
```

## 2) Pokreni Grafanu:

- Pokreni Grafana servis:

```
$ sudo systemctl start grafana-server
```

```
$ sudo systemctl enable grafana-server
```

## 3) Provjeri status Grafana servisa:

```
$ sudo systemctl status grafana-server
```

## 4) Pristupi Grafana web sučelju:

- Otvori Firefox web preglednik i unesi za pristup:

```
http://localhost:3000
```


- Prijavi se koristeći zadane vjerodajnice:

**Korisničko ime:** admin

**Lozinka:** admin

Unesi novu lozinku ili ostavi istu za potrebe vježbe.

## 5) Dodaj Prometheus kao izvor podataka:

- Idi na (Home)  > **Connections** > **Search all:** Prometheus
- Klikni **Add data source**.
- Odaberi **Prometheus**.
- Unesi URL Prometheusa:  

```
http://localhost:9090
```
- Ostale postavke možete ostaviti na zadanim vrijednostima.
- Klikni **Save & Test**
- Trebali bi vidjeti poruku:  

```
Successfully queried the Prometheus API.
```
- Ova poruka potvrđuje da je Grafana uspješno povezana s Prometheusom.

# 6. Konfiguracija nadzora u Grafani

## 1) Kreiraj dashboard za nadzor:

- Na istoj stranici Grafane idi na > **Bild a dashboard** > **Add visualisation**
- **Data source:**
  - Odaberi **Prometheus** kao izvor podataka.

- Unesi upit za CPU korištenje. Unesite u [Metrics browser](#) (lijevo niže u browseru):

```
100 - (avg by (instance) (irate(node_cpu_seconds_total{mode="idle"}[1m])) * 100)
```

Konfiguriraj panel prema potrebama (npr. naslov, osi, boje)

Ponovi korake za druge metrike (memorija, disk, mreža).

## 2) Kreiraj upit za metriku memorije:

Pritisni **+ Add query** i dodaj u [Metrics browser](#):

```
node_memory_MemAvailable_bytes / node_memory_MemTotal_bytes * 100
```

## 3) Kreiraj upit za metriku diska:

Pritisni **+ Add query** i dodaj u [Metrics browser](#):

```
100 - (node_filesystem_avail_bytes{mountpoint="/"}/  
node_filesystem_size_bytes{mountpoint="/"}) * 100) // jedan red
```

## 4) Kreiraj upit za metriku mreže:

Pritisni **+ Add query** i dodaj u [Metrics browser](#):

```
rate(node_network_receive_bytes_total[1m])
```

# 7. Konfiguracija obavijesti (Alertmanager)

## 1) Instaliraj Alertmanager

- Preuzmi najnoviju verziju Alertmanager-a:

```
$ wget
```

```
https://github.com/prometheus/alertmanager/releases/download/v0.26.0/alertmanager-0.26.0.linux-amd64.tar.gz //jedan red
```

- Raspakiraj arhivu:

```
$ tar xvfz alertmanager-0.26.0.linux-amd64.tar.gz
```

- Premjesti Alertmanager u `/opt` direktorij:

```
$ sudo mv alertmanager-0.26.0.linux-amd64 /opt/alertmanager
```

## 2) Konfiguriraj Alertmanager:

- Uredi konfiguracijsku datotetku:

```
$ sudo nano /opt/alertmanager/alertmanager.yml
```

- Obriši sve s `Ctrl + K` i dodaj slijedeće u datoteku sa svojim kredencijalima:

```
global:
  smtp_smarthost: 'smtp.skole.hr:587'
  smtp_from: 'ime.prezime@skole.hr'
  smtp_auth_username: 'ime.prezime@skole.hr'
  smtp_auth_password: 'tvoja_školska_lozinka'
```

```
route:
  receiver: 'email-notifications'
```

```
receivers:
  - name: 'email-notifications'
    email_configs:
      - to: 'ime.prezime@skole.hr'
```

#### NAPOMENA:

ime.prezime@skole.hr je tvoja školska email adresa, te unosiš svoju CARNet školsku lozinku u datoteku na traženim mjestima.

Spremi promjene i zatvori datoteku (Ctrl+X, Y, Enter).

### 3) Pokreni Alertmanager:

```
$ /opt/alertmanager/alertmanager --
config.file=/opt/alertmanager/alertmanager.yml & //sve u jednom redu
```

### 4) Konfiguriraj obavijesti Prometheusu:

- Pokreni novi Terminal i slijedi dalje vježbu.
- Uredi Prometheus konfiguraciju:

```
$ sudo nano /opt/prometheus/prometheus.yml
```

- Obriši sve s Ctrl + K i dodaj sljedeće u datoteku:

```
global:
  scrape_interval: 15s # Interval prikupljanja metrika (svakih 15 sekundi)
  evaluation_interval: 15s # Interval evaluacije pravila za obavijesti

# Konfiguracija za prikupljanje metrika
scrape_configs:
  - job_name: 'node_exporter' # Naziv "posla" za prikupljanje metrika s
    node_exporter-a
    static_configs:
      - targets: ['localhost:9100'] # Adresa i port na kojem node_exp sluša
        labels:
```

```

    env: 'production' # Oznaka za okruženje (npr. production)
    role: 'monitoring' # Oznaka za ulogu (npr. monitoring, database)

- job_name: 'prometheus' # Naziv "posla" za prikupljanje metrika
  static_configs:
    - targets: ['localhost:9090'] # Adresa i port na kojem Prometheus sluša

# Konfiguracija za Alertmanager
alerting:
  alertmanagers:
    - static_configs:
      - targets: ['localhost:9093'] # Adresa/port na kojem Alertmanager sluša

# Datoteke s pravilima za obavijesti
rule_files:
  - 'alerts.yml' # Naziv datoteke s pravilima za obavijesti

```

Spremi promjene i zatvori datoteku (Ctrl+X, Y, Enter).

- Kreiraj `alerts.yml` datoteku. U datoteci se definiraju pravila za obavijesti koja će se aktivirati kada su određeni uvjeti ispunjeni (npr. visoka upotreba CPU-a, memorije ili diska).

```
$ sudo nano /opt/prometheus/alerts.yml
```

- Dodaj sljedeće u datoteku:

```

groups:
- name: node_alerts # Naziv grupe pravila za obavijesti
  rules:
    # Pravilo za visoku upotrebu CPU-a
    - alert: HighCpuUsage
      expr: 100 - (avg by (instance)
(irate(node_cpu_seconds_total{mode="idle"}[1m])) * 100 > 80 # CPU preko 80%
      for: 5m # Uvjet mora biti ispunjen 5 minuta prije slanja obavijesti
      labels:
        severity: critical # Oznaka ozbiljnosti (npr. critical, warning)
      annotations:
        summary: "High CPU usage on {{ $labels.instance }}" # Sažetak
obavijesti
        description: "CPU usage on {{ $labels.instance }} is above 80% for 5
minutes." # Detaljan opis

    # Pravilo za visoku upotrebu memorije
    - alert: HighMemoryUsage
      expr: (node_memory_MemTotal_bytes - node_memory_MemAvailable_bytes) /
node_memory_MemTotal_bytes * 100 > 80 # Memorija preko 80%
      for: 5m
      labels:
        severity: critical
      annotations:
        summary: "High memory usage on {{ $labels.instance }}"
        description: "Memory usage on {{ $labels.instance }} is above 80% for
5 minutes."

    # Pravilo za visoku upotrebu diska
    - alert: HighDiskUsage
      expr: 100 - (node_filesystem_avail_bytes{mountpoint="/" } /
node_filesystem_size_bytes{mountpoint="/" } * 100) > 80 # Disk preko 80%
      for: 5m
      labels:
        severity: critical

```



```

        annotations:
          summary: "High disk usage on {{ $labels.instance }}"
          description: "Disk usage on {{ $labels.instance }} is above 80% for 5
minutes."

# Pravilo za visoku mrežnu aktivnost
- alert: HighNetworkTraffic
  expr: rate(node_network_receive_bytes_total[1m]) > 100000000 # Mrežni
promet veći od 10 MB/s
  for: 5m
  labels:
    severity: warning
  annotations:
    summary: "High network traffic on {{ $labels.instance }}"
    description: "Network traffic on {{ $labels.instance }} is above 10
MB/s for 5 minutes."

```

Spremi promjene i zatvori datoteku (Ctrl+X, Y, Enter).

## 5) Ponovo pokreni Prometheus:

```

$ pkill prometheus
$ /opt/prometheus/prometheus --
config.file=/opt/prometheus/prometheus.yml & //jedan red

```

## 8. Izvođenje stres testova

Otvori novi Terminal.

### 1) Stres test za CPU:

- Instaliraj `stress` alat:

```
$ sudo apt install stress
```

- Pokreni stres test:

```
$ stress --cpu 4 --timeout 300
```

- Pratite stanje u Grafana nadzornoj ploči.

### 2) Stres test za memoriju:

- Pokreni stres test:

```
$ stress --vm-bytes $(awk '/MemAvailable/{printf "%d\n", $2 *
1024 * 0.9;}' /proc/meminfo) --vm-keep -m 1 //jedan red

```

- Pratite stanje u Grafana nadzornoj ploči.

### 3) Stres test za disk:

- Instalacija `stress-ng`:

```
$ sudo apt install stress-ng
```

- Pokreni stres test:

```
$ sudo stress-ng --hdd 1 --hdd-bytes 20M --timeout 300s
```

- Pratite stanje u Grafana nadzornoj ploči.

#### **Zaključak:**

Nakon ovih koraka, imat ćeš funkcionalan sustav za nadzor koristeći **Prometheus** i **Grafana**. Ova vježba omogućuje praćenje osnovnih resursa (CPU, memorija, disk, mreža) i slanje obavijesti putem e-maila.

**9. Ako čitate ovu rečenicu znači da ste uspješno završili vježbu i čeka vas opaka petica! Čestitamo na vašem trudu i napretku! Hvala! Postavite računalo na cleanInstall i Home.**