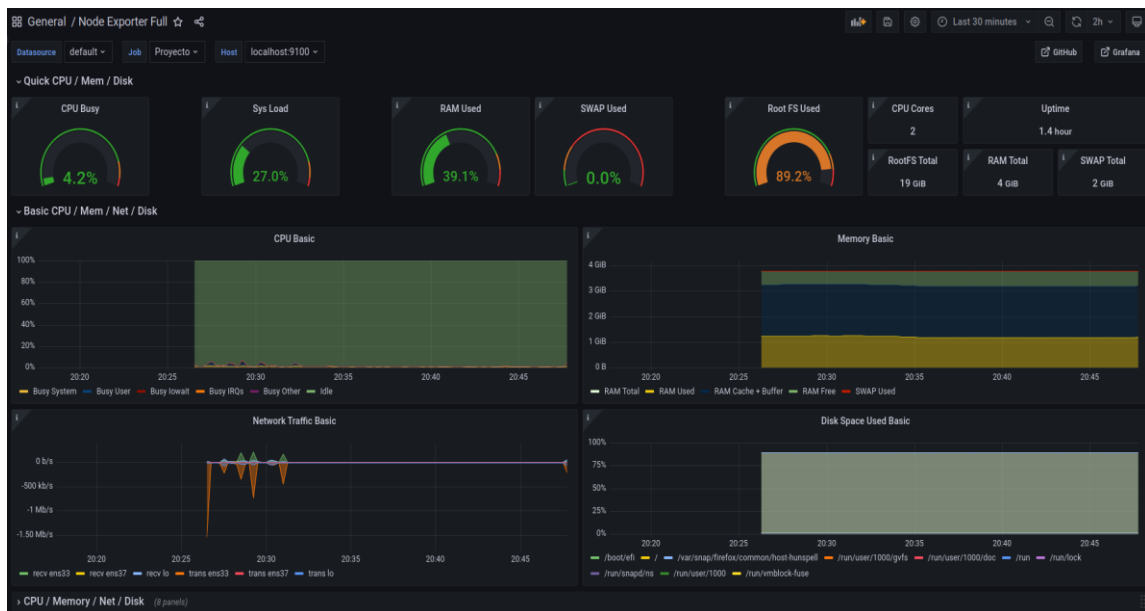


Monitorización con Grafana en Azure y on-premises



Ángel Barahona García

Abril 2025

Índice

Grafana Azure	3
Creación Máquina Virtual.....	3
Habilitar Azure Monitor	5
Creación Azure Managed Grafana	6
Asignación de permisos	7
Creación Dashboard	8
Configuración Alerta	10
Comprobación	12
Grafana on-premises	14
Herramientas necesarias	14
Instalación Herramientas	14
Configuración Prometheus	15
Configuración Grafana	16
Dashboard final	17

Grafana Azure

Creación Máquina Virtual

Creamos el grupo de recursos e indicamos el nombre de la vm.

Home > Compute infrastructure > Virtual machines >

Create a virtual machine

Help me create a low cost VM | Help me create a VM optimized for high availability | Help me choose the right VM size for my workload

Basics | Disks | Networking | Management | Monitoring | Advanced | Tags | Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription: Azure for Students
Resource group: (New) proyecto
[Create new](#)

Instance details

Virtual machine name: vm-proyecto
Region: (US) East US

Seleccionamos la zona, el tipo de máquina y el tamaño.

Zone options

☒ Self-selected zone
Choose up to 3 availability zones, one VM per zone

☐ Azure-selected zone (Preview)
Let Azure assign the best zone for your needs

Using an Azure-selected zone is not supported in region 'East US'.

Availability zone

Zone 3

You can now select multiple zones. Selecting multiple zones will create one VM per zone. [Learn more](#)

Security type

Trusted launch virtual machines

Configure security features

Trusted launch virtual machine is required when using 1P Gallery images.

Image

Ubuntu Server 22.04 LTS - x64 Gen2

[See all images](#) | [Configure VM generation](#)

VM architecture

☐ Arm64
☒ x64

Run with Azure Spot discount

☐

Size

Standard_B1s - 1 vcpu, 1 GiB memory (7.59 US\$/month) (free services eligibl...)

[See all sizes](#)

Generamos las claves SSH para tener más seguridad al acceder.

Escogemos un disco SSD para que vaya con rapidez.

Dejamos por defecto la configuración de networking, que nos crea una red, una subred, una IP pública y un NSG con el puerto 22 abierto.

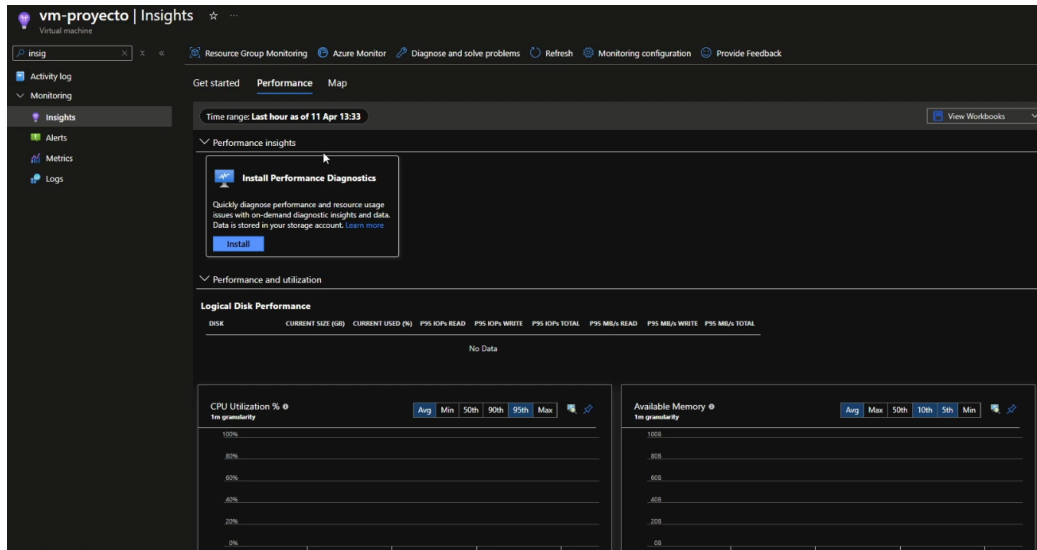
The screenshot shows the 'Networking' tab in the Azure portal for a virtual machine. It includes tabs for Basics, Disks, Networking, Management, Monitoring, Advanced, Tags, and Review + create. The Networking tab is active, showing instructions to define network connectivity by configuring the network interface card (NIC) settings. Below the instructions, there are several configuration options: Virtual network (set to '(new) vm-proyecto-vnet'), Subnet (set to '(new) default (10.0.0.0/24)'), Public IP (set to '(new) vm-proyecto-ip'), NIC network security group (set to 'Basic'), Public inbound ports (set to 'Allow selected ports'), and Select inbound ports (set to 'SSH (22)'). A warning message at the bottom states: 'This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.'

Habilitar Azure Monitor

Para permitir la monitorización detallada del rendimiento y la actividad de la máquina virtual, habilitamos Azure Monitor. Esta funcionalidad proporciona métricas, gráficos y diagnósticos en tiempo real sobre el estado de la VM.

The screenshot shows the 'vm-proyecto | Insights' page in the Azure portal. The left sidebar has a red box around the 'Insights' link. The main content area shows the 'Monitoring configuration' dialog. The dialog has a 'Subscription' dropdown set to 'Azure for Students' and a 'Data collection rule' dropdown set to 'MSVM: DefaultWorkspace-f39159ca-41a3-4dc7-92ef-ced7a8eeae10-EUS'. Below these, there are checkboxes for 'Guest performance' (Enabled), 'Processes and dependencies (Map)' (Disabled), and 'Log Analytics workspace' (DefaultWorkspace-f39159ca-41a3-4dc7-92ef-ced7a8eeae10-EUS). At the bottom, there is a blue bar with a message: 'This will also enable System Assigned Managed Identity, in addition to existing User Assigned identities (if any). Note: Unless specified in the request, the machine will default to using System Assigned Identity. Learn More'. Below this bar are 'Configure' and 'Cancel' buttons.

Comprobamos que se ha activado.



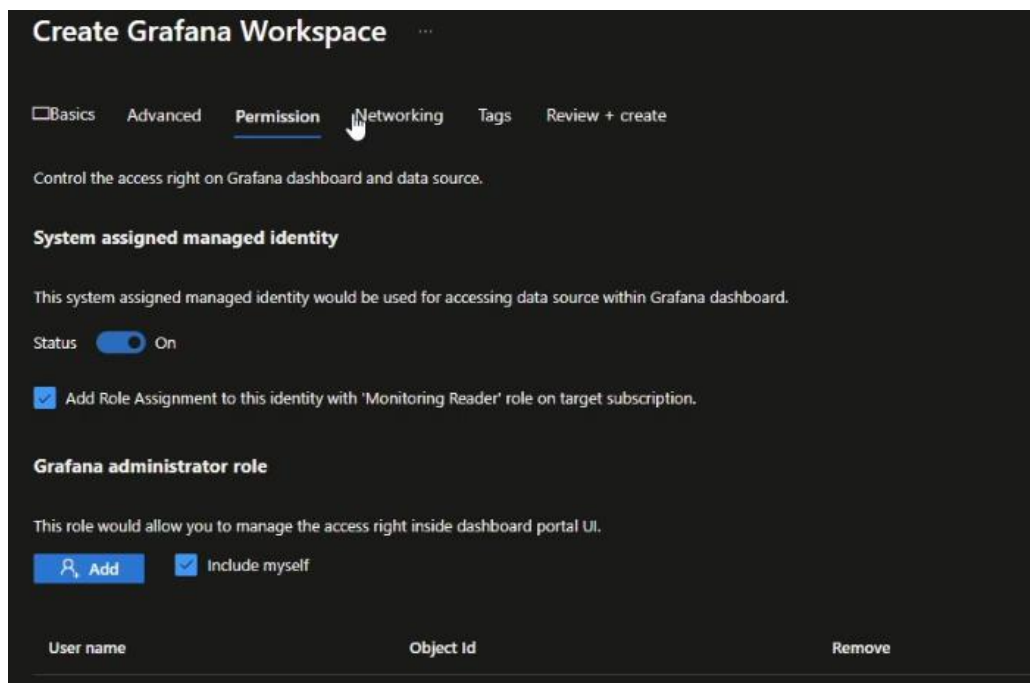
Creación Azure Managed Grafana

Indicamos el grupo de recursos, la misma región que los anteriores recursos creados, un nombre y el plan en standard ya que el gratuito esta bastante limitado.

The screenshot shows the 'Create Grafana Workspace' form in the Azure portal. The form has tabs for Basics, Advanced, Permission, Networking, Tags, and Review + create. The Basics tab is selected. The form includes the following sections and fields:

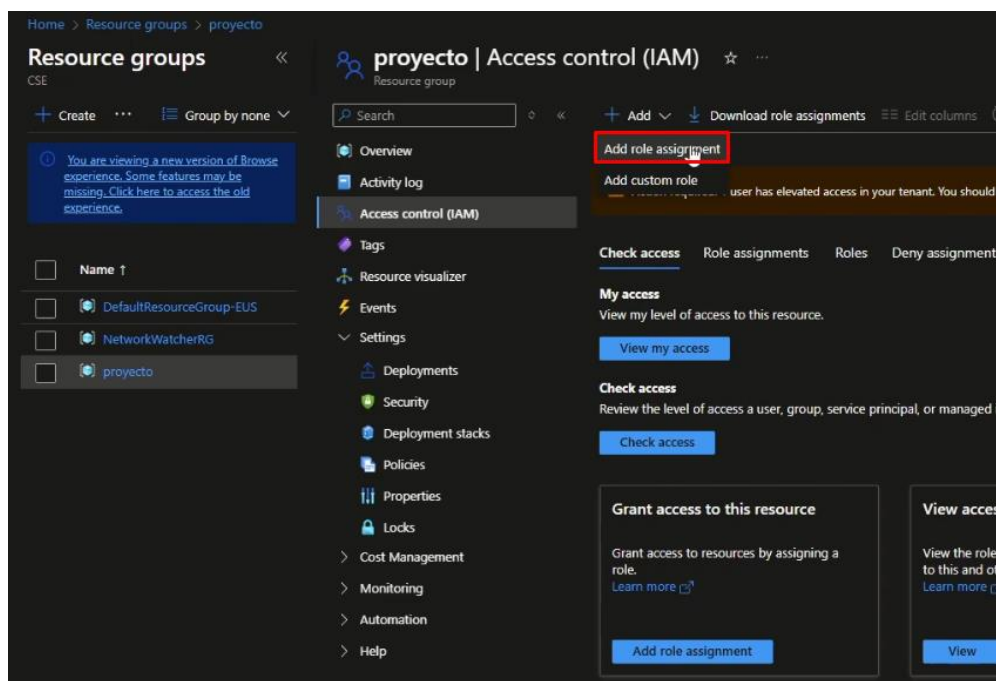
- Project Details**
 - Subscription: Azure for Students
 - Resource group name: proyecto (with a 'Create new' link)
- Instance Details**
 - Location: (US) East US
 - Name: grafana-proyecto
- Pricing Plans**
 - Pricing plan determines features, resources, and charges associated with your Managed Grafana workspace
 - Pricing Plan (View Pricing Details): Standard
 - 0.0684 USD for per hour (Default 2 nodes per instance) + 6 USD for per month per user (Estimated)
- Grafana Version**
 - Grafana Version: 11

Importante dejar marcada esta opción para asignar una managed identity y luego poder utilizarla.

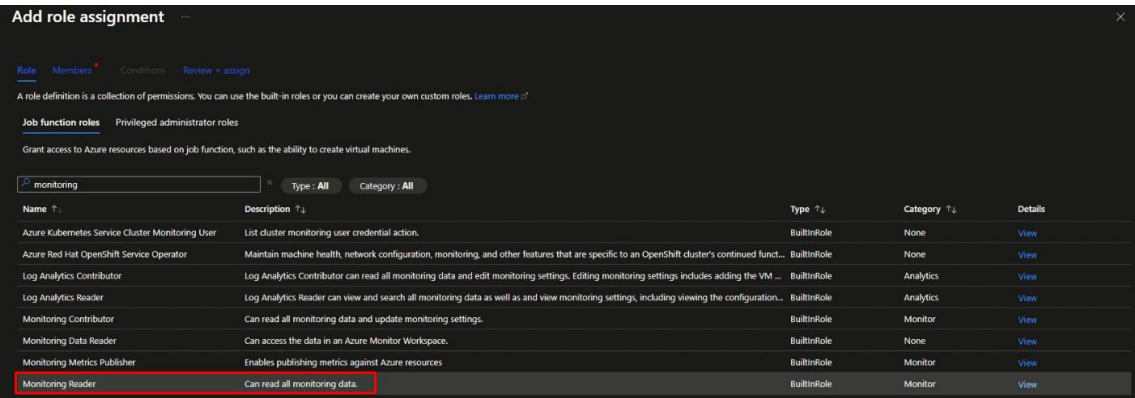


Asignación de permisos

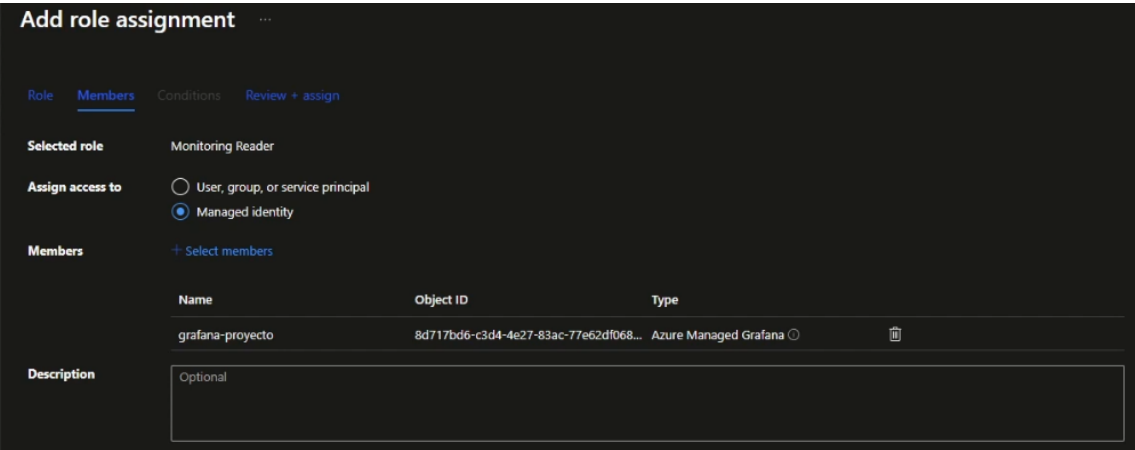
Accedemos a nuestro grupo de recursos y añadimos un rol.



Seleccionamos el rol de Monitoring Reader para que grafana pueda leer las métricas.

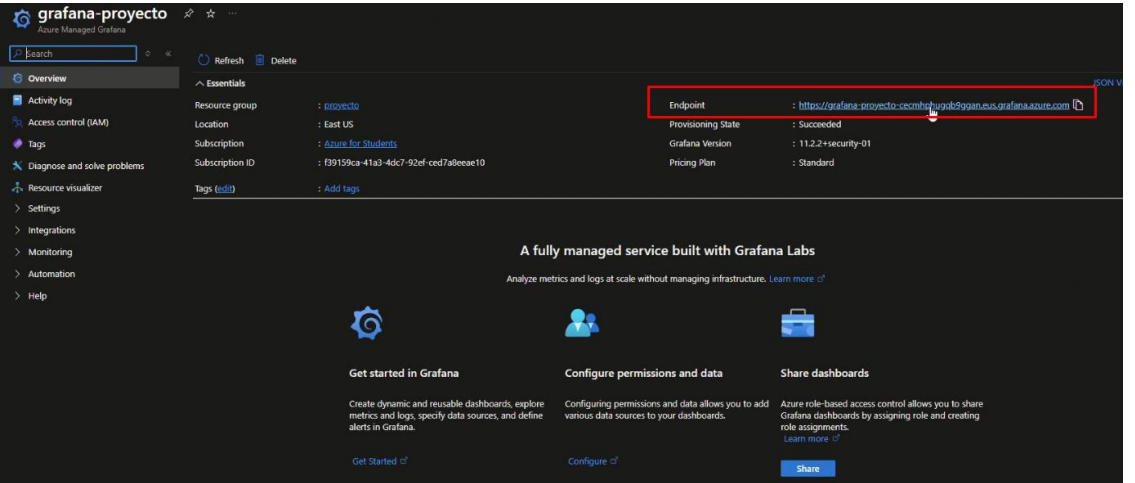


Asignamos el acceso a la Managed Identity de grafana.

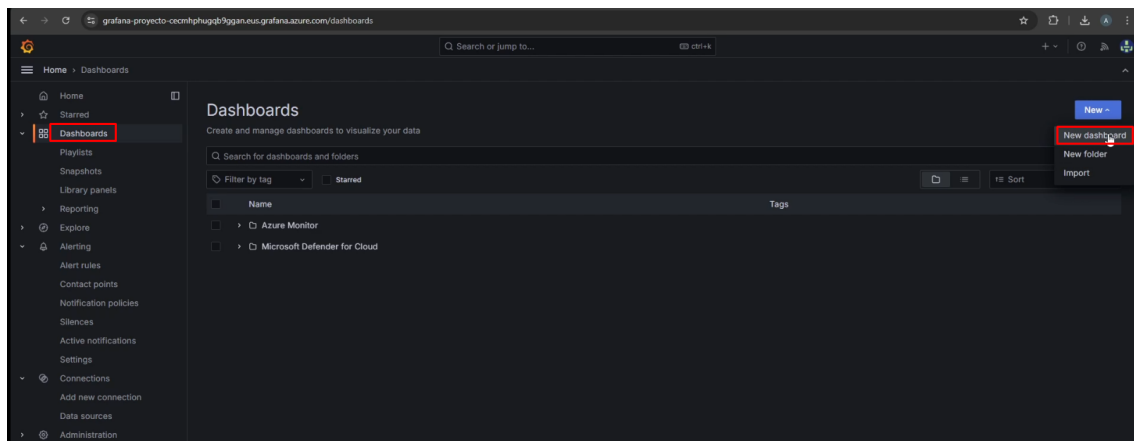


Creación Dashboard

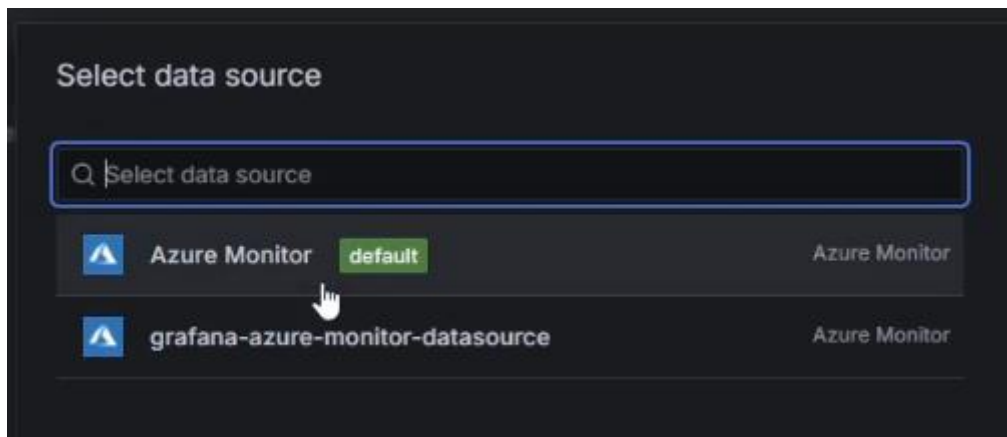
Accedemos a grafana a través del link de Endpoint.



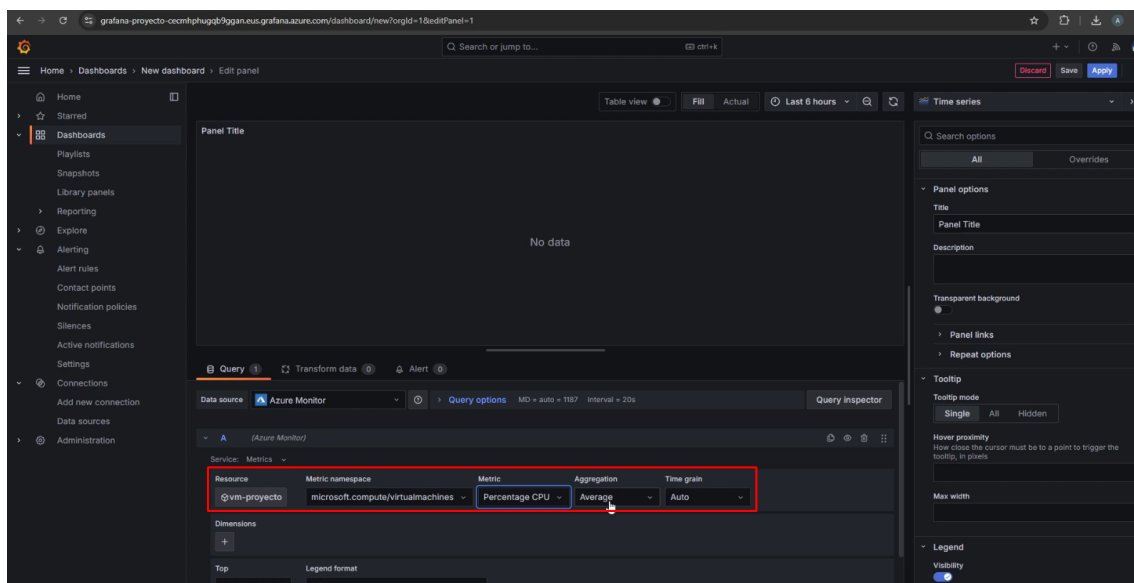
Nos dirigimos a dashboards y a new dashboard.



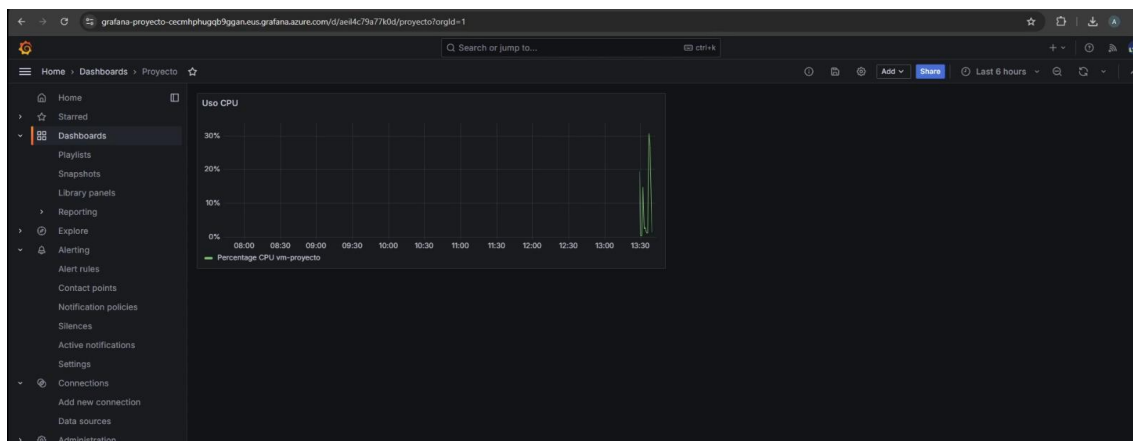
Creamos una visualización y añadimos azure monitor como datasource.



Indicamos como recurso la máquina virtual y la métrica que queremos medir (CPU).



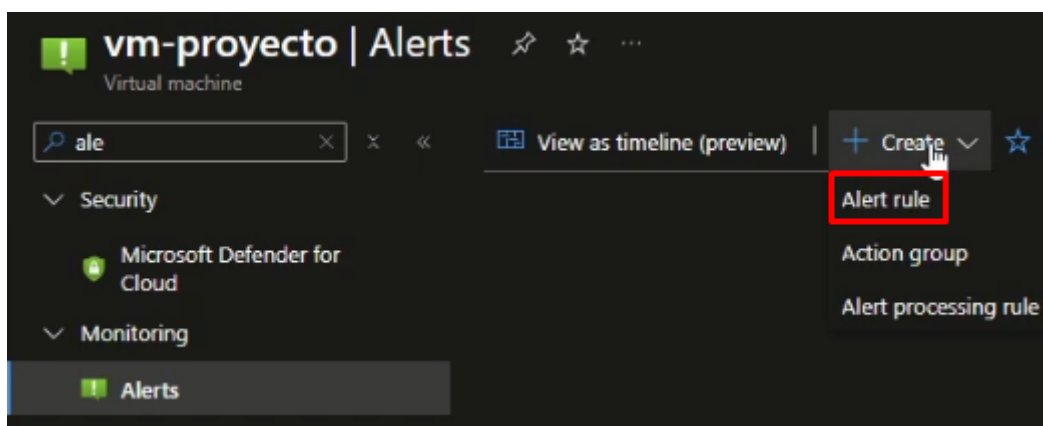
Así quedaría el panel creado, aquí podríamos poner más paneles con más métricas de otros recursos, también tenemos la opción de importar un dashboard ya terminado desde internet para disminuir trabajo.



Configuración Alerta

No he decidido utilizar las alertas de Grafana, ya que requieren la configuración de un servidor SMTP para el envío de correos electrónicos. Es un proceso adicional que no es tan sencillo como las alertas de **Azure Monitor**, que ya vienen configuradas y listas para su uso dentro de la plataforma de Azure, por lo que he decidido utilizar las alertas de **Azure Monitor** debido a su simplicidad y facilidad de integración.

Nos dirigimos a la máquina virtual y en el apartado alertas creamos una nueva alert rule.



Indicamos que queremos una señal de CPU , que salte cuando sobrepase el 30% y que vuelva a analizar pasados 5 minutos.

Create an alert rule

Scope Condition **Actions** Details Tags Review + create

Configure when the alert rule should trigger by selecting a signal and defining its logic.

Signal name

[See all signals](#)

Alert logic

Threshold type ☒ Static ☐ Dynamic

Aggregation type

Value is

Threshold %

When to evaluate

Check every

Lookback period

Creamos un grupo de acción donde indicamos el correo electrónico que recibirá la alerta.

Create an alert rule

Scope Condition **Actions** Details Tags Review + create

An action group is a set of actions that can be applied to an alert rule. [Learn more](#)

Select actions

☒ Use quick actions (preview)
Select one or more of the quick actions.

☐ Use action groups
Add an existing action group or create a new one.

☐ None

Quick actions

Quick actions not configured yet
[Manage quick actions](#)

Use quick actions (preview)

Details

Action group name

Display name

Actions

☒ Email

☐ Email Azure Resource Manager Role

☐ Azure mobile app notification

Por último indicamos el grupo de recursos, un nombre para la alerta y la clasificación según la gravedad.

Create an alert rule

Scope Condition **Actions** **Details** Tags Review + create

Project details

Select the subscription and resource group in which to save the alert rule.

Subscription

Resource group
[Create new](#)

Alert rule details

Severity

Alert rule name

Alert rule description

Advanced options

Comprobación

Accedemos a la máquina virtual desde nuestro terminal a través de SSH con nuestra clave e indicando el usuario y la IP pública de la máquina.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

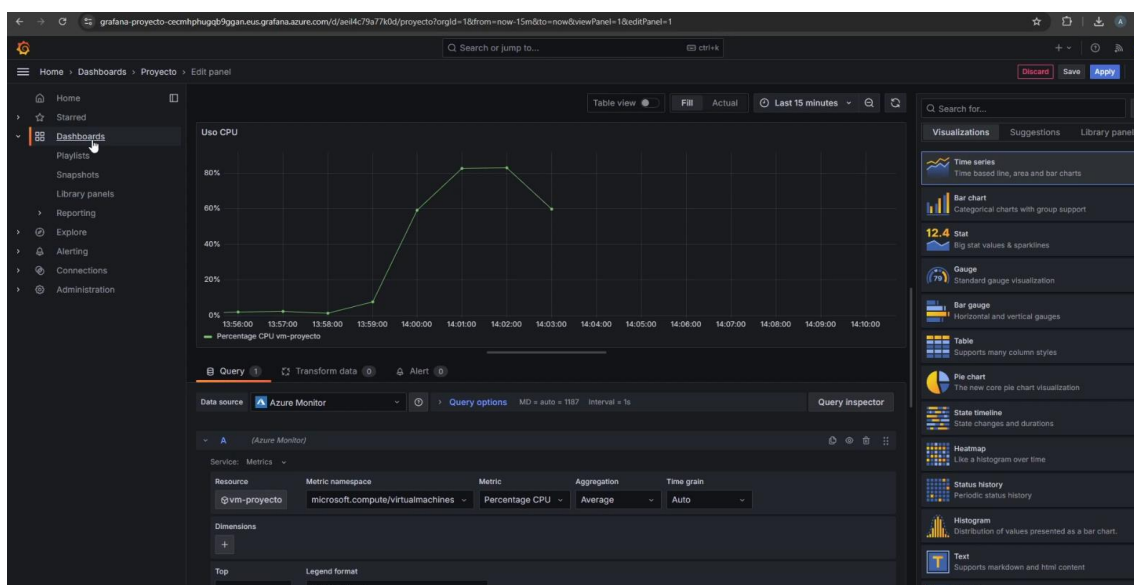
Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\Angel> ssh -i C:\Users\Angel\Downloads\vm-proyecto_key.pem angel@20.119.98.248
The authenticity of host '20.119.98.248 (20.119.98.248)' can't be established.
ED25519 key fingerprint is SHA256:zgEso3lKTRv0NXaDv48/qoZTVeLFJMY2uk9t5I54AY.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

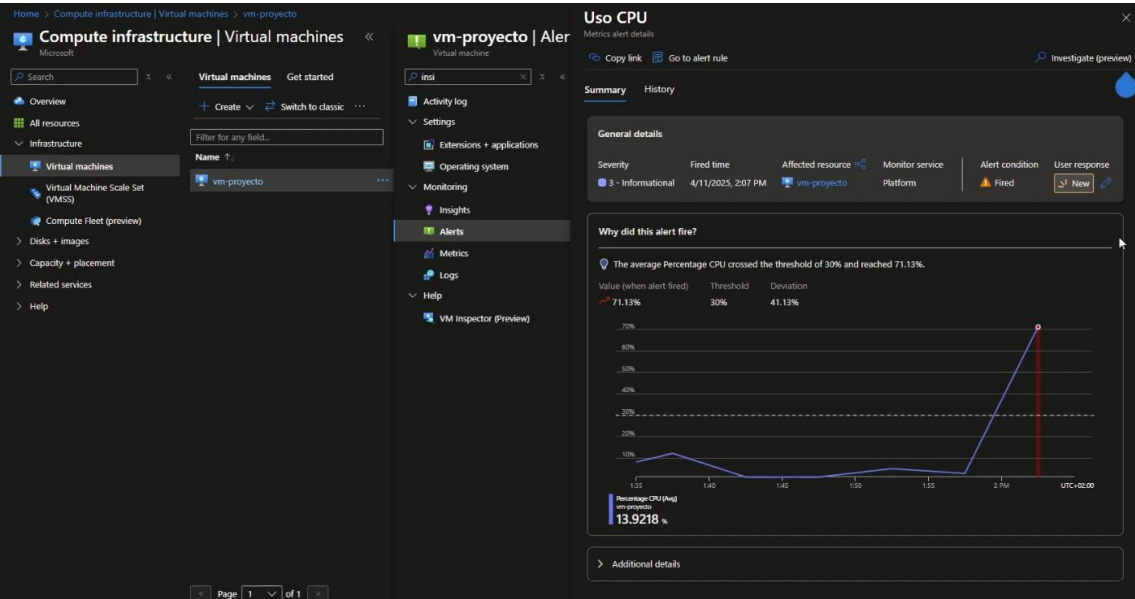
Instalamos y ejecutamos la herramienta stress para simular el alto porcentaje de CPU.

```
angel@vm-proyecto: ~$ stress-ng --cpu 0 --timeout 300s --verbose
stress-ng: debug: [2647] stress-ng 0.15.12
stress-ng: debug: [2647] system: Linux vm-proyecto 6.8.0-1026-azure #31~22.04.1-Ubuntu SMP Thu Mar 20 04:12:50 UTC 2025
x86_64
stress-ng: debug: [2647] RAM total: 842.8M, RAM free: 117.1M, swap free: 0.0
stress-ng: debug: [2647] 1 processor online, 1 processor configured
stress-ng: info: [2647] setting to a 300 second (5 mins, 0.00 secs) run per stressor
stress-ng: info: [2647] dispatching hogs: 1 cpu
stress-ng: debug: [2647] cache allocate: shared cache buffer size: 51200K
stress-ng: debug: [2647] starting stressors
stress-ng: debug: [2647] 1 stressor started
stress-ng: debug: [2648] stress-ng-cpu: started [2648] (instance 0)
stress-ng: debug: [2648] stress-ng-cpu using method 'all'
```

Nos dirigimos a grafana y observamos la subida de CPU, en la parte derecha también podemos ver que se puede cambiar el tipo de gráfico visual.



Volvemos a Azure y vemos que ha saltado la alerta.



Y que nos ha llegado un correo avisándonos.

Fired:Sev3 Azure Monitor Alert Uso CPU on vm-proyecto (microsoft.compute/virtualmachines) at 4/11/2025 12:07:16 PM

Microsoft Azure <azure-noreply@microsoft.com>
Para: Ángel Barahona García

Parte del contenido de este mensaje se ha bloqueado porque el remitente no está en la lista de remitentes seguros. Remitente de confianza Mostrar contenido bloqueado

Microsoft Azure

Fired:Sev3 Azure Monitor Alert Uso CPU on vm-proyecto (microsoft.compute/virtualmachines) at 4/11/2025 12:07:16 PM

[View the alert in Azure Monitor >](#) [Investigate >](#)

Summary

Alert name	Uso CPU
Severity	Sev3
Monitor condition	Fired
Affected resource	vm-proyecto
Resource type	microsoft.compute/virtualmachines
Resource group	proyecto
Monitoring service	Platform

Grafana on-premises

Herramientas necesarias

Node exporter: Se encarga de extraer las métricas del sistema operativo y hardware del servidor.

Prometheus: Se encarga de recopilar y almacenar las métricas del servidor de manera eficiente.

Grafana: Se encarga de visualizar los datos de manera clara y detallada.

Instalación Herramientas

Instalamos las herramientas desde internet con wget y luego las descomprimos.

Node Exporter

```
root@angel-virtual-machine: /home/angel
root@angel-virtual-machine:/home/angel# wget https://github.com/prometheus/node_exporter/releases/download/v1.9.1/node_exporter-1.9.1.linux-amd64.tar.gz
root@angel-virtual-machine:/home/angel# tar xvfz node_exporter-1.9.1.linux-amd64.tar.gz
node_exporter-1.9.1.linux-amd64/
node_exporter-1.9.1.linux-amd64/LICENSE
node_exporter-1.9.1.linux-amd64/NOTICE
node_exporter-1.9.1.linux-amd64/node_exporter
root@angel-virtual-machine:/home/angel#
```

Ejecutamos para iniciarlo.

```
root@angel-virtual-machine:/home/angel/node_exporter-1.9.1.linux-amd64# ./
```

Prometheus

```
root@angel-virtual-machine:/home/angel# wget https://github.com/prometheus/prometheus/releases/download/v2.52.0/prometheus-2.52.0.linux-amd64.tar.gz
```



```

root@angel-virtual-machine:/home/angel# tar xvf prometheus-2.52.0.linux-amd64.tar.gz
prometheus-2.52.0.linux-amd64/
prometheus-2.52.0.linux-amd64/LICENSE
prometheus-2.52.0.linux-amd64/promtool
prometheus-2.52.0.linux-amd64/console_libraries/
prometheus-2.52.0.linux-amd64/console_libraries/prom.lib
prometheus-2.52.0.linux-amd64/console_libraries/menu.lib
prometheus-2.52.0.linux-amd64/prometheus
prometheus-2.52.0.linux-amd64/consoles/
prometheus-2.52.0.linux-amd64/consoles/node-overview.html
prometheus-2.52.0.linux-amd64/consoles/prometheus.html
prometheus-2.52.0.linux-amd64/consoles/node.html
prometheus-2.52.0.linux-amd64/consoles/index.html.example
prometheus-2.52.0.linux-amd64/consoles/node-disk.html
prometheus-2.52.0.linux-amd64/consoles/node-cpu.html
prometheus-2.52.0.linux-amd64/consoles/prometheus-overview.html
prometheus-2.52.0.linux-amd64/prometheus.yml
prometheus-2.52.0.linux-amd64/NOTICE

```

Ejecutamos para iniciarlo.

```

root@angel-virtual-machine:/home/angel/prometheus-2.52.0.linux-amd64# ./prometheus

```

Grafana

```

root@angel-virtual-machine:/home/angel# wget https://dl.grafana.com/oss/release/grafana_9.3.2_amd64.deb

```

```

root@angel-virtual-machine:/home/angel# sudo dpkg -i grafana_9.3.2_amd64.deb

```

Activamos el servicio de grafana.

```

root@angel-virtual-machine:/home/angel# sudo systemctl enable grafana-server
Synchronizing state of grafana-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /lib/systemd/system/grafana-server.service.

```

Configuración Prometheus

Indicamos la IP del equipo que vamos a monitorizar o en este caso localhost junto al puerto de la herramienta que vamos a utilizar en este caso el de node exporter (9100).

```

GNU nano 6.2 prometheus.yml *
# - targets:
#   - alertmanager:9093

# Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
rule_files:
# - "first_rules.yml"
# - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
# The job name is added as a label 'job=<job_name>' to any timeseries scraped from this config.
- job_name: "Proyecto"

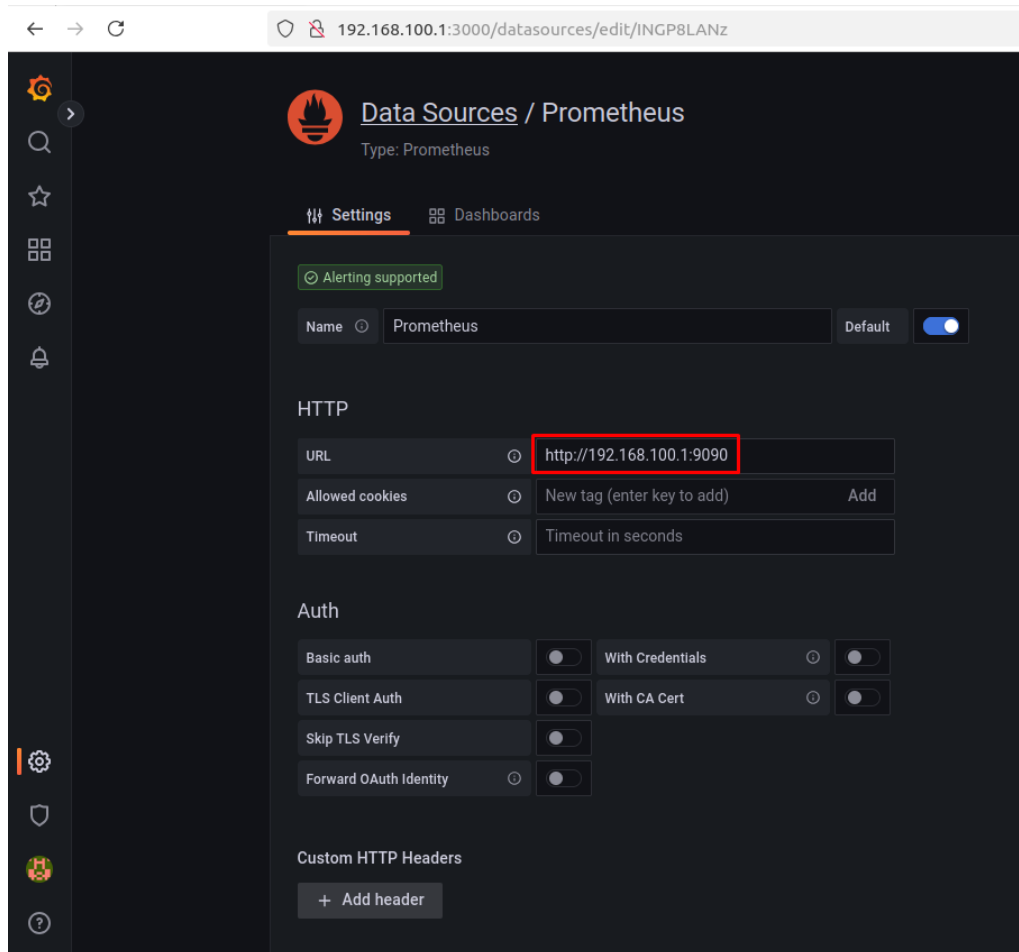
# metrics_path defaults to '/metrics'
# scheme defaults to 'http'.

static_configs:
- targets: ["localhost:9100"]

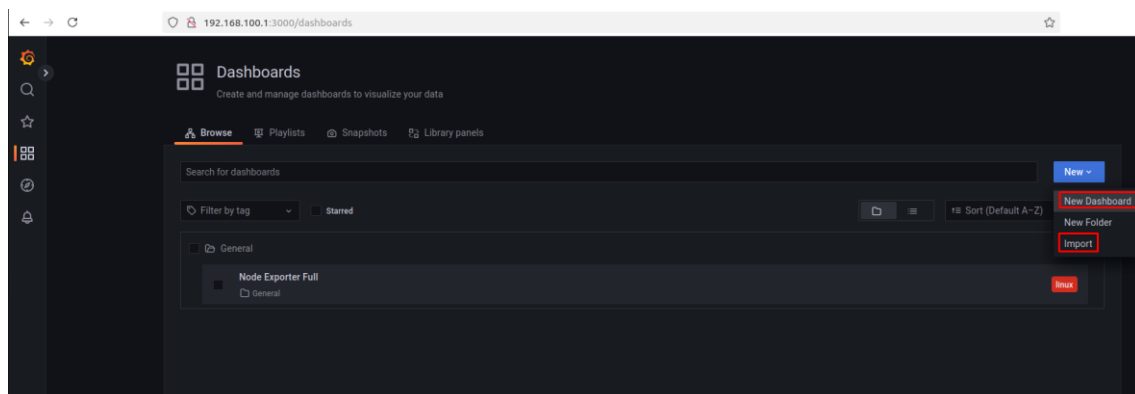
```

Configuración Grafana

Vamos a añadir un datasource para ello ponemos la ip de nuestro servidor y el puerto 9090 que es el de prometheus, nos metemos en configuration > Data Sources.



Podremos crear nuestro propio dashboard o importar uno desde internet.



Dashboard final

