

Laborübung

Monitoring

Dieses Dokument beinhaltet die Versuchsanleitung für die Durchführung der Laborübung Monitoring.

Bei Fragen zur Versuchsanleitung wenden Sie sich bitte direkt an das Laborpersonal.

Autoren: S. Renggli, P. Infanger

Version: 1.1

Letzte Änderung: 03. Oktober 2024

Änderungsverzeichnis

Version	Datum	Status	Änderungen und Bemerkungen	Bearbeitet von
1.0	30.09.2022	Final	Anpassen an Standard-NISLAB-Übung	S. Renggli
1.1	03.10.2022	Final	Ausgangslage & kleine Korrekturen	J. Keiser
1.2	30.09.2024	Final	Update auf neue Infrastruktur	S. Renggli
1.3	03.10.2024	Final	Update HS24 ITIA	F. Wamser

Inhaltsverzeichnis

Vorwort	3
Feedback	3
Legende	3
Bemerkungen / Rechtlicher Hinweis.....	3
1 Hausaufgaben	4
1.1 Theorie	4
1.2 Benötigte Mittel	4
1.3 Versuchsumgebung	4
2 Ausgangslage / Szenario	6
3 Monitoring mit Prometheus und Grafana.....	7
3.1 Installation von Prometheus	7
3.2 Konfiguration des Prometheus Service Files	10
3.3 Konfiguration der Prometheus Exporter	12
3.3.1 Konfiguration des Windows Exporter	12
3.3.2 Konfiguration des Node Exporter	15
3.4 Konfiguration von Prometheus	16
3.5 Installation von Grafana.....	20
3.6 Konfiguration von Grafana	21
3.7 Erste Prometheus Queries mit Grafana	23
3.8 Grafana Dashboard importieren	24
3.9 Erstellen eines eigenen Dashboards	26
3.10 Grafana Plugins	27

Vorwort

Feedback

Mit Ihrer Mithilfe kann die Qualität des Versuches laufend den Bedürfnissen angepasst und verbessert werden.

Falls in diesem Versuchsablauf etwas nicht so funktioniert wie es beschrieben ist, melden Sie dies bitte direkt dem Laborpersonal oder erwähnen Sie es in Ihrem Laborbericht oder Protokoll. Behandeln Sie die zur Verfügung gestellten Geräte mit der entsprechenden Umsicht.

Bei Problemen wenden Sie sich bitte ebenfalls an das Laborpersonal.

Legende

In den Versuchen gibt es Passagen, die mit den folgenden Zeichen markiert sind. Diese sind wie folgt zu verstehen:



Dringend beachten. Was hier steht, unbedingt merken oder ausführen.



Beantworten und dokumentieren Sie die Antworten im Laborprotokoll.



Ergänzender Hinweis / Notiz / Hilfestellung.



Weiterführende Informationen. Dies sind Informationen, die nicht zur Ausführung der Versuche benötigt werden, aber bekannt sein sollten.

Bemerkungen / Rechtlicher Hinweis

Die vorliegenden Übungen werden als Partnerarbeiten geführt. Es ist daher notwendig, vorgängig Zweierteams zu bilden.

1 Hausaufgaben

Dieses Kapitel beschreibt die Vorbereitungsmaßnahmen, die Sie vor Beginn des Laborversuches durchführen müssen.

1.1 Theorie

Die notwendige Theorie für diese Laborübung wurde vorgängig in einer Theorielektion vermittelt. Zusätzliche Informationen zu Prometheus finden Sie im Dokument «Prometheus Grundlagen».

Sehen Sie vor Beginn der Übung folgende Seite an: <https://prometheus.io/docs/introduction/overview/>

1.2 Benötigte Mittel

Im Rahmen dieser Laborübungen werden keine expliziten Hardwareressourcen benötigt. Sämtliche Aufgaben werden auf den Laborgeräten bzw. auf virtuellen Servern und Maschinen durchgeführt. Bei Bedarf kann auch das eigene Gerät herangezogen werden.

1.3 Versuchsumgebung

Die Studierenden erhalten zu Beginn der Übung folgende Arbeitsumgebung, welche auf Basis von virtuellen Maschinen im HSLU-Rechencluster umgesetzt ist:

ansible-XXX.itia.ls.eee.intern: Virtuelle Maschine mit OS Debian. Für spätere Übungen ist Ansible und VSCode Web vorinstalliert. Während der Übung wird hier Prometheus und Grafana installiert.

debian-XXX.itia.ls.eee.intern: Virtuelle Maschine mit OS Debian. Während der Übung wird hier ein Prometheus Exporter installiert.

windows-XXX.itia.ls.eee.intern: Virtuelle Maschine mit OS Windows Server. Während der Übung wird hier ein Prometheus Exporter installiert.

Benutzer: labadmin

Passwort: stylish@CURRENT4scuff

Auf ILIAS steht die Zuordnungsliste bereit: https://hsluzern-my.sharepoint.com/:x/g/personal/florian_wamser_hslu_ch/ESAmfVckjXBIgkxdBmAPV3IB9YqieGdr8pigplFueHzDqA?e=pWrTZ1

HSLU Hochschule Luzern ILIAS HSLU

Magazin > Informatik > Ausbildung > 2024 > I.BA_ITIA.H2401

Dashboard

Streaming (Zoom)

Zoom-Link

Informationen und Unterlagen zum Modul

Unterlagen zum Unterricht
Foliensätze, Übungen/Labore, Vorlesungsaufzeichnungen, Materialien

Wochenplan - HS24 - v2 - 20240920
pdf 180.79 KB Version: 6 20. Sep 2024, 16:01

Modulbeschreibung - HS24 - v20240610
pdf 152.76 KB Version: 2 27. Aug 2024 10:45 Anzahl Seiten: 3

Zuordnungen LAB-VMs zu Studenten (v20241003)

Leistungsnachweise und Kommunikation

Ankündigungsforum

The screenshot shows a Microsoft Excel spreadsheet titled "20241003_ITIA_Zuweisung" in a web browser. The spreadsheet contains a table with the following data:

Index	Hostname	Student
1	ansible-001.itia.ls.eee.intern	Affentranger, Andreas (ibaffent)
	debian-001.itia.ls.eee.intern	
	windows-001.itia.ls.eee.intern	
2	ansible-002.itia.ls.eee.intern	Alig, Nicola (iaalig)
	debian-002.itia.ls.eee.intern	
	windows-002.itia.ls.eee.intern	
3	ansible-003.itia.ls.eee.intern	Bühler, Dominik (iabuehle)
	debian-003.itia.ls.eee.intern	
	windows-003.itia.ls.eee.intern	
4	ansible-004.itia.ls.eee.intern	Corti, Aurel (iacorti)
	debian-004.itia.ls.eee.intern	
	windows-004.itia.ls.eee.intern	
5	ansible-005.itia.ls.eee.intern	Dalic, Pere (iadalic)
	debian-005.itia.ls.eee.intern	

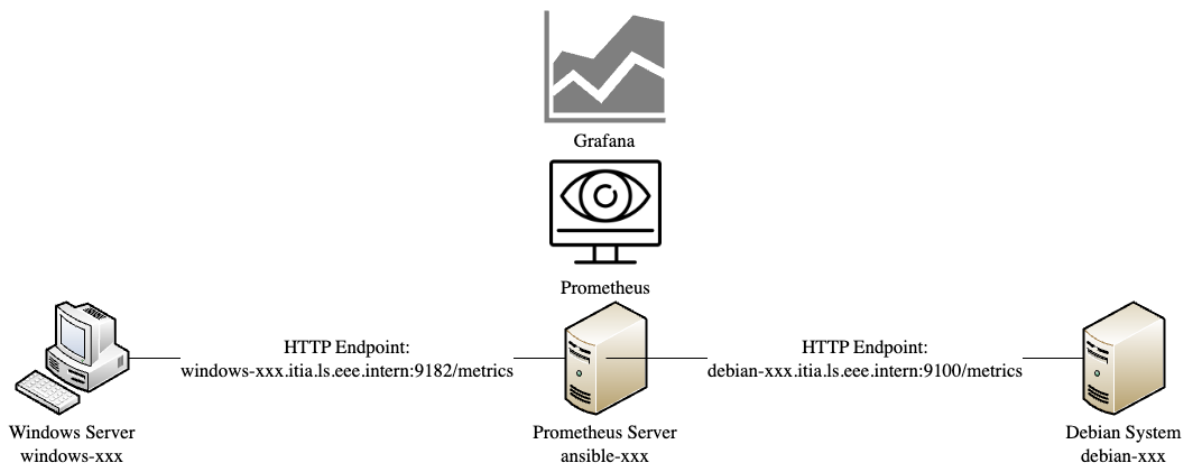
Additional text in the spreadsheet includes a title "Zuweisungsliste" and instructions: "Jeder Student erhält drei virtuelle Maschinen. Diese lauten ansible-XXX.itia.ls.eee.intern, debian-XXX.itia.ls.eee.intern und windows-XXX.itia.ls.eee.intern. Der Benutzername lautet: labadmin. Das Passwort lautet: stylish@CURRENT4scuff."

2 Ausgangslage / Szenario

In diesem Abschnitt wird die Ausgangslage beziehungsweise das Szenario, auf welchem die Laborübung aufbaut, beschrieben und definiert.

Wie im vorherigen Abschnitt beschrieben, besteht die Versuchsumgebung für diesen Laborversuch aus drei virtuellen Maschinen. Zwei der VMs führen Prometheus Exporter aus, welche Metriken über das jeweilige System sammeln und über einen HTTP-Endpoint zur Verfügung stellen. Weil es sich dabei um zwei verschiedene Betriebssysteme handelt, werden unterschiedliche Ausführungen des Exporters eingesetzt. Auf dem Debian 11 System soll «Node Exporter» eingesetzt werden, während auf dem Windows Server der «Windows Exporter» installiert wird.

Eine weitere VM führt den Prometheus Server aus, welcher die von den Exportern bereitgestellten Metriken sammelt. Mithilfe der Abfragesprache PromQL können die Metriken selektieren und aggregiert werden. Grafana dient dazu die aufbereiteten Daten in Panels zu visualisieren welche auf einem Dashboard angeordnet sind.



3 Monitoring mit Prometheus und Grafana

In diesem Kapitel wird die Thematik Monitoring anhand der beiden Open-Source Tools Prometheus und Grafana behandelt. Sie werden sehen, wie Prometheus und Grafana sowie Prometheus Exporter installiert und konfiguriert werden können.

3.1 Installation von Prometheus

Auf der Website von Prometheus (<https://prometheus.io/docs/prometheus/latest/installation/>) stehen verschiedene Installationsmöglichkeiten zur Verfügung. Wir entscheiden uns in dieser Übung dafür, vorkompilierte Binaries zu verwenden, um Prometheus zu installieren. Wenn Sie sich die Download Sektion auf der Webseite ansehen (<https://prometheus.io/download/>) stellen Sie fest, dass Prometheus Downloads als .zip (Windows) und als .tar.gz (Linux/macOS) zur Verfügung stehen.

Da kein .deb oder ähnliche Installer zur Verfügung stehen, muss die Installation von Prometheus von Hand vorgenommen werden. Nachfolgend werden Sie einige Vorbereitungsschritte durchführen.

Verbinden Sie sich als erstes mit dem SSH-Client ihrer Wahl mit der VM (ersetzen Sie dabei xxx mit ihrer Gruppennummer).

```

labadmin@ansible-001: ~
flo~ ssh ansible-001.itia.ls.eee.intern -l labadmin [13:11:16]
labadmin@ansible-001.itia.ls.eee.intern's password:
Warning: untrusted X11 forwarding setup failed: xauth key data not generated

Debian GNU/Linux 12 (bookworm) (6.1.0-23-amd64)

As of: Thu Oct 3 13:11:26 CEST 2024

System load: 0.00 Processes: 130
Memory usage: 0.0% Uptime: 22 days
Disk usage: 20%

  _ _ _ _ _
 | | / _ _ | | | |
 | | \ _ _ \ | | | |
 | _ | _ _ ) | | | |
 | | | | _ _ / | _ _ \ _ _
 | | _ _ | | / _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 | | / _ _ ' _ _ \ _ _ \ / _ _ ' _ _ \ / _ _ \ / _ _ \ / _ _ \
 | _ _ ( | | _ _ ) | _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
 | _ _ \ _ _ , _ _ _ / | _ _ _ \ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Last login: Thu Oct 3 12:21:34 2024 from 10.146.88.37
labadmin@ansible-001:~$


```

Damit Prometheus unter einem eigenen Benutzer läuft, erstellen Sie mit folgenden Befehlen eine Gruppe und ein Benutzer «prometheus»:

```
sudo groupadd --system prometheus
```

Mit der Option «--system» wird spezifiziert, dass eine Systemgruppe angelegt werden soll. Die GID der Gruppe «prometheus» wird also kleiner als 1000 sein.

```
sudo useradd -s /sbin/nologin --system -g prometheus prometheus
```

 Erklären Sie kurz, was obenstehender Befehl bewirkt. Achten Sie dabei speziell auf die Option «-s /sbin/nologin».

.....
.....

Nachdem Sie nun einen Benutzer für Prometheus angelegt haben, werden Sie einige benötigte Verzeichnisse für Prometheus Daten und Konfigurationsdateien erstellen.

In der Standardkonfiguration liegt die Zeitseriendatenbank von Prometheus unter /var/lib/prometheus. Mit folgendem Befehl erstellen wir das Verzeichnis.

```
sudo mkdir /var/lib/prometheus
```

Die Konfigurationsdateien von Prometheus befinden sich unter /etc/prometheus. Wir erstellen also dieses Verzeichnis und einige Unterverzeichnisse.

```
for i in rules rules.d files_sd; do sudo mkdir -p /etc/prometheus/${i}; done
```

Führen Sie folgende Befehle aus, um sicherzustellen, dass einige benötigte Tools installiert sind:

```
sudo apt update  
sudo apt -y install wget curl vim
```

Wir haben nun alle Vorbereitungen getroffen und können Prometheus herunterladen. Führen Sie dazu folgenden Befehl aus:

```
mkdir -p /tmp/prometheus && cd /tmp/prometheus  
curl -s https://api.github.com/repos/prometheus/prometheus/releases/latest | grep  
browser_download_url | grep linux-amd64 | cut -d '"' -f 4 | wget -qi -
```

Wenn Sie nun die Dateien im aktuellen Verzeichnis auflisten, stellen Sie fest, dass die neuste Version von Prometheus für Linux wie erwartet als .tar.gz heruntergeladen wurde.

```
labadmin@itia-ansible-dev-01:/tmp/prometheus$ ls -al  
total 81912  
drwxr-xr-x  2 labadmin labadmin   4096 Sep 30 13:30 .  
drwxrwxrwt 11 root      root     4096 Sep 30 13:30 ..  
-rw-r--r--  1 labadmin labadmin 83865783 Sep 12 15:13 prometheus-2.37.1.linux-amd64.tar.gz
```


Extrahieren Sie nun die .tar.gz Datei.

```
tar xvf prometheus*.tar.gz
cd prometheus*/
```

Nach dem Entpacken der Datei können sie die Inhalte einsehen. Für uns von Interesse sind vor allem die beiden Binaries «prometheus» und «promtool» sowie die Konfigurationsdatei «prometheus.yml».

```
labadmin@itia-ansible-dev-01:/tmp/prometheus/prometheus-2.37.1.linux-amd64$ ls -al
total 206268
drwxr-xr-x 4 labadmin labadmin    4096 Sep 12 15:08 .
drwxr-xr-x 3 labadmin labadmin    4096 Sep 30 13:47 ..
drwxr-xr-x 2 labadmin labadmin    4096 Sep 12 15:04 console_libraries
drwxr-xr-x 2 labadmin labadmin    4096 Sep 12 15:04 consoles
-rw-r--r-- 1 labadmin labadmin   11357 Sep 12 15:04 LICENSE
-rw-r--r-- 1 labadmin labadmin    3773 Sep 12 15:04 NOTICE
-rwxr-xr-x 1 labadmin labadmin 109681846 Sep 12 14:46 prometheus
-rw-r--r-- 1 labadmin labadmin     934 Sep 12 15:04 prometheus.yml
-rwxr-xr-x 1 labadmin labadmin 101497637 Sep 12 14:49 promtool
```

Die beiden Binaries werden ins Verzeichnis /usr/local/bin/ verschoben.

```
sudo mv prometheus promtool /usr/local/bin/
```

Starten Sie nun Prometheus um die Version zu überprüfen.

```
cd /usr/local/bin/
./prometheus --version
./promtool --version
```



Beachten Sie, dass sich die bei Ihnen dargestellte Version von den untenstehenden Abbildungen unterscheiden kann.

```
labadmin@itia-ansible-dev-01:/usr/local/bin$ ./prometheus --version
prometheus, version 2.37.1 (branch: HEAD, revision: 1ce2197e7f9e95089bfb95cb61762b5a89a8c0da)
 build user:      root@3caaaea7ba87
 build date:      20220912-12:42:39
 go version:      go1.18.6
 platform:        linux/amd64
```

```
labadmin@itia-ansible-dev-01:/usr/local/bin$ ./promtool --version
promtool, version 2.37.1 (branch: HEAD, revision: 1ce2197e7f9e95089bfb95cb61762b5a89a8c0da)
 build user:      root@3caaaea7ba87
 build date:      20220912-12:42:39
 go version:      go1.18.6
 platform:        linux/amd64
```

Verschieben Sie nun die Konfigurationsdatei von Prometheus «prometheus.yml» nach /etc/prometheus.



Mit «cd -» können Sie ins vorherige Verzeichnis wechseln. Falls Sie inzwischen andere Verzeichnisse aufgerufen haben, stellen Sie sicher, dass Sie sich wieder im korrekten Verzeichnis befinden.

```
cd -  
sudo mv prometheus.yml /etc/prometheus/prometheus.yml
```

Die Console Libraries werden ebenfalls nach /etc/prometheus verschoben. Anschliessend wechseln Sie zurück in Ihr Home-Verzeichnis.

```
sudo mv consoles/ console_libraries/ /etc/prometheus/  
cd ~
```

Zum Schluss passen Sie noch Owner, Gruppe und Berechtigungen der erstellten Verzeichnisse und den verschobenen Binaries an.

```
sudo chown -R prometheus:prometheus /etc/prometheus/  
for i in rules rules.d files sd; do sudo chmod -R 775 /etc/prometheus/${i}; done  
sudo chown -R prometheus:prometheus /var/lib/prometheus/
```

Damit ist die Installation von Prometheus abgeschlossen. Im nächsten Kapitel wird ein Service für Prometheus erstellt.

3.2 Konfiguration des Prometheus Service Files

Damit Prometheus direkt beim Systemstart gestartet wird und immer im Hintergrund ausgeführt wird, erstellen wir in diesem Kapitel einen Systemd Service, welcher Prometheus startet.

Mit folgendem Befehl erstellen Sie den systemd Service.

```
sudo tee /etc/systemd/system/prometheus.service<<EOF  
[Unit]  
Description=Prometheus  
Documentation=https://prometheus.io/docs/introduction/overview/  
Wants=network-online.target  
After=network-online.target  
  
[Service]  
Type=simple  
User=prometheus  
Group=prometheus  
ExecReload=/bin/kill -HUP $MAINPID  
ExecStart=/usr/local/bin/prometheus \\  
--config.file=/etc/prometheus/prometheus.yml \\  
--storage.tsdb.path=/var/lib/prometheus \\  
--web.console.templates=/etc/prometheus/consoles \\  
--web.console.libraries=/etc/prometheus/console_libraries \\  
--web.listen-address=0.0.0.0:9090  
  
SyslogIdentifier=prometheus  
Restart=always  
RestartSec=10  
  
[Install]  
WantedBy=multi-user.target
```

EOF

Studieren Sie wiederum die Konfigurationsoptionen von Prometheus im Unit File. Nehmen Sie dabei die offizielle Dokumentation zur Hand. Auch die Hilfe der Prometheus Binary gibt Auskunft über die CLI Parameter.

Sie können die Hilfe mit folgendem Befehl aufrufen.

```
prometheus --help
```



Zu welchem Zeitpunkt beim Systemstart wird Prometheus gestartet?

.....

.....



*Was bewirkt der Prometheus CLI-Parameter «--web.listen-address=0.0.0.0:9090»?
Scheint Ihnen diese Konfiguration sinnvoll? Begründen Sie.*

.....

.....

.....

.....

Damit das neue Service File vom System erkannt wird muss die System Manager Konfiguration neu geladen werden.

```
sudo systemctl daemon-reload
```

Zum Abschluss der Konfiguration des Prometheus Services wird der Service aktiviert und gestartet. Prometheus wird damit bei jedem Systemstart automatisch gestartet.

```
sudo systemctl start prometheus  
sudo systemctl enable prometheus
```

Den Status des Prometheus Services können Sie mit folgendem Befehl überprüfen.

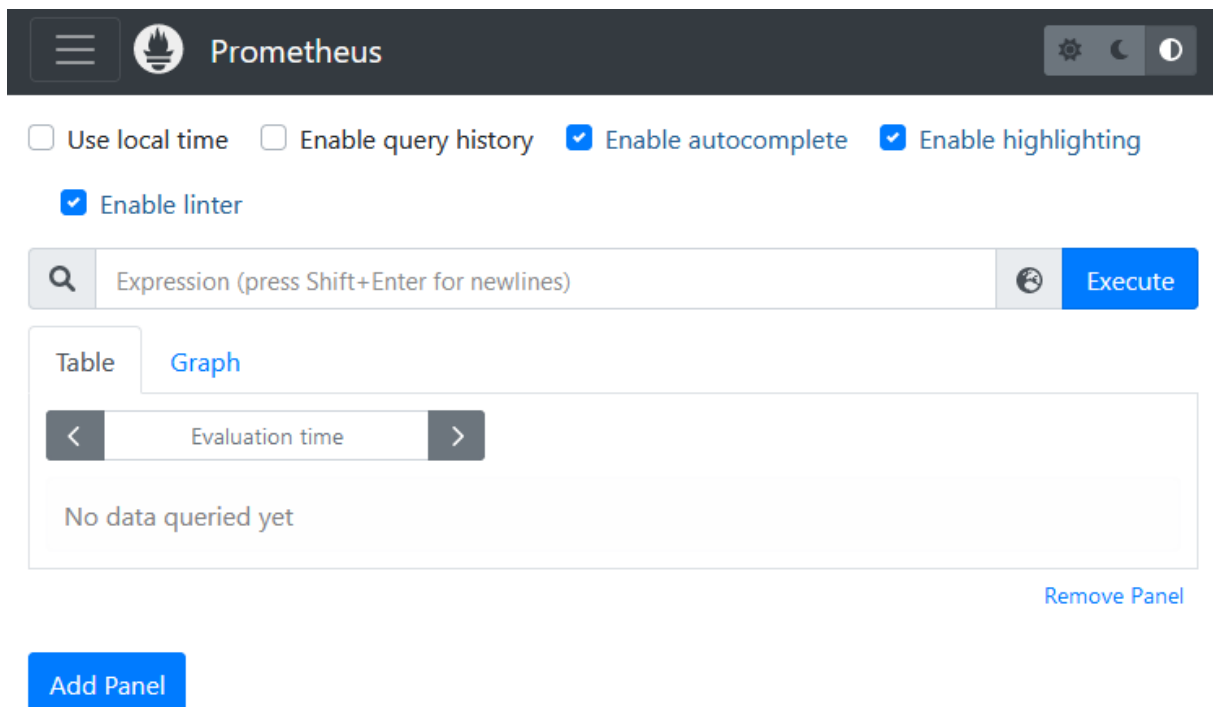
```
systemctl status prometheus
```

```
labadmin@itia-ansible-dev-01:~$ systemctl status prometheus
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-09-30 15:34:02 CEST; 2min 53s ago
     Docs: https://prometheus.io/docs/introduction/overview/
   Main PID: 50412 (prometheus)
    Tasks: 6 (limit: 2337)
   Memory: 36.7M
      CPU: 168ms
   CGroup: /system.slice/prometheus.service
           └─50412 /usr/local/bin/prometheus --config.file=/etc/prometheus/prometheus.yml --
lines 1-10/10 (END)
```

Die Konfiguration des Prometheus Services ist damit abgeschlossen und das Webinterface von Prometheus sollte unter <http://ansible-xxx.itia.ls.eee.intern:9090> verfügbar sein.



Beachten Sie, dass Port 9090/tcp in der Firewall erlaubt werden muss, falls eine Firewall aktiv ist. In unserem Setup ist dies jedoch nicht nötig, da keine Firewall aktiv ist.



3.3 Konfiguration der Prometheus Exporter

Nachdem wir nun Prometheus (Server) installiert haben, wollen wir in einem nächsten Schritt zwei Prometheus Exporter installieren. Prometheus Exporter sammeln Metriken und stellen diese über einen HTTP Endpunkt zur Verfügung.

3.3.1 Konfiguration des Windows Exporter

In diesem Kapitel installieren wir den Windows-spezifischen Prometheus Exporter «windows_exporter». Der Windows Exporter wird von der Community und nicht offiziell von Prometheus zur Verfügung gestellt.

Verbinden Sie sich für diesen Schritt per RDP mit Ihrem **Windows Server windows-xxx.itia.ls.eee.intern** (ersetzen Sie dabei xxx mit Ihrer Gruppennummer).

Unter folgendem GitHub Link finden Sie das Repository des Windows Exporters: https://github.com/prometheus-community/windows_exporter. Unter Releases finden Sie verschiedene

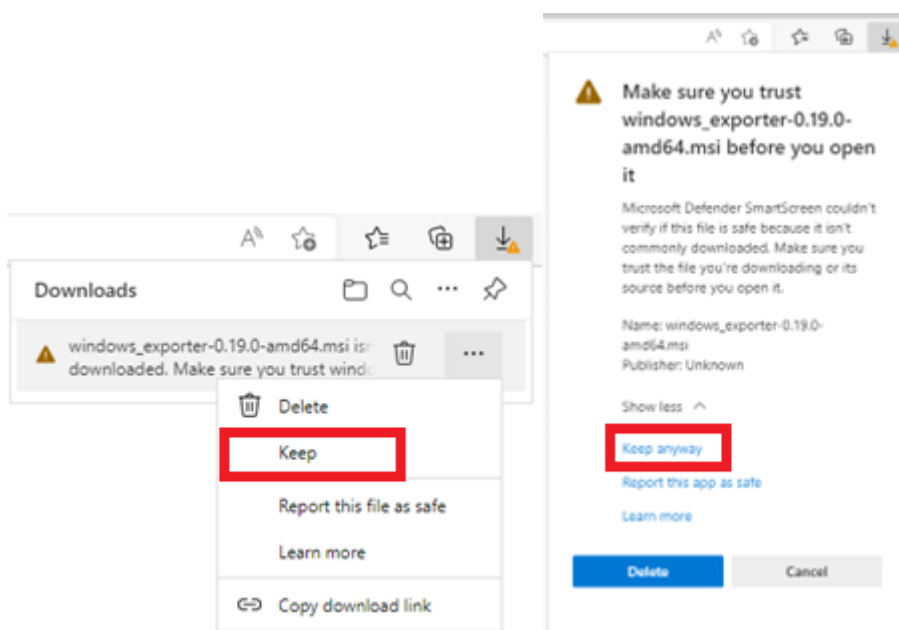
Downloads für den Windows Exporter. Nebst dem Quellcode können so auch .exe und .msi Dateien heruntergeladen werden.

The screenshot shows the GitHub repository page for `prometheus-community/windows_exporter`. The repository is public and has 518 forks and 1.9k stars. The 'About' section describes it as a 'Prometheus exporter for Windows machines'. The 'Releases' section is highlighted with a red box, showing the latest release, `v0.19.0`, dated July 23rd. The 'Code' button is also visible.

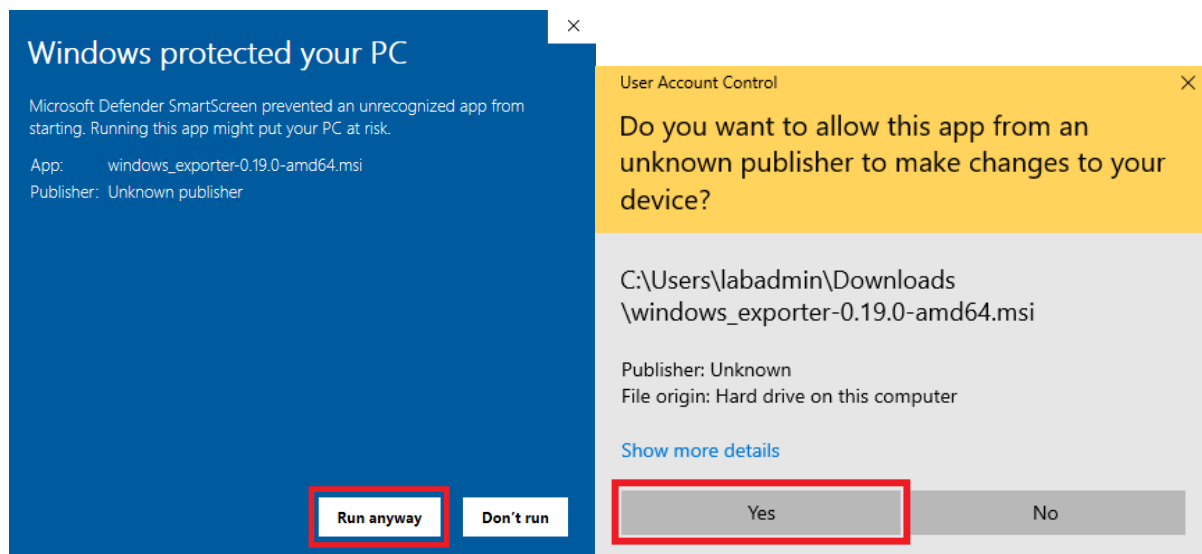
File	Description	Time
.github	Don't spellcheck Go dependency files	last month
collector	Added WebService uptime metric in iis collector	17 days ago
config	Update gopkg.in/yaml_v2 to yaml_v3	4 months ago
contrib/console_templa...	Update console_template metrics	2 years ago
docs	Update collector.mssql.md	9 days ago
headers	golanci-lint: Acknowledge all remaining checks a...	9 months ago
initiate	Additional Comments	last month
installer	Port should default to 9182 if not defined	9 months ago
kubernetes	Update kubernetes.md	3 months ago
log	revive: make type exportable and remove unneces...	10 months ago
tools	feat: adapting to support CimSession and alternat...	3 months ago

Laden Sie nun auf Ihrem Windows Server die neuste Version des Windows Exporters als **.msi Datei** herunter. Achten Sie dabei darauf, dass Sie die **amd64 Version** wählen.

Möglicherweise müssen Sie den Download im Edge Browser zusätzlich bestätigen. Wählen Sie hierzu die Option «Keep» wie in untenstehender Abbildung dargestellt und anschliessend «Show more» und «Keep anyway».

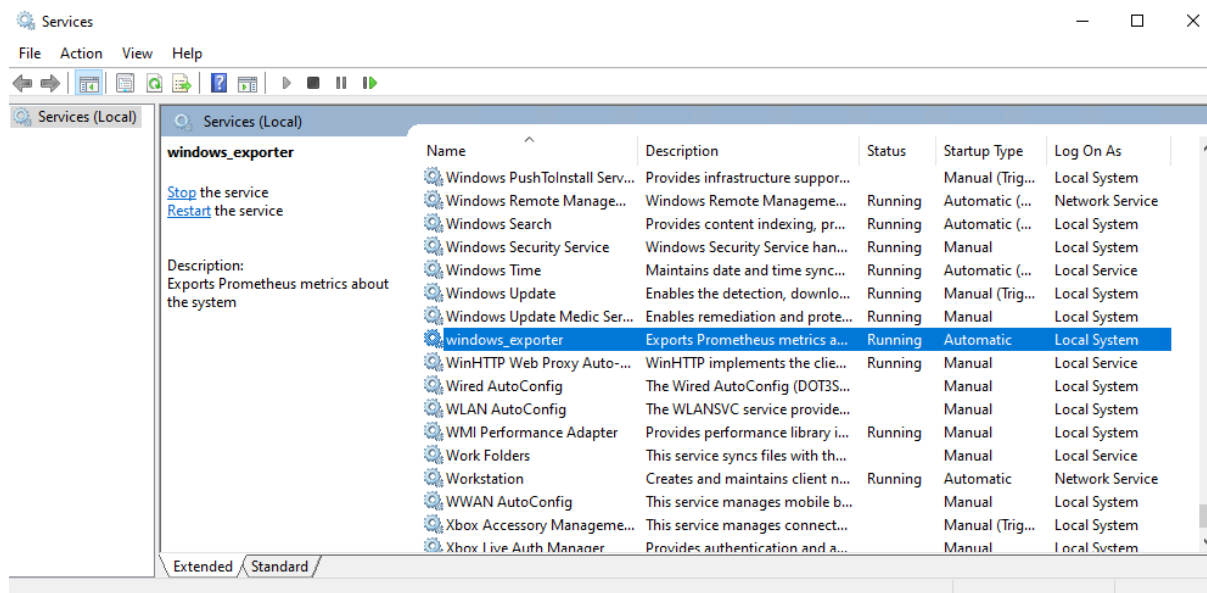


Öffnen Sie anschliessend den Downloads Ordner, in welchem sich die soeben heruntergeladene Datei befindet. Starten Sie die .msi Datei, um den Windows Exporter zu installieren. Auch hier müssen Sie die Warnung von Microsoft Defender mit «Run anyway» bestätigen.




Nach einer letzten Windows Hürde in Form der UAC haben Sie erfolgreich den Windows Exporter installiert. Gratulation!

Um zu überprüfen, ob der Service erfolgreich installiert und gestartet wurde, öffnen Sie «Services» und suchen Sie nach dem Dienst namens «windows_exporter». Wenn alles funktioniert hat, sollte der Service sichtbar und der Status sollte auf «Running» sein.



Um nun Metriken einsehen zu können, navigieren sie nach <http://windows-xxx.itia.lseee.intern:9182/metrics>.

 Was hat es mit dieser /metrics Seite auf sich? Wie funktionieren Prometheus Metriken? Recherchieren Sie.

.....
.....
.....

Wir können nun theoretisch schon Metriken von unserem Windows Server abfragen. Im nächsten Kapitel installieren wir einen Prometheus Exporter auf dem Linux Client, um auch von dort Metriken erhalten zu können.

Die Konfiguration des Windows Server ist damit beendet. Sie können die Remote Desktop Verbindung also wieder schliessen.


3.3.2 Konfiguration des Node Exporter

In diesem Kapitel werden Sie den Prometheus Node Exporter installieren, welcher unter Linux dieselbe Funktion übernimmt wie der im vorherigen Kapitel installierte Windows Exporter.

Die Installation des Node Exporters unter Linux läuft ähnlich ab wie die Installation von Prometheus:

1. Erstellen einer Gruppe und eines Users für Prometheus
2. Download der .tar.gz Datei welche Binaries und Konfigurationsdateien enthält
3. Entpacken der .tar.gz Datei
4. Verschieben der einzelnen Dateien in die korrekten Ordner
5. Anpassen der Berechtigungen
6. Erstellen eines Service Files
7. Starten des Services

Da wir diese Schritte für Prometheus bereits einmal durchgeführt haben und verstehen, installieren wir den Node Exporter der Einfachheit halber mittels apt.

 *Beachten Sie, dass die Version des Prometheus Node Exporters in den apt Quellen von Debian einiges älter ist als die neuste von Prometheus zur Verfügung gestellte Version. Die Kompatibilität zwischen verschiedenen Versionen ist jedoch gegeben.*

Verbinden Sie sich nun mittels SSH mit ihrem **Linux Client** **debian-xxx.itia.ls.eee.intern** und installieren Sie das Paket «prometheus-node-exporter» mittels apt.

```
sudo apt update && sudo apt install -y prometheus-node-exporter
```

Um zu überprüfen, ob wir Daten des Linux Clients abfragen können, navigieren Sie nach <http://debian-xxx.itia.ls.eee.intern:9100/metrics>.

Wenn alles funktioniert hat, sollten Sie hier Metriken des Linux Client sehen, ähnlich wie beim Windows Server.

Die Konfiguration des Linux Clients ist damit abgeschlossen. Sie können die SSH Verbindung schliessen.

3.4 Konfiguration von Prometheus

Da wir nun auf dem Windows Server und Linux Client je einen Prometheus Exporter installiert haben und überprüfen konnten, dass Metriken verfügbar sind, konfigurieren wir in diesem Kapitel Prometheus so, dass die Metriken der beiden Exporter regelmässig abgefragt werden.

Verbinden Sie sich für die nächsten Schritte wieder mittels SSH mit der VM **ansible-xxx.itia.ls.eee.intern**.

Sehen Sie sich als erstes die Konfigurationsdatei `/etc/prometheus/prometheus.yml` an und versuchen Sie die einzelnen Abschnitte zu verstehen.

```
cat /etc/prometheus/prometheus.yml
```



Die offizielle Dokumentation von Prometheus kann ihnen beim Verstehen der einzelnen Parameter behilflich sein:

<https://prometheus.io/docs/prometheus/latest/configuration/configuration/>.



Was bewirkt der folgende Ausschnitt der Konfigurationsdatei?

Gehen Sie dabei speziell auf «- targets: ["localhost:9090"]» ein. Warum ist das so?

```
scrape_configs:
  # The job name is added as a label 'job=<job_name>' to any timeseries scraped from this config.
  - job_name: "prometheus"

  # metrics_path defaults to '/metrics'
  # scheme defaults to 'http'.

static_configs:
  - targets: ["localhost:9090"]
```

.....

.....

.....

.....

Öffnen Sie die Datei `/etc/prometheus/prometheus.yml` mit dem Editor Ihrer Wahl und passen Sie die Sektion «`scrape_configs`» so an, dass die Metriken der beiden zuvor installierten Exporter abgefragt werden.



Achten Sie darauf, dass Sie den Editor mit root Rechten starten müssen, da ihr Benutzer «labadmin» keine Schreibberechtigung auf der Datei `prometheus.yml` hat.



Da die Konfigurationsdatei `prometheus.yml` das YAML Format verwendet, ist sie anfällig auf Einrückungsfehler. Achten Sie daher genau darauf, dass die einzelnen Konfigurationen korrekt eingerückt sind.

Erstellen Sie für jeden der beiden Exporter einen neuen Job:

- «win» für den Windows Server
- «deb» für den Debian Client
- Als Target können Sie entweder die IP oder den FQDN wählen
- Achten Sie darauf, dass Sie den jeweils korrekten Port verwenden (9182 für den Windows Exporter, 9100 für den Node (Linux) Exporter)

Um die Änderungen in der Konfigurationsdatei zu übernehmen, starten Sie den Prometheus Service neu.

```
sudo systemctl restart prometheus.service
```

Folgender Befehl kann Ihnen beim Finden von allfälligen Fehlern in der Konfigurationsdatei helfen.

```
sudo journalctl -eu prometheus.service
```

Nachdem Sie die Konfigurationsdatei korrekt angepasst haben und Prometheus startet, können Sie die beiden neu hinzugefügten targets im Webinterface einsehen.

Navigieren Sie hierzu nach <http://ansible-xxx.itia.ls.eee.intern:9090/targets>. Die Ansicht sollte wie auf nachfolgender Abbildung aussehen.

Targets

All Unhealthy Collapse All

deb11 (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://itia-debian11-dev-01.el.eee.intern:9100/metrics	UP	instance="itia-debian11-dev-01.el.eee.intern:9100" job="deb11"	14.150s ago	65.464ms	

prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	3.524s ago	4.223ms	

win10 (1/1 up) show less

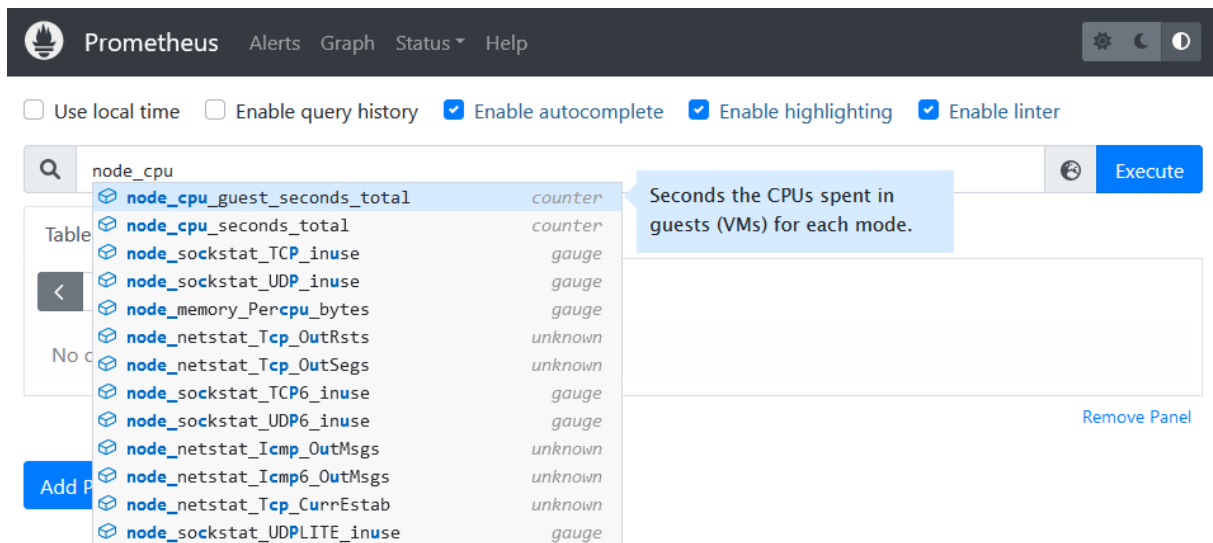
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://itia-w10-dev-01.el.eee.intern:9182/metrics	UP	instance="itia-w10-dev-01.el.eee.intern:9182" job="win10"	6.376s ago	861.446ms	

Im Webinterface können Sie nun unter dem Menüpunkt «Graph» einzelne Metriken der beiden Exporter einsehen:


- Metriken des Windows Exporters haben den Prefix «windows_»
- Metriken des Node Exporters haben den Prefix «node_»

Nachfolgende Abbildung zeigt den Menüpunkt «Graph», über welchen einzelne Metriken angezeigt werden können. Metriken können als aktuelle Werte oder in Form eines Graphen angezeigt werden.

Über das Suchfeld können Sie Metriken suchen. Die Autovervollständigung und der Metrics Explorer (Weltkugel links neben Execute) hilft dabei zu entdecken, welche Metriken es gibt. Mit Execute kann ein Query abgesetzt werden.



Sehen Sie sich einige Metriken der beiden Exporter an und beantworten Sie folgende Fragen:

 Wie viele logische Prozessoren hat Ihr Windows Server?


.....

.....

 Welche Informationen erhalten Sie aus der Metrik «windows_os_info»?


.....

.....

 Versuchen Sie herauszufinden, zu welchem Zeitpunkt ihr Linux Client zum letzten Mal neu gestartet wurde. In welchem Format wird diese Information zur Verfügung gestellt?

.....

.....


 Welche Informationen erhalten Sie aus der Metrik «node_network_up»? Erklären Sie.

Die Query Language PromQL bietet verschiedene Operatoren und Funktionen an, mit welchen Abfragen kombiniert oder weiterverwertet werden können. Die komplette Liste der Operatoren und Funktionen kann in der Dokumentation von Prometheus eingesehen werden:

<https://prometheus.io/docs/prometheus/latest/querying/basics/> (siehe auch Unterpunkte «Operators» und «Functions»).

Wir wollen uns nun die Netzerklastung des Windows Servers ansehen. Hierfür verwenden wir folgenden Query:

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

 Erläutern Sie mit Hilfe der Dokumentation, was die Funktion «rate» macht und wofür das «[1m]» am Ende des Queries steht.



Wir haben nun gesehen, dass wir einzelne Metriken auslesen, diese als Graph darstellen und sogar mittels Operatoren und Funktionen verändern können.

Während das Prometheus Webinterface zwar zu Testzwecken gut verwendet werden kann, scheint dies keine optimale Lösung zum Betrachten von Metriken in einem produktiven Umfeld. Wir werden uns daher im nächsten Kapitel die Lösung Grafana näher ansehen, mit welcher die von Prometheus gesammelten Daten etwas komfortabler dargestellt werden können.

3.5 Installation von Grafana

In diesem Kapitel werden Sie Grafana installieren.

Grafana Labs beschreiben auf ihrer Website verschiedene Installationsmöglichkeiten (.deb Paket, .tar.gz Datei, Kubernetes und weitere) für Grafana. Wir werden uns in diesem Versuch dafür entscheiden, Grafana von der von Grafana Labs zur Verfügung gestellten apt Quelle zu installieren. Weitere Informationen zu den verschiedenen Installationsvarianten und eine ausführliche Installationsanleitung ist auf der Website von Grafana Labs zu finden (<https://grafana.com/docs/grafana/latest/setup-grafana/installation/>).

Grafana Labs bieten Grafana in zwei verschiedenen Versionen an: Grafana Enterprise und Grafana Open Source (OSS). Wir werden mit den folgenden Befehlen Grafana OSS installieren. Weitere Informationen zu den Unterschieden der beiden Versionen finden Sie in einem Blogpost: <https://grafana.com/blog/2019/09/04/how-we-differentiate-grafana-enterprise-from-open-source-grafana/>.

In einem ersten Schritt wird sichergestellt, dass notwendige Abhängigkeiten installiert sind.

```
sudo apt update
sudo apt install -y apt-transport-https
sudo apt install -y software-properties-common wget
```

Als nächstes fügen wir den GPG Key von Grafana Labs hinzu, um die Validität des Grafana Pakets überprüfen zu können.

```
sudo wget -q -O /usr/share/keyrings/grafana.key https://packages.grafana.com/gpg.key
```

Um Grafana von den apt Quellen installieren zu können, müssen wir das Repository von Grafana Labs hinzufügen.

```
echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://packages.grafana.com/oss/deb
stable main" | sudo tee -a /etc/apt/sources.list.d/grafana.list
```


Als letztes wird Grafana installiert. Da wir soeben ein neues Repository zu unseren Quellen hinzugefügt haben, ist ein update der apt Quellen nötig.

```
sudo apt update
sudo apt install -y grafana
```

Folgende Meldung weist darauf hin, dass ein Service für Grafana installiert wurde, dieser jedoch nicht aktiviert ist. Mit den beschriebenen Befehlen ist es möglich den Service zu aktivieren, sodass Grafana automatisch beim Systemstart startet.

```
### NOT starting on installation, please execute the following statements to configure grafana
to start automatically using systemd
sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable grafana-server
### You can start grafana-server by executing
sudo /bin/systemctl start grafana-server
```

Aktivieren und starten Sie den Grafana Service. Sie können den Status von Grafana wiederum mit dem Tool `systemctl` überprüfen.

 Wie lautet der Befehl um den Status von Grafana zu überprüfen?

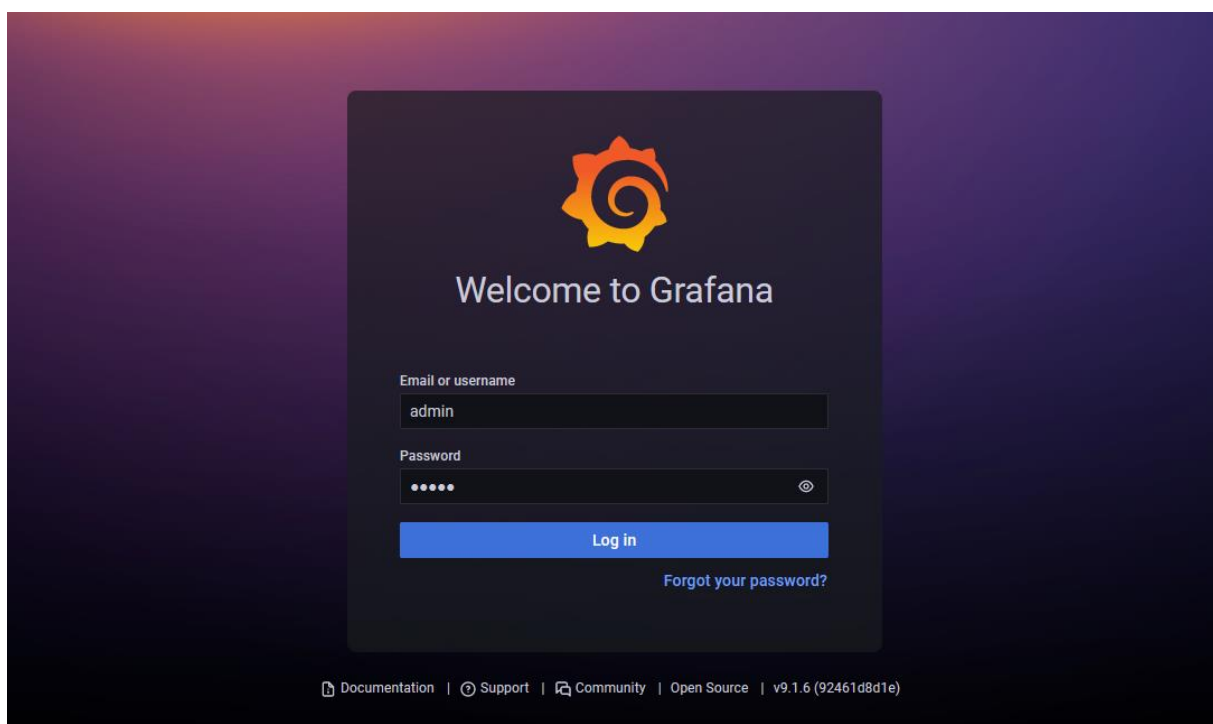
.....

.....

Gratulation! Sie haben Grafana erfolgreich installiert und können nun mit der Konfiguration von Grafana fortfahren.

3.6 Konfiguration von Grafana

Nach der Installation von Grafana sollte das Webinterface nun auf Port 3000 verfügbar sein. Navigieren Sie also nach <http://ansible-xxx.itia.ls.eee.intern:3000> und melden Sie sich an. Die Anmeldedaten direkt nach der Installation lauten: **admin:admin**.

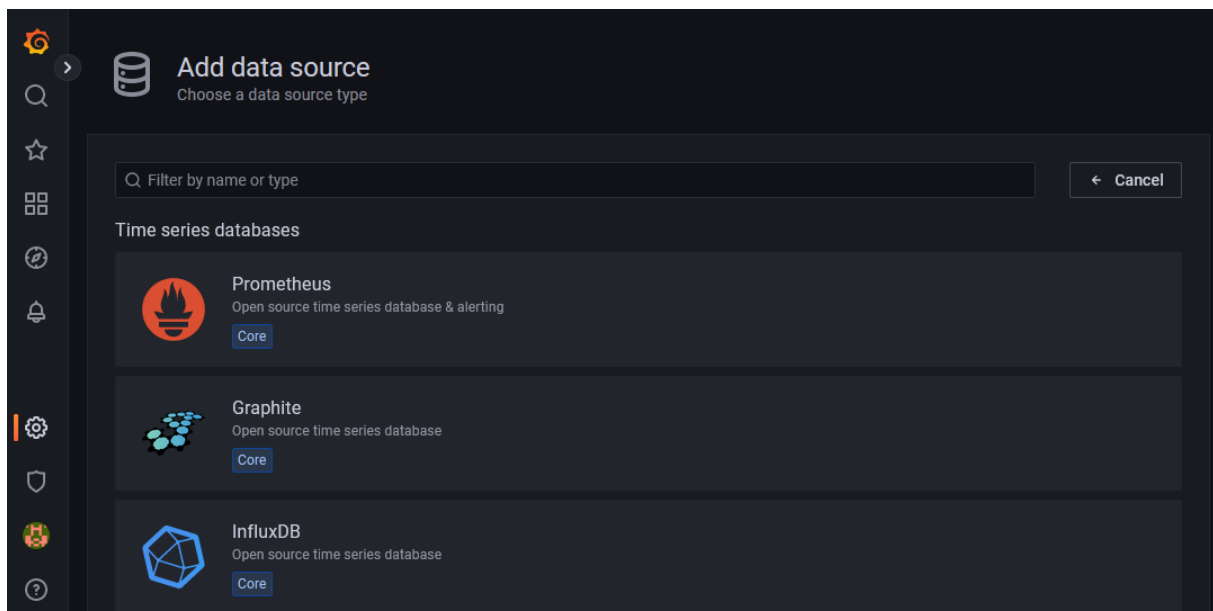


Nach dem Anmelden werden Sie dazu aufgefordert das Passwort zu ändern. Wählen Sie **Hslu123** als neues Passwort.

Sie befinden sich nun auf dem Home Screen von Grafana, welcher einige hilfreiche Informationen und Tutorials zu den ersten Schritten mit Grafana darstellt. Wir sind im Rahmen dieses Laborversuchs nur daran interessiert unsere Prometheus Daten mittels Grafana darzustellen. Mit den folgenden Schritten werden Sie Prometheus als Datenquelle hinzufügen und ein vorgefertigtes Dashboard importieren, welches Daten des Node Exporters anzeigen kann.

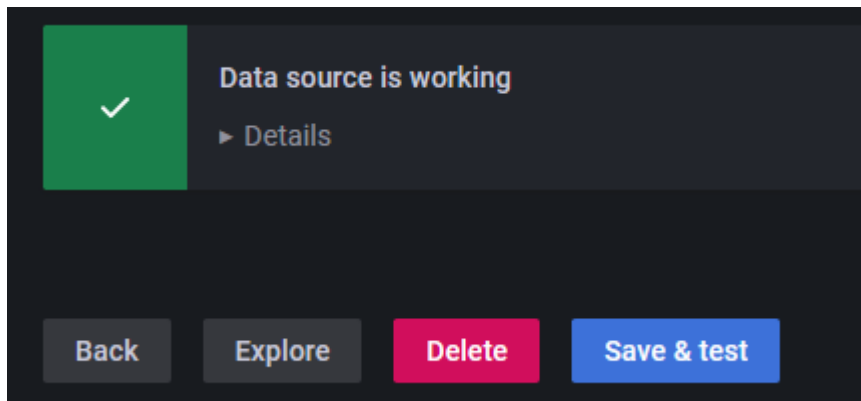
👉 Ähnlich wie bei Prometheus steht auch für Grafana eine Konfigurationsdatei unter `/etc/grafana/grafana.ini` zur Verfügung. Über diese Konfigurationsdatei werden globale Grafana Einstellungen verwaltet wie z.B. Single-Sign-On Optionen oder auf welchem Port Grafana erreichbar sein soll. Da Grafana eine multiuser Applikation ist, werden alle Einstellungen, die per Webinterface getätigt werden in einer Datenbank gespeichert.

Um Prometheus als Datenquelle hinzuzufügen, klicken Sie links auf das Zahnrad und anschliessend auf «Data Sources». Über den Button «Add data source» kann eine neue Datenquelle hinzugefügt werden. Wählen Sie «Prometheus».



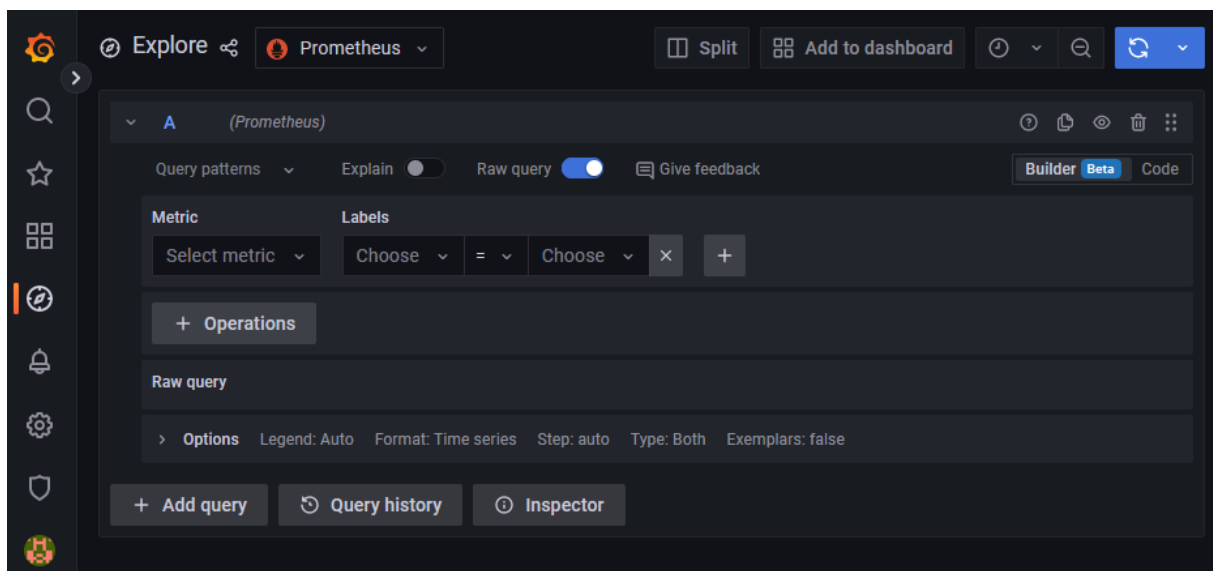
Im nachfolgenden Dialog können Sie die Prometheus Datenquelle konfigurieren. Das einzige für uns relevante Feld ist «URL». Tragen Sie hier <http://localhost:9090> ein. Alle anderen Einstellungen können auf den Standardwerten belassen werden.

Klicken Sie nun auf «Save & test». Wurde die Datenquelle korrekt konfiguriert erhalten Sie die Meldung «Data source is working».



3.7 Erste Prometheus Queries mit Grafana

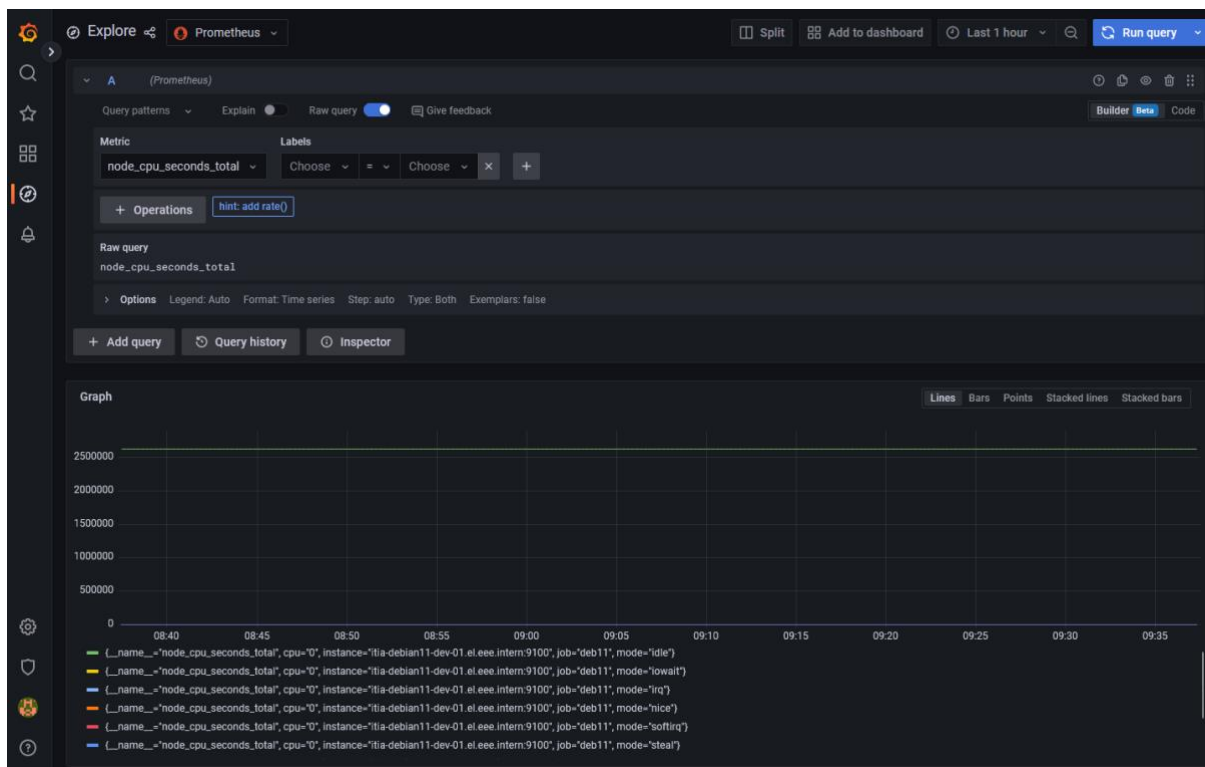
Grafana kann nun Daten von Prometheus abfragen. Um dies kurz zu testen, wählen links das Kompass Icon (Explore) aus.



Mithilfe der «Explore» Funktion von Grafana ist es möglich, Ad-hoc Daten abzufragen, ohne dass ein Dashboard erstellt werden muss. Im Falle von Prometheus ist dieses Interface sehr ähnlich, wie das von Prometheus zur Verfügung gestellte.

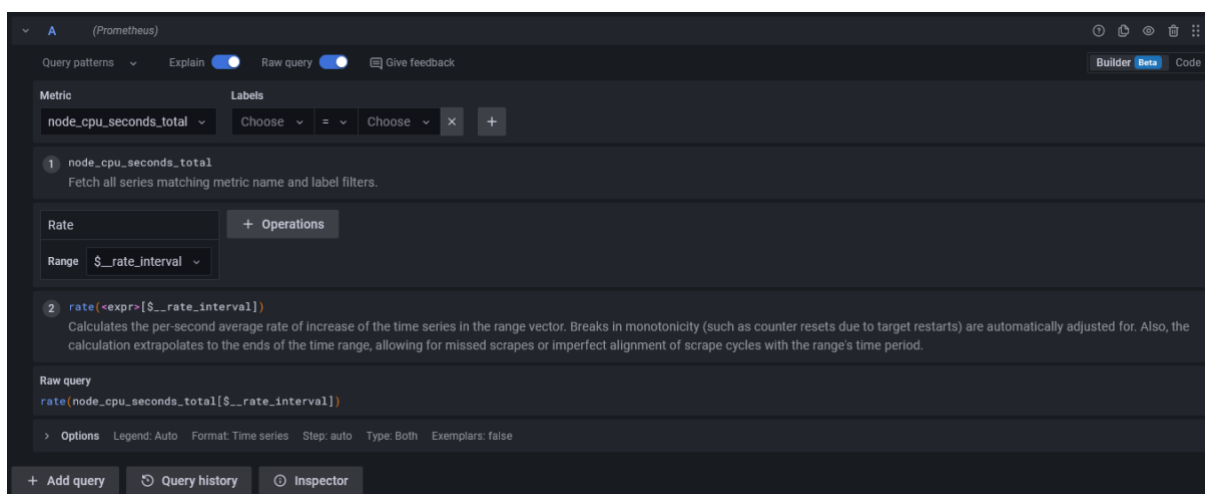
Wählen Sie eine beliebige Metrik aus und klicken Sie auf «Run query» um einen Graphen der Metrik anzeigen zu lassen. Die dargestellten Datenpunkte sind weiter unten zusätzlich in tabellarischer Form ersichtlich.

Über die Schaltfläche «Operations» können Sie die Daten mithilfe von Aggregation, Funktionen und anderen Operationen transformieren. Im untenstehenden Beispiel wurde die Metrik «node_cpu_seconds_total» gewählt. Grafana schlägt automatisch vor, diese Metrik mithilfe der bereits bekannten Funktion rate() etwas sinnvoller darzustellen.



Fügen Sie die Funktion `rate()` wie vorgeschlagen hinzu und aktualisieren Sie die Abfrage über «Run query» rechts oben.

Wenn Sie den Schalter «Explain» aktivieren, werden Erklärungen zu den verwendeten Metriken und Operations hinzugefügt.

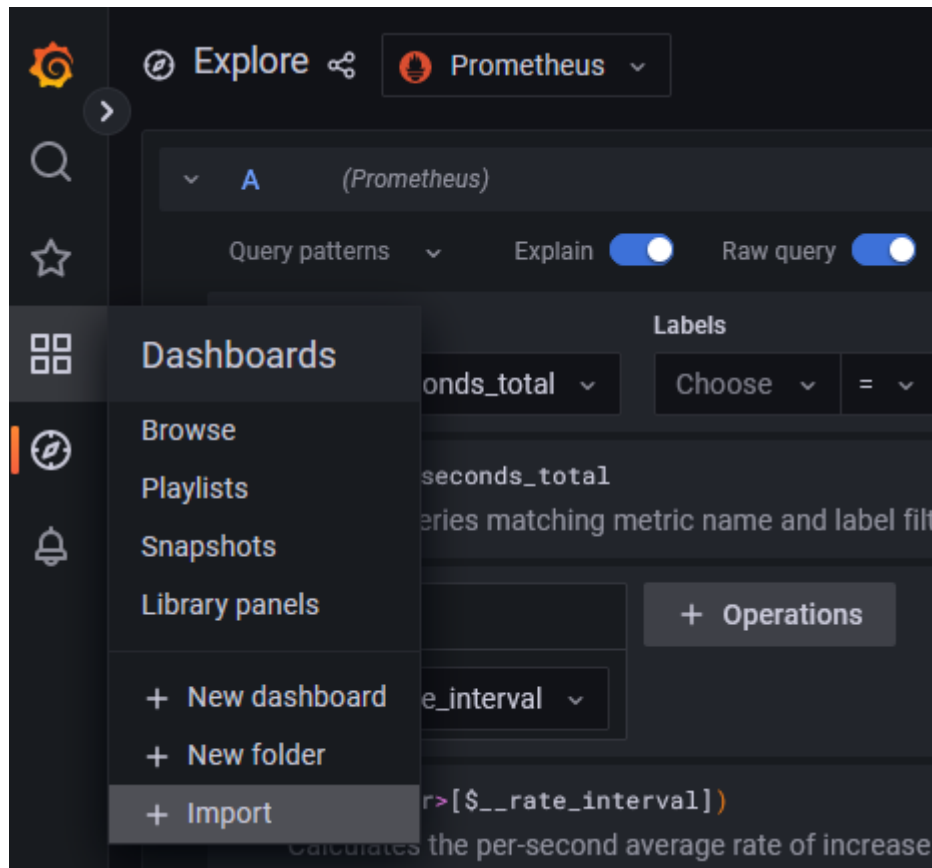


Experimentieren Sie mit den verschiedenen Optionen und Darstellungsarten auf der «Explore» Seite und wechseln Sie anschliessend ins nächste Kapitel um ein vorgefertigtes Dashboard zu importieren.

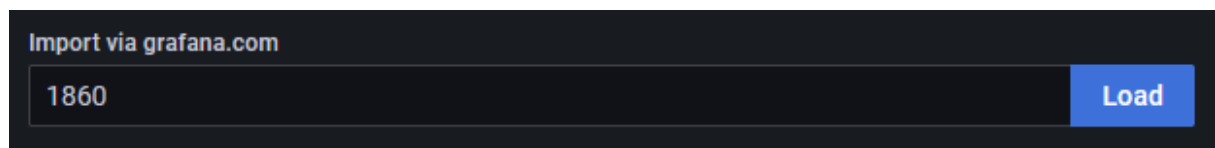
3.8 Grafana Dashboard importieren

Auf der Website von Grafana steht eine von der Community gepflegte Sammlung von vorgefertigten Grafana Dashboards für verschiedenste Produkte/Exporter zur Verfügung. Suchen Sie nach einem Dashboard um Node Exporter Daten darzustellen.

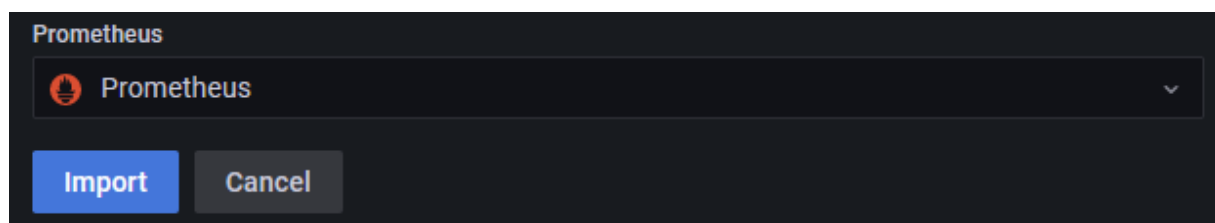
Sie sollten das Dashboard «Node Exporter Full» finden, welches die ID 1860 hat. Wechseln Sie zurück ins Grafana Webinterface und wählen Sie links «Dashboards > Import».



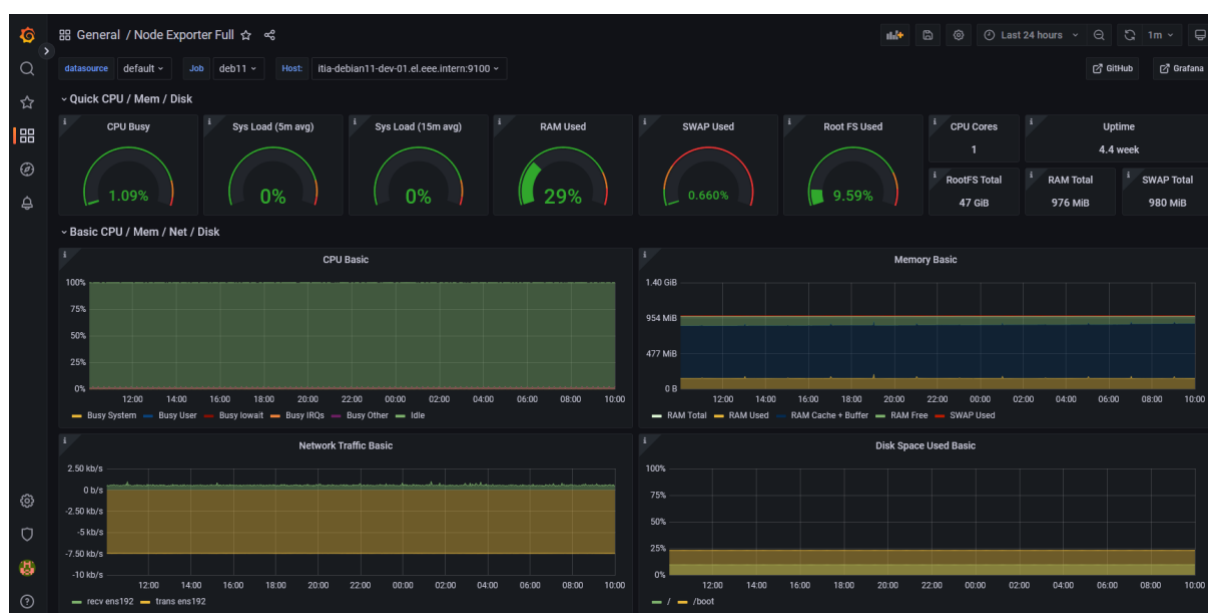
Im nächsten Dialog können Sie ein Dashboard über die Eingabe der zuvor gefundenen ID oder als JSON importieren. Geben Sie im Feld «Import via grafana.com» die ID 1860 ein und klicken Sie auf «load».




Wählen Sie im nächsten Fenster die korrekte Datenquelle «Prometheus» aus und klicken Sie auf «Import». Die restlichen Parameter können auf den Standardwerten belassen werden.



Nach dem Import des Dashboards werden Sie automatisch weitergeleitet. Das Dashboard sollte ähnlich wie im folgenden Screenshot aussehen.

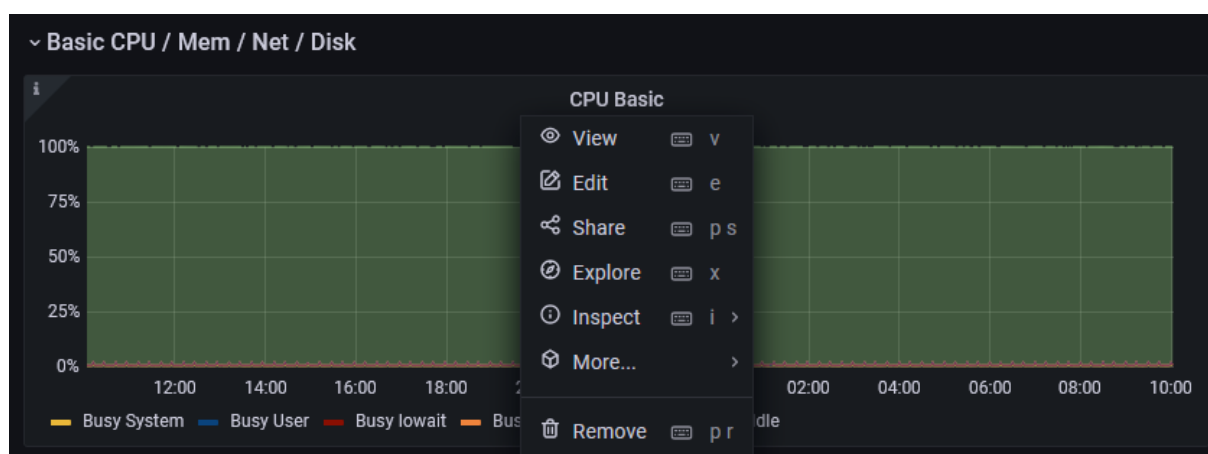


Nehmen Sie sich etwas Zeit, um die dargestellten Daten zu betrachten.

 Unter der Reihe «Basic CPU / Mem / Net / Disk» gibt es viele weitere Reihen, welche eingeklappt sind. Betrachten Sie auch einige dieser Anzeigen.

Das importierte Dashboard kann nach belieben bearbeitet werden. Fahren Sie hierzu mit der Maus über das gewünschte Panel und klicken Sie auf den kleinen Pfeil rechts neben dem Titel des Panels.

Im nun erschienenen Kontextmenu kann das Panel unter anderem bearbeitet oder gelöscht werden.



Sehen Sie sich einige Queries im Dashboard an und versuchen Sie deren Funktionsweise zu verstehen.


Im nächsten Kapitel werden Sie ein einfaches eigenes Dashboard erstellen, um Daten des Windows Exporters darzustellen.

3.9 Erstellen eines eigenen Dashboards

Mithilfe des importierten Dashboards «Node Exporter Full» ist es uns möglich eine Vielzahl an Metriken welche vom Node Exporter geliefert werden zu visualisieren. Erstellen Sie zum Abschluss dieser Laborübung ein eigenes Dashboard, welches folgende Informationen darstellen soll:

- CPU Auslastung des Windows Servers

- RAM Auslastung des Windows Servers
- Netzwerkverkehr des Windows Servers
- Disk Verbrauch/Total des Windows Servers

 Dokumentieren Sie die *Queries*, welche Sie zum Abfragen der obenstehenden Daten verwendet haben.

.....

.....

.....


.....

.....

.....

.....

.....

 Es ist anzumerken, dass in dieser Laborübung nur ein kleiner Teil des Funktionsumfangs von Grafana betrachtet wird.

Das Erstellen des eigenen Dashboards markiert das Ende dieses Laborversuchs.

3.10 Grafana Plugins

Der Funktionsumfang von Grafana kann mit Plugins erweitert werden. So ist es beispielsweise möglich, neue Datenquellen, Panels oder Apps (Kombination von Datenquellen und Panels) hinzuzufügen. Weitere Informationen zu Plugins sowie eine Liste der verfügbaren Plugins ist unter <https://grafana.com/grafana/plugins/> zu finden.

Notizen

[illegible]