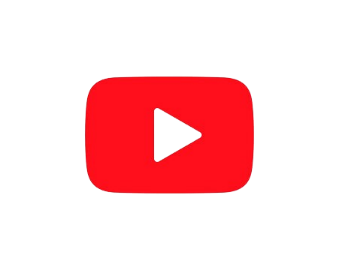
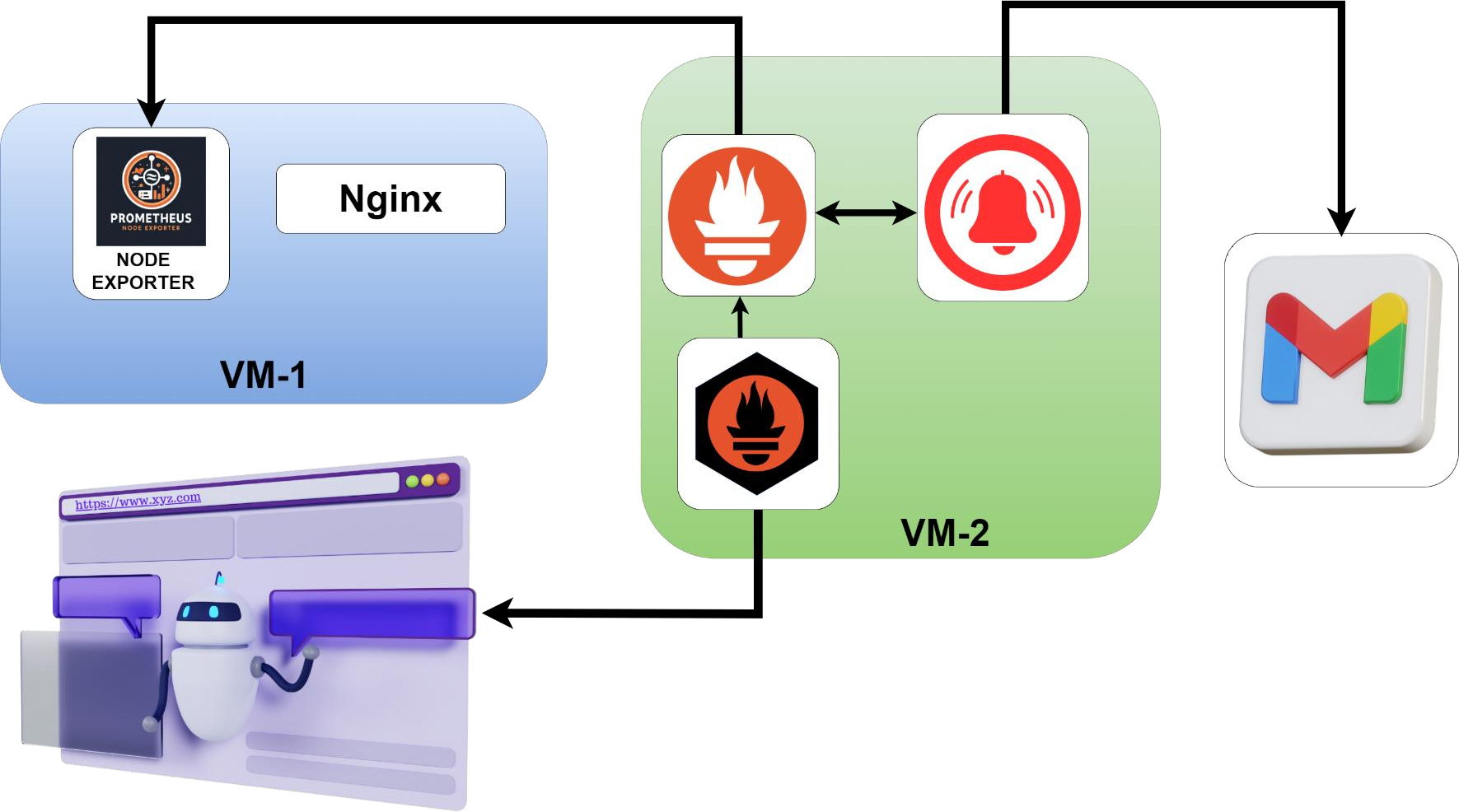
DevOps Shack



# Ultimate DevOps Monitoring Project

[**Click Here To Enrol To Batch-5 | DevOps & Cloud DevOps**](https://www.devopsshack.com/courses/Batch-5--Zero-To-Hero--DevSecOps--Cloud-DevOps-661d3ad119a57f6bb98a762f)



### Prerequisites

* Ensure you have wget and tar installed on both VMs.
* Ensure you have appropriate permissions to download, extract, and run these binaries.
* Replace <version> with the appropriate version number you wish to download.

### VM-1 (Node Exporter)

1. **Download Node Exporter**

wget https://github.com/prometheus/node\_exporter/releases/download/v1.8.1/ node\_exporter-1.8.1.linux-amd64.tar.gz

### Extract Node Exporter

tar xvfz node\_exporter-1.8.1.linux-amd64.tar.gz

### Start Node Exporter

1. cd node\_exporter-1.8.1.linux-amd64

./node\_exporter &

### VM-2 (Prometheus, Alertmanager, Blackbox Exporter)

Prometheus

### Download Prometheus

wget https://github.com/prometheus/prometheus/releases/download/v2.52.0/pr ometheus-2.52.0.linux-amd64.tar.gz

### Extract Prometheus

tar xvfz prometheus-2.52.0.linux-amd64.tar.gz

### Start Prometheus

1. cd prometheus-2.52.0.linux-amd64

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

Alertmanager

### Download Alertmanager

wget https://github.com/prometheus/alertmanager/releases/download/v0.27.0/ alertmanager-0.27.0.linux-amd64.tar.gz

### Extract Alertmanager

tar xvfz alertmanager-0.27.0.linux-amd64.tar.gz

### Start Alertmanager

1. cd alertmanager-0.27.0.linux-amd64

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

Blackbox Exporter

### Download Blackbox Exporter

wget https://github.com/prometheus/blackbox\_exporter/releases/download/v0. 25.0/blackbox\_exporter-0.25.0.linux-amd64.tar.gz

### Extract Blackbox Exporter

tar xvfz blackbox\_exporter-0.25.0.linux-amd64.tar.gz

### Start Blackbox Exporter

1. cd blackbox\_exporter-0.25.0.linux-amd64

./blackbox\_exporter &

## Notes:

* + The & at the end of each command ensures the process runs in the background.
  + Ensure that you have configured

the prometheus.yml and alertmanager.yml configuration files correctly before starting the services.

* + Adjust the firewall and security settings to allow the necessary ports (typically 9090 for Prometheus, 9093 for Alertmanager, 9115 for Blackbox Exporter, and 9100 for Node Exporter) to be accessible.

# Prometheus and Alertmanager Configuration

**Prometheus Configuration (prometheus.yml)**

## Global Configuration

global:

scrape\_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.

evaluation\_interval: 15s # Evaluate rules every 15 seconds.

The default is every 1 minute.

# scrape\_timeout is set to the global default (10s).

## Alertmanager Configuration

alerting: alertmanagers:

- static\_configs:

- targets:

- 'localhost:9093' # Alertmanager endpoint

## Rule Files

rule\_files:

- "alert\_rules.yml" # Path to alert rules file

# - "second\_rules.yml" # Additional rule files can be added here

## Scrape Configuration

### Prometheus Itself

scrape\_configs:

* job\_name: "prometheus" # Job name for Prometheus

# metrics\_path defaults to '/metrics' # scheme defaults to 'http'.

static\_configs:

* + targets: ["localhost:9090"] # Target to scrape (Prometheus itself)

### Node Exporter

* job\_name: "node\_exporter" # Job name for node exporter # metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

* + targets: ["3.110.195.114:9100"] # Target node exporter endpoint

### Blackbox Exporter

* job\_name: 'blackbox' # Job name for blackbox exporter metrics\_path: /probe # Path for blackbox probe params:

module: [http\_2xx] # Module to look for HTTP 200 response

static\_configs:

* + targets:
    - [http://prometheus.io](http://prometheus.io/) # HTTP target
    - https://prometheus.io # HTTPS target

- http://3.110.195.114:8080/ # HTTP target with port 8080 relabel\_configs:

* + source\_labels: [ address ] target\_label: param\_target
  + source\_labels: [ param\_target] target\_label: instance
  + target\_label: address

replacement: 13.235.248.225:9115 # Blackbox exporter address

**Alert Rules Configuration (alert\_rules.yml)**

## Alert Rules Group

groups:

- name: alert\_rules # Name of the alert rules group rules:

- alert: InstanceDown

expr: up == 0 # Expression to detect instance down for: 1m

labels:

severity: "critical" annotations:

summary: "Endpoint {{ $labels.instance }} down"

description: "{{ $labels.instance }} of job {{ $labels.job }} has been down for more than 1 minute."

- alert: WebsiteDown

expr: probe\_success == 0 # Expression to detect website down for: 1m

labels:

severity: critical annotations:

description: The website at {{ $labels.instance }} is down. summary: Website down

- alert: HostOutOfMemory

expr: node\_memory\_MemAvailable / node\_memory\_MemTotal \* 100 < 25 #

Expression to detect low memory for: 5m

labels:

severity: warning annotations:

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

description: "Node memory is filling up (< 25% left)\n VALUE = {{

$value }}\n LABELS: {{ $labels }}"

- alert: HostOutOfDiskSpace

expr: (node\_filesystem\_avail{mountpoint="/"} \* 100) / node\_filesystem\_size{mountpoint="/"} < 50 # Expression to detect low disk space

for: 1s labels:

severity: warning annotations:

summary: "Host out of disk space (instance {{ $labels.instance }})" description: "Disk is almost full (< 50% left)\n VALUE = {{ $value

}}\n LABELS: {{ $labels }}"

- alert: HostHighCpuLoad expr: (sum by (instance)

(irate(node\_cpu{job="node\_exporter\_metrics",mode="idle"}[5m]))) > 80 #

Expression to detect high CPU load for: 5m

labels:

severity: warning annotations:

summary: "Host high CPU load (instance {{ $labels.instance }})" description: "CPU load is > 80%\n VALUE = {{ $value }}\n LABELS:

{{ $labels }}"

- alert: ServiceUnavailable

expr: up{job="node\_exporter"} == 0 # Expression to detect service unavailability

for: 2m labels:

severity: critical annotations:

summary: "Service Unavailable (instance {{ $labels.instance }})" description: "The service {{ $labels.job }} is not available\n

VALUE = {{ $value }}\n LABELS: {{ $labels }}"

- alert: HighMemoryUsage

expr: (node\_memory\_Active / node\_memory\_MemTotal) \* 100 > 90 #

Expression to detect high memory usage for: 10m

labels:

severity: critical annotations:

summary: "High Memory Usage (instance {{ $labels.instance }})" description: "Memory usage is > 90%\n VALUE = {{ $value }}\n

LABELS: {{ $labels }}"

- alert: FileSystemFull

expr: (node\_filesystem\_avail / node\_filesystem\_size) \* 100 < 10 #

Expression to detect file system almost full for: 5m

labels:

severity: critical annotations:

summary: "File System Almost Full (instance {{ $labels.instance

}})"

description: "File system has < 10% free space\n VALUE = {{ $value

}}\n LABELS: {{ $labels }}"

**Alertmanager Configuration (alertmanager.yml)**

## Routing Configuration

route:

group\_by: ['alertname'] # Group by alert name

group\_wait: 30s # Wait time before sending the first notification

group\_interval: 5m # Interval between notifications

repeat\_interval: 1h # Interval to resend notifications receiver: 'email-notifications' # Default receiver

receivers:

- name: 'email-notifications' # Receiver name email\_configs:

- to: [jaiswaladi246@gmail.com](mailto:jaiswaladi246@gmail.com) # Email recipient from: [test@gmail.com](mailto:test@gmail.com) # Email sender smarthost: smtp.gmail.com:587 # SMTP server auth\_username: your\_email # SMTP auth username auth\_identity: your\_email # SMTP auth identity

auth\_password: "bdmq omqh vvkk zoqx" # SMTP auth password send\_resolved: true # Send notifications for resolved

alerts

## Inhibition Rules

inhibit\_rules:

- source\_match:

severity: 'critical' # Source alert severity target\_match:

severity: 'warning' # Target alert severity equal: ['alertname', 'dev', 'instance'] # Fields to match