

DevOps

PIN FINAL

Grupo 5

Emilio Pascutti - emiliopascutti2164@gmail.com

Natalio Cardozo - nathaliocardozo@gmail.com

Julio Gonzalez - aresden113@gmail.com

Jonathan David Martino - jonamartino@gmail.com

Gonzalo Martín Montalvo - montalvog@gmail.com

Objetivos:

Terraform - EKS – AWS – Prometheus – Grafana

El objetivo de este PIN, será desarrollar una Infraestructura como Código (IaaC) de Terraform, para desplegar un clúster de kubernetes en AWS, monitoreado con Prometheus y Grafana.

Flujo de Trabajo

/PINFINAL

- └─ Creación de cuenta IAM con AmazonEC2FullAccess.
- └─ Configuración de Terraform para el Clúster EKS.
- └─ Deploy de Prometheus y Grafana con Helm automático.
- └─ Verificación de entornos
- └─ Configuración de Prometheus como source de Grafana y configuración de Dashboards.
- └─ Clean up
- └─ Análisis y conclusiones

Creación de usuario IAM en AWS para EC2

Rol IAM con policy de AmazonEC2FullAccess

Nombre de usuario IAM: manzana_devops

BWS

Search

Global ▼

Emilio Pascutti

IAM > Users > Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Step 4
Retrieve password

Specify user details

User details

User name

manzana_devops

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + , . @ _ - (hyphen)

☒ Provide user access to the AWS Management Console - optional

If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

Are you providing console access to a person?

User type

☐ Specify a user in Identity Center - Recommended

We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.

☒ I want to create an IAM user

We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.

Console password

☒ Autogenerated password

You can view the password after you create the user.

☐ Custom password

Enter a custom password for the user.

☐ Show password

☐ Users must create a new password at next sign-in - Recommended

Users automatically get the [IAMUserChangePassword](#) policy to allow them to change their own password.

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user.

[Learn more](#)

Cancel

Next

cloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

DevOps group:

AWS

Q Search

[Option+S]

Global

Emilio Pascutti

devops

Identity and Access Management (IAM)

Q Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Resource access management

Access reports

Access Analyzer

External access

Unused access

Analyzer settings

Credential report

Organization activity

Service control policies

Resource control policies

IAM Identity Center

AWS Organizations

devops

Summary

User group name
devops

Creation time
December 16, 2024, 21:53 (UTC-03:00)

ARN
arn:aws:iam::905418008126:group/devops

Users

Permissions

Access Advisor

Permissions policies (4)

You can attach up to 10 managed policies.

Q Search

Filter by Type
All types

Policy name	Type	Attached entities
<input type="checkbox"/> AdministratorAccess	AWS managed - job function	2
<input type="checkbox"/> AmazonDynamoDBFullAccess	AWS managed	2
<input type="checkbox"/> AmazonEC2FullAccess	AWS managed	2
<input type="checkbox"/> AmazonS3FullAccess	AWS managed	2

aws

Search

[Option+S]

Global

Emilio Pascutti

IAM

Users

Create user

Step 1
Specify user details

Step 2
Set permissions

Step 3
Review and create

Step 4
Retrieve password

Review and create

Review your choices. After you create the user, you can view and download the autogenerated password, if enabled.

User details

User name
manzana_devops

Console password type
Autogenerated

Require password reset
No

Permissions summary

Name

Type

Used as

devops

Group

Permissions group

Tags - optional

Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel

Previous

Create user

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Search

[Option+S]

Global

Emilio Pascutti

IAM

Users

manzana_devops

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Root access management

Access reports

Access Analyzer

External access

Unused access

Analyzer settings

Credential report

Organization activity

Service control policies

Resource control policies

IAM Identity Center

AWS Organizations

Passkey MFA device assigned

As a security best practice, we encourage registering multiple devices in the case that your primary method is lost, disabled, or unavailable. Choose any of your MFA devices to use to sign in to your AWS account.

Delete

manzana_devops

Info

Delete

Summary

ARN
arn:aws:iam::905418008126:user/manzana_devops

Console access
Enabled with MFA

Access key 1
Create access key

Created
March 03, 2025, 22:54 (UTC-03:00)

Last console sign-in
Never

Permissions

Groups (1)

Tags

Security credentials

Last Accessed

Console sign-in

Manage console access

Console sign-in link
https://905418008126.signin.aws.amazon.com/console

Console password
Updated 1 minute ago (2025-03-03 22:54 GMT-3)

Last console sign-in
Never

Multi-factor authentication (MFA) (1)

Remove Resync Assign MFA device

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more

Type

Identifier

Certifications

Created on

Passkeys and security keys

arn:aws:iam::905418008126:user/manzana_devops/macOs-XG6MHUAN4VH3PC6RTE3JUXK63I

Mon Mar 03 2025

Access keys (0)

Create access key

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more

No access keys. As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. Learn more

Create access key

SSH public keys for AWS CodeCommit (0)

Actions Upload SSH public key

Use SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. Learn more

SSH Key ID

Uploaded

Status

No SSH public keys

Upload SSH public key

HTTPS Git credentials for AWS CodeCommit (0)

Actions Generate credentials

Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can have a maximum of 2 sets of credentials (active or inactive) at a time. Learn more

User name

Created

Status

No credentials

Generate credentials

Credentials for Amazon Keyspaces (for Apache Cassandra) (0)

Actions Generate credentials

Generate a user name and password you can use to authenticate to Amazon Keyspaces. You can have a maximum of two sets of credentials (active or inactive) at a time. Learn more

User name

Created

Status

No credentials

Generate credentials

X.509 Signing certificates (0)

Actions Upload Create X.509 certificate

Use X.509 certificates to make secure SOAP-protocol requests to some AWS services. You can have a maximum of two X.509 certificates (active or inactive) at a time. Learn more

Creation time

Thumbprint

Status

No X.509 certificates

Create X.509 certificate

CloudShell

Feedback

© 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Configuración de Terraform para EKS:

Terraform:

```
echo "deb [arch=$(dpkg --print-architecture)
signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com
$(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
```

```
sudo apt update && sudo apt install terraform
```

```

Selecting previously unselected package terraform.
(Reading database ... 90812 files and directories currently installed.)
Preparing to unpack .../terraform_1.11.0-1_arm64.deb ...
Unpacking terraform (1.11.0-1) ...
Setting up terraform (1.11.0-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
manzana@devops:~$ terraform -v
Terraform v1.11.0
on linux_arm64
manzana@devops:~$ █

```

AWS CLI:

kubectrl para ARM:

```
curl -LO "https://dl.k8s.io/release/$(curl -L -s
https://dl.k8s.io/release/stable.txt)/bin/linux/arm64/kubectl"
```

```
echo "$(cat kubect1.sha256) kubect1" | sha256sum --check
```

```
sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
```

```
kubectl version --client
```

```
manzanar@devops:~$ curl -XO "https://dl.k8s.io/release/stable.txt/bin/linux/arm64/kubectl"
% Total % Received % Xferd Average Speed Time Time Current
                                Dload Upload Total Spent Left Speed
100 138 100 138 0 0 525 0 0 0 0 0 0 0 0 0 0 0 0 0 0
100 53.2M 100 53.2M 0 0 3988K 0 0 0 0 0 0 0 0 0 0 0 0 0 0
manzanar@devops:~$ curl -LO "https://dl.k8s.io/release/stable.txt/bin/linux/arm64/kubectl.sha256"
% Total % Received % Xferd Average Speed Time Time Current
                                Dload Upload Total Spent Left Speed
100 138 100 138 0 0 528 0 0 0 0 0 0 0 0 0 0 0 0 0
100 64 100 64 0 0 176 0 0 0 0 0 0 0 0 0 0 0 0 0
manzanar@devops:~$ echo "$(cat kubectl.sha256) kubectl" | sha256sum --check
kubectl: OK
manzanar@devops:~$ sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
manzanar@devops:~$ kubectl version --client
Client Version: v1.32.0
Kustomize Version: v5.5.0
```

Repositorio de Github:

Creamos un repositorio con todas las configuraciones del IaaS donde detallamos los pasos para el despliegue del Clúster:

<https://github.com/manzana2164/pin-final-tf.git>

Inicialización y aplicación de Terraform:

`terraform init`

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
manzana@devops:~/pin-final-tf$ terraform init
Initializing the backend...
Initializing modules...
- Downloading registry.terraform.io/terraform-aws-modules/eks/aws 18.31.2 for eks...
- eks in .terraform/modules/eks
- eks.eks_managed_node_group in .terraform/modules/eks/modules/eks-managed-node-group
- eks.eks_managed_node_group.user_data in .terraform/modules/eks/modules/user_data
- eks.fargate_profile in .terraform/modules/eks/modules/fargate-profile
- Downloading registry.terraform.io/terraform-aws-modules/kms/aws 1.0.2 for eks.kms...
- eks.eks in .terraform/modules/eks.kms
- eks.self_managed_node_group in .terraform/modules/eks/modules/self-managed-node-group
- eks.self_managed_node_group.user_data in .terraform/modules/eks/modules/user_data
- Downloading registry.terraform.io/terraform-aws-modules/vpc/aws 3.19.0 for vpc...
- vpc in .terraform/modules/vpc
Initializing provider plugins...
- Finding hashicorp/helm versions matching "~>= 2.0"...
- Finding hashicorp/aws versions matching "~>= 3.72.0, <= 3.73.0, <= 4.0"...
- Finding hashicorp/kubernetes versions matching "~>= 2.0, <= 2.10.0"...
- Finding hashicorp/tls versions matching "~>= 3.0.0"...
- Finding hashicorp/cloudinit versions matching "~>= 2.0.0"...
- Installing hashicorp/helm v2.17.0...
- Installed hashicorp/helm v2.17.0 (signed by HashiCorp)
- Installing hashicorp/aws v4.67.0...
- Installed hashicorp/aws v4.67.0 (signed by HashiCorp)
- Installing hashicorp/kubernetes v2.36.0...
- Installed hashicorp/kubernetes v2.36.0 (signed by HashiCorp)
- Installing hashicorp/tls v4.0.6...
- Installed hashicorp/tls v4.0.6 (signed by HashiCorp)
- Installing hashicorp/cloudinit v2.3.6...
- Installed hashicorp/cloudinit v2.3.6 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
manzana@devops:~/pin-final-tf$
```

```
+ tag_specifications {
+   resource_type = "instance"
+   tags = {
+     "Environment" = "Dev"
+     "Name"         = "mundose-node-group"
+     "Project"      = "PINFINAL"
+   }
+ }
+ tag_specifications {
+   resource_type = "network-interface"
+   tags = {
+     "Environment" = "Dev"
+     "Name"         = "mundose-node-group"
+     "Project"      = "PINFINAL"
+   }
+ }
+ tag_specifications {
+   resource_type = "volume"
+   tags = {
+     "Environment" = "Dev"
+     "Name"         = "mundose-node-group"
+     "Project"      = "PINFINAL"
+   }
+ }
}

# module.eks.module.eks_managed_node_group["mundose-node-group"].aws_security_group.this[0] will be created
+ resource "aws_security_group" "this" {
+   arn              = (known after apply)
+   description      = "EKS managed node group security group"
+   egress            = (known after apply)
+   id               = (known after apply)
+   ingress           = (known after apply)
+   name             = (known after apply)
+   name_prefix      = "mundose-node-group-eks-node-group-"
+   owner_id         = (known after apply)
+   revoke_rules_on_delete = false
+   tags = {
+     "Environment" = "Dev"
+     "Name"         = "mundose-node-group-eks-node-group"
+     "Project"      = "PINFINAL"
+   }
+   tags_all = {
+     "Environment" = "Dev"
+     "Name"         = "mundose-node-group-eks-node-group"
+     "Project"      = "PINFINAL"
+   }
+   vpc_id = (known after apply)
+ }

Plan: 49 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ cluster_endpoint = (known after apply)
+ cluster_name     = (known after apply)
+ nginx_service_endpoint = (known after apply)
```

Después del terraform apply, chequeamos que es accesible el cluster:

```
manzana@devops:~$ aws eks --region us-east-1 update-kubeconfig --name eks-cluster-mundose
Added new context arns:aws:eks:us-east-1:985418808126:cluster/eks-cluster-mundose to /home/manzana/.kube/config
manzana@devops:~$ kubectl config current-context
arns:aws:eks:us-east-1:985418808126:cluster/eks-cluster-mundose
manzana@devops:~$ kubectl get pods
NAME                                STATUS    ROLES    AGE   VERSION
ip-18-81-87-ec2.internal            Ready     <none>   20m   v1.32.1-eks-5d632ec
manzana@devops:~$ kubectl get pods -A
NAMESPACE   NAME                                READY    STATUS    RESTARTS   AGE
kube-system  aws-node-6h5mk                    2/2      Running   0           21m
kube-system  coredns-6b9575c64c-1lzh          1/1      Running   0           23m
kube-system  coredns-6b9575c64c-nhv87         1/1      Running   0           23m
kube-system  kube-proxy-8285f                 0/1      Pending   0           21m
monitoring   grafana-release-567d4456f5-5x9mx  0/1      Pending   0           6s29s
monitoring   prometheus-release-alertmanager-4  0/1      Pending   0           6m15s
monitoring   prometheus-release-kube-state-metrics-59dc9d6fb-kjjc9  0/1      Pending   0           6m14s
monitoring   prometheus-release-prometheus-node-exporter-tx5dp      0/1      Pending   0           6m14s
monitoring   prometheus-release-prometheus-pushgateway-7f44fcd957-ubpzb  0/1      Pending   0           6m14s
monitoring   prometheus-release-server-796d7bdc7-7f5d4              0/2      Pending   0           6m14s
manzana@devops:~$ kubectl cluster-info
Kubernetes control plane is running at https://4E84C37360703F207083E8DA1AAE3E45.g7.us-east-1.eks.amazonaws.com
CoreDNS is running at https://4E84C37360703F207083E8DA1AAE3E45.g7.us-east-1.eks.amazonaws.com/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy
To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
manzana@devops:~$
```

Accedemos al cluster:

```
aws eks --region us-east-1 update-kubeconfig --name eks-cluster-mundose
```

```
kubectl get pods -A
```

Chequeamos los LoadBalancers:

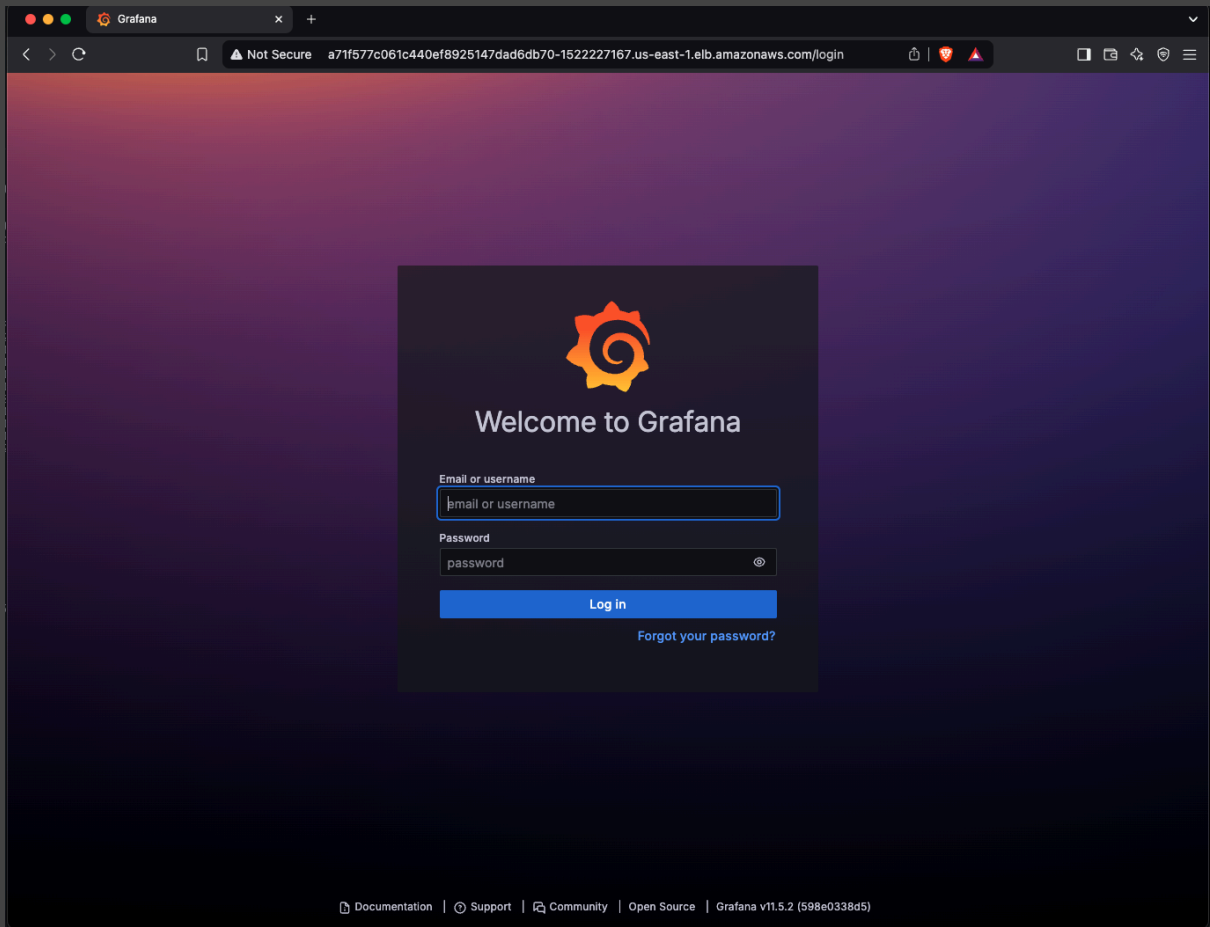
```
kubectl get svc -n monitoring
```

```
kubectl get svc -n default
```

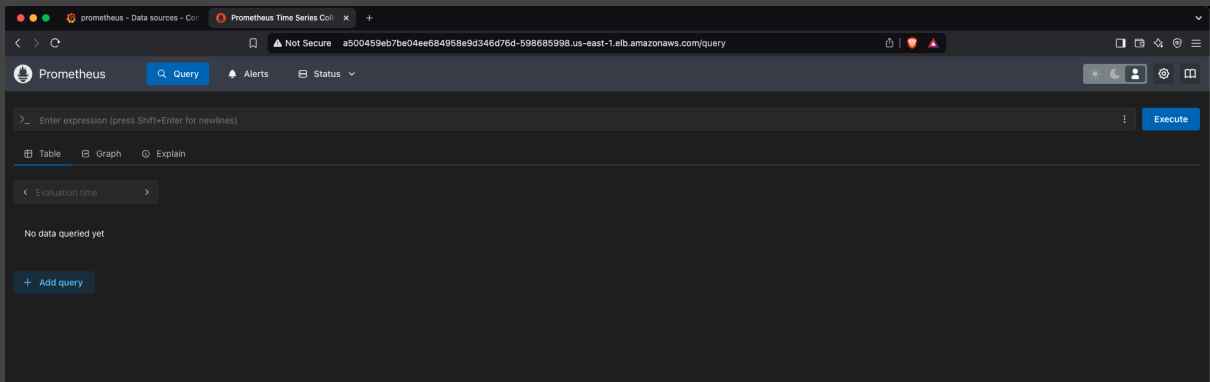
```
manzana@devops:~$ aws eks --region us-east-1 update-kubeconfig --name eks-cluster-mundose
Updated context arns:aws:eks:us-east-1:985418808126:cluster/eks-cluster-mundose in /home/manzana/.kube/config
manzana@devops:~$ kubectl get pods -A
NAMESPACE   NAME                                READY    STATUS    RESTARTS   AGE
kube-system  aws-node-2g6pc                    2/2      Running   0           3m37s
kube-system  coredns-6b9575c64c-6dw7v         1/1      Running   0           3m37s
kube-system  coredns-6b9575c64c-9h4ht         1/1      Running   0           3m37s
kube-system  kube-proxy-fghss                 1/1      Running   0           10s
monitoring   grafana-release-5d97d9b765-248x7  0/1      Pending   0           2m38s
monitoring   prometheus-release-alertmanager-0  0/1      Pending   0           2m23s
monitoring   prometheus-release-kube-state-metrics-59dc9d6fb-zv5hp  0/1      Pending   0           2m24s
monitoring   prometheus-release-prometheus-node-exporter-zzbrz      1/1      Running   0           100s
monitoring   prometheus-release-prometheus-pushgateway-7f44fcd957-p54jd  1/1      Running   0           2m24s
monitoring   prometheus-release-server-8467c6fb4c-lfx4c             2/2      Pending   0           2m24s
manzana@devops:~$ kubectl get pods -n monitoring
NAMESPACE   NAME                                READY    STATUS    RESTARTS   AGE
kube-system  aws-node-2g6pc                    2/2      Running   0           3m37s
kube-system  coredns-6b9575c64c-6dw7v         1/1      Running   0           3m37s
kube-system  coredns-6b9575c64c-9h4ht         1/1      Running   0           3m37s
kube-system  kube-proxy-fghss                 1/1      Running   0           10s
monitoring   grafana-release-5d97d9b765-248x7  0/1      Pending   0           2m38s
monitoring   prometheus-release-alertmanager-0  0/1      Pending   0           2m23s
monitoring   prometheus-release-kube-state-metrics-59dc9d6fb-zv5hp  0/1      Pending   0           2m24s
monitoring   prometheus-release-prometheus-node-exporter-zzbrz      1/1      Running   0           100s
monitoring   prometheus-release-prometheus-pushgateway-7f44fcd957-p54jd  1/1      Running   0           2m24s
monitoring   prometheus-release-server-8467c6fb4c-lfx4c             2/2      Pending   0           2m24s
manzana@devops:~$ kubectl get svc -n monitoring
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
grafana-release                     LoadBalancer       172.20.136.11   ad8dc9142697346d5bdc3c6c5ccbd7-1735530287.us-east-1.elb.amazonaws.com  80:38172/TCP  3m10s
prometheus-release-alertmanager      ClusterIP            172.20.110.50   <none>            9093/TCP          3m4s
prometheus-release-alertmanager-headless ClusterIP            None            <none>            9093/TCP          3m4s
prometheus-release-kube-state-metrics ClusterIP            172.20.134.83   <none>            8888/TCP          3m4s
prometheus-release-prometheus-node-exporter ClusterIP            172.20.162.45   <none>            9100/TCP          3m4s
prometheus-release-prometheus-pushgateway ClusterIP            172.20.171.198   <none>            9091/TCP          3m4s
prometheus-release-server            LoadBalancer       172.20.222.224   a580459eb7be04ee684958e9d346d76d-598685998.us-east-1.elb.amazonaws.com  80:31326/TCP  3m4s
manzana@devops:~$ kubectl get svc -n default
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
grafana-release                     LoadBalancer       172.20.123.199   a71f577c061c440ef8925147dad6db70-1522227167.us-east-1.elb.amazonaws.com  80:30945/TCP  4m5s
prometheus-release-alertmanager      ClusterIP            172.20.215.82   <none>            9093/TCP          3m59s
prometheus-release-alertmanager-headless ClusterIP            None            <none>            9093/TCP          3m59s
prometheus-release-kube-state-metrics ClusterIP            172.20.101.54   <none>            8888/TCP          3m59s
prometheus-release-prometheus-node-exporter ClusterIP            172.20.23.250   <none>            9100/TCP          3m59s
prometheus-release-prometheus-pushgateway ClusterIP            172.20.55.194   <none>            9091/TCP          3m59s
prometheus-release-server            LoadBalancer       172.20.69.210   a19ce557ba0fe448bb28b32cf1d9993d-1458695701.us-east-1.elb.amazonaws.com  80:31766/TCP  3m59s
kubernetes                          ClusterIP            172.20.0.1      <none>            443/TCP           5m2s
nginx-logs-service                  LoadBalancer       172.20.73.187   a044932d377254dcda08dbcf4f117ab-980648276.us-east-1.elb.amazonaws.com  80:32710/TCP  3m29s
manzana@devops:~$
```

```
manzana@devops:~$ kubectl get svc -n monitoring
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
grafana-release                     LoadBalancer       172.20.123.199   a71f577c061c440ef8925147dad6db70-1522227167.us-east-1.elb.amazonaws.com  80:30945/TCP  4m5s
prometheus-release-alertmanager      ClusterIP            172.20.215.82   <none>            9093/TCP          3m59s
prometheus-release-alertmanager-headless ClusterIP            None            <none>            9093/TCP          3m59s
prometheus-release-kube-state-metrics ClusterIP            172.20.101.54   <none>            8888/TCP          3m59s
prometheus-release-prometheus-node-exporter ClusterIP            172.20.23.250   <none>            9100/TCP          3m59s
prometheus-release-prometheus-pushgateway ClusterIP            172.20.55.194   <none>            9091/TCP          3m59s
prometheus-release-server            LoadBalancer       172.20.69.210   a19ce557ba0fe448bb28b32cf1d9993d-1458695701.us-east-1.elb.amazonaws.com  80:31766/TCP  3m59s
kubernetes                          ClusterIP            172.20.0.1      <none>            443/TCP           5m2s
nginx-logs-service                  LoadBalancer       172.20.73.187   a044932d377254dcda08dbcf4f117ab-980648276.us-east-1.elb.amazonaws.com  80:32710/TCP  3m29s
manzana@devops:~$ kubectl get svc -n default
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
grafana-release                     LoadBalancer       172.20.123.199   a71f577c061c440ef8925147dad6db70-1522227167.us-east-1.elb.amazonaws.com  80:30945/TCP  4m5s
prometheus-release-alertmanager      ClusterIP            172.20.215.82   <none>            9093/TCP          3m55s
prometheus-release-alertmanager-headless ClusterIP            None            <none>            9093/TCP          3m55s
prometheus-release-kube-state-metrics ClusterIP            172.20.101.54   <none>            8888/TCP          3m55s
prometheus-release-prometheus-node-exporter ClusterIP            172.20.23.250   <none>            9100/TCP          3m55s
prometheus-release-prometheus-pushgateway ClusterIP            172.20.55.194   <none>            9091/TCP          3m55s
prometheus-release-server            LoadBalancer       172.20.69.210   a19ce557ba0fe448bb28b32cf1d9993d-1458695701.us-east-1.elb.amazonaws.com  80:31766/TCP  3m55s
manzana@devops:~$
```

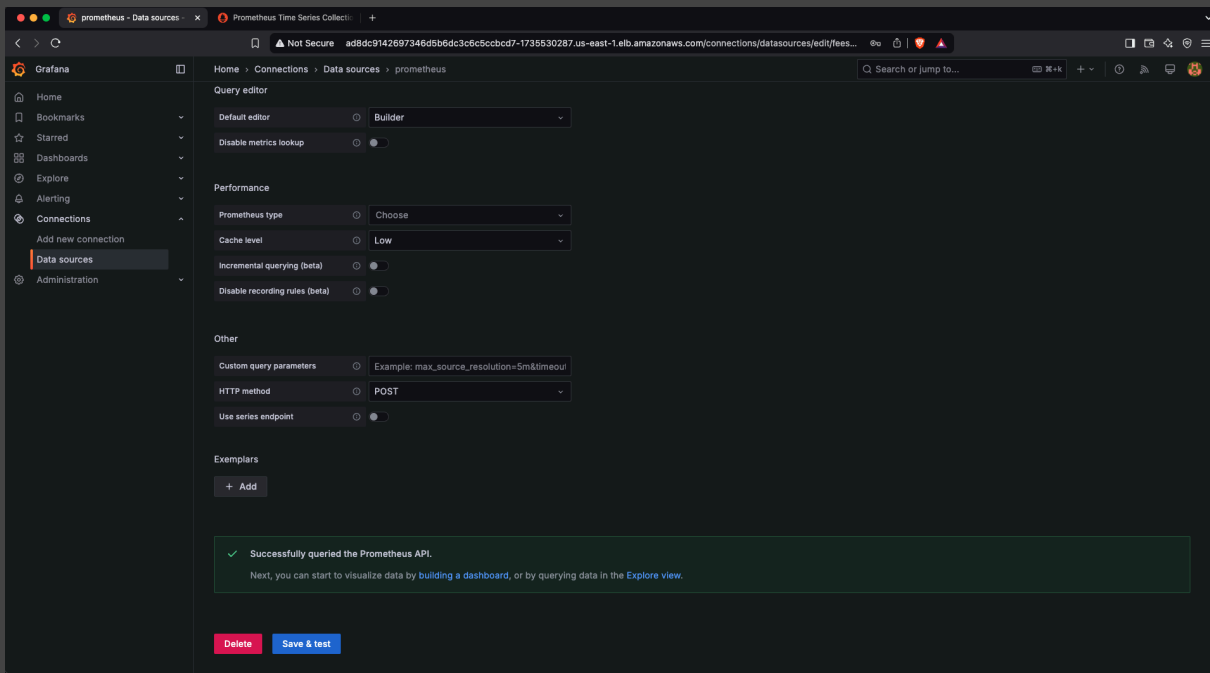
Accedemos a Grafana:
ad8dc9142697346d5b6dc3c6c5ccbcd7-1735530287.us-east-1.elb.amazonaws.com



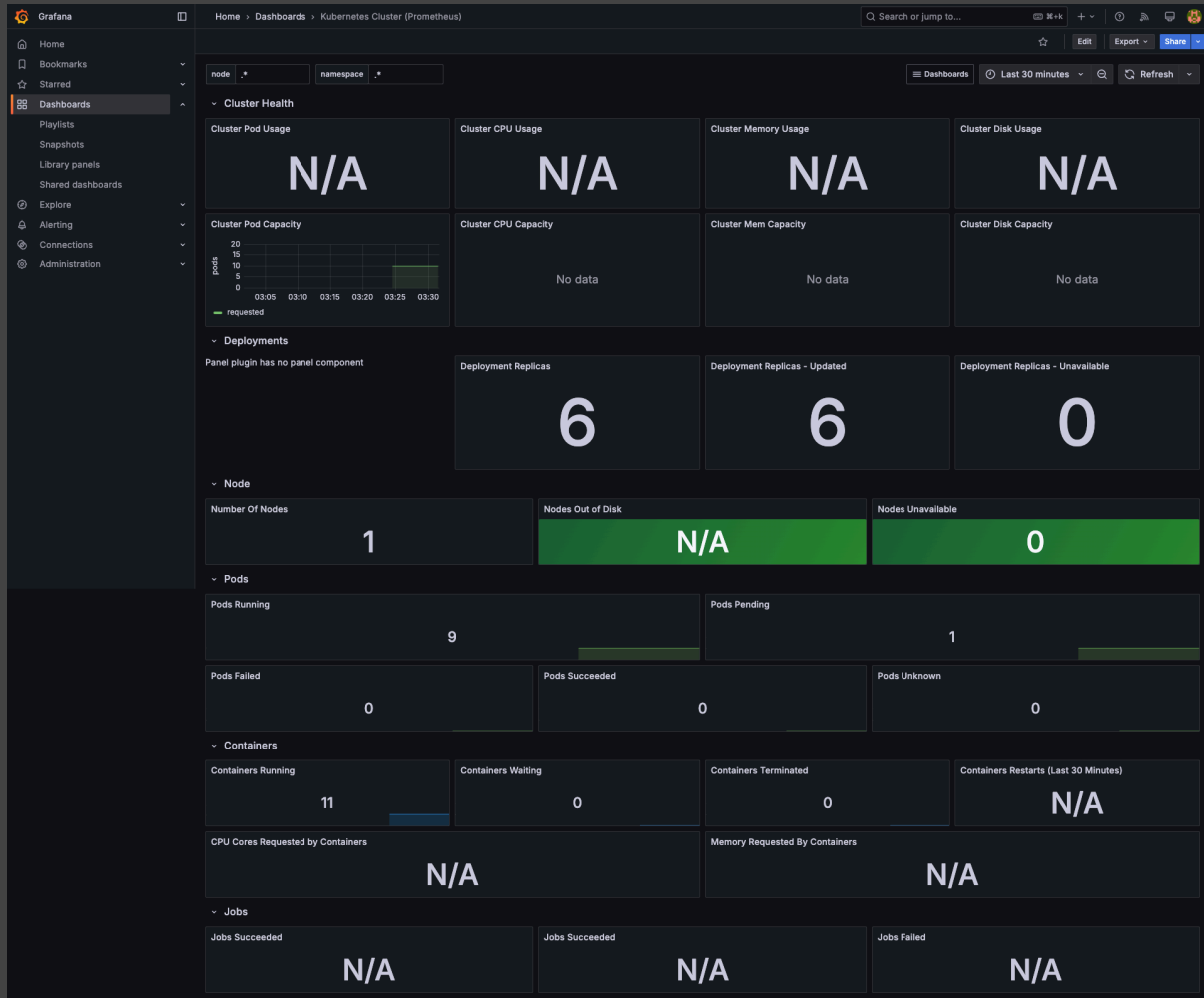
Accedemos a Prometheus:
a500459eb7be04ee684958e9d346d76d-598685998.us-east-1.elb.amazonaws.com



Agregamos la source de Prometheus:



Creamos los dashboards por importación con los códigos 3119 y 6417



Luego de verificar los dashboards, procedemos con un **terraform destroy** a eliminar la instancia EC2

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS  PROBLEMS

- "Environment" = "Dev"
- "Name" = "mundose-node-group"
- "Project" = "PINFINAL"
} -> null

- tag_specifications {
- resource_type = "network-interface" -> null
- tags = {
- "Environment" = "Dev"
- "Name" = "mundose-node-group"
- "Project" = "PINFINAL"
- "Project" = "PINFINAL"
} -> null
}

- tag_specifications {
- resource_type = "volume" -> null
- tags = {
- "Environment" = "Dev"
- "Name" = "mundose-node-group"
- "Project" = "PINFINAL"
} -> null
}

}

# module.eks.module.eks_managed_node_group["mundose-node-group"].aws_security_group.this[0] will be destroyed
- resource "aws_security_group" "this" {
- arn = "arn:aws:ec2:us-east-1:905418008126:security-group/sg-0334629b134ea682a" -> null
- description = "EKS managed node group security group" -> null
- egress = [] -> null
- id = "sg-0334629b134ea682a" -> null
- ingress = [] -> null
- name = "mundose-node-group-eks-node-group-20250304061409002500000000" -> null
- name_prefix = "mundose-node-group-eks-node-group-" -> null
- owner_id = "905418008126" -> null
- revoke_rules_on_delete = false -> null
- tags = {
- "Environment" = "Dev"
- "Name" = "mundose-node-group-eks-node-group"
- "Project" = "PINFINAL"
- "Project" = "PINFINAL"
} -> null
- tags_all = {
- "Environment" = "Dev"
- "Name" = "mundose-node-group-eks-node-group"
- "Project" = "PINFINAL"
- "Project" = "PINFINAL"
} -> null
- vpc_id = "vpc-0f25bfc6b375d2c06" -> null
}

Plan: 0 to add, 0 to change, 49 to destroy.

Changes to Outputs:
- cluster_endpoint = "https://928407c1c365eca877e2580c520a3e2f.gr7.us-east-1.eks.amazonaws.com" -> null
- cluster_name = "eks-cluster-mundose" -> null
- nginx_service_endpoint = "a04932d377254cdc808dc84f117ab-980648276.us-east-1.elb.amazonaws.com" -> null

Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes
SSH: 192.168.0.150 0 0 0 4
```

```
OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS  PROBLEMS

module.eks.module.eks_managed_node_group["mundose-node-group"].aws_iam_role.this[0]: Destruction complete after 1s
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 18s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 28s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 38s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 48s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 58s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 1m0s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 1m18s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 1m28s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 1m38s elapsed]
module.eks.aws_eks_cluster.this[0]: Still destroying... [id=eks-cluster-mundose, 1m48s elapsed]
module.eks.aws_eks_cluster.this[0]: Destruction complete after 1m47s
module.eks.aws_security_group_rule.node["ingress_self_coredns_tcp"]: Destroying... [id=sgrule-3848675672]
module.eks.aws_security_group_rule.node["ingress_cluster_443"]: Destroying... [id=sgrule-3813858990]
module.eks.aws_security_group_rule.node["egress_https"]: Destroying... [id=sgrule-1836271345]
module.vpc.aws_subnet.private[1]: Destroying... [id=subnet-0988b1d837dd289f8]
module.eks.aws_iam_role_policy_attachment.this["arn:aws:iam::aws:policy/AmazonEKSClusterPolicy"]: Destroying... [id=eks-cluster-mundose-cluster-20250384061354613400000002-20250384061357077700000006]
module.eks.aws_security_group_rule.node["egress_ntp_udp"]: Destroying... [id=sgrule-2625986482]
module.eks.aws_security_group_rule.node["egress_cluster_443"]: Destroying... [id=sgrule-845185654]
module.eks.aws_cloudwatch_log_group.this[0]: Destroying... [id=/aws/eks/eks-cluster-mundose/cluster]
module.eks.aws_security_group_rule.node["ingress_cluster_kubelet"]: Destroying... [id=sgrule-63882393]
module.eks.aws_iam_role_policy_attachment.this["arn:aws:iam::aws:policy/AmazonEKSClusterPolicy"]: Destruction complete after 0s
module.eks.aws_cloudwatch_log_group.this[0]: Destruction complete after 8s
module.eks.aws_security_group_rule.cluster["ingress_nodes_443"]: Destroying... [id=sgrule-283547140]
module.eks.aws_security_group_rule.node["egress_self_coredns_udp"]: Destroying... [id=sgrule-458798076]
module.eks.aws_security_group_rule.node["egress_https"]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["egress_self_coredns_tcp"]: Destroying... [id=sgrule-2495152346]
module.eks.aws_security_group_rule.cluster["egress_nodes_kubelet"]: Destruction complete after 1s
module.vpc.aws_subnet.private[0]: Destroying... [id=subnet-8dfc22a26619868e]
module.vpc.aws_subnet.private[1]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["ingress_self_coredns_udp"]: Destroying... [id=sgrule-1894288855]
module.eks.aws_security_group_rule.cluster["ingress_nodes_443"]: Destroying... [id=sgrule-2967431644]
module.eks.aws_security_group_rule.cluster["ingress_nodes_443"]: Destruction complete after 2s
module.eks.aws_iam_role_policy_attachment.this["arn:aws:iam::aws:policy/AmazonEKSVPResourceController"]: Destroying... [id=eks-cluster-mundose-cluster-20250384061354613400000002-20250384061357223900000007]
module.vpc.aws_subnet.private[0]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["egress_ntp_tcp"]: Destroying... [id=sgrule-3793355759]
module.eks.aws_iam_role_policy_attachment.this["arn:aws:iam::aws:policy/AmazonEKSVPResourceController"]: Destruction complete after 0s
module.eks.aws_iam_role.this[0]: Destroying... [id=eks-cluster-mundose-cluster-20250384061354613400000002]
module.eks.aws_security_group_rule.node["ingress_cluster_443"]: Destruction complete after 3s
module.eks.aws_security_group_rule.cluster["ingress_nodes_443"]: Destruction complete after 1s
module.eks.aws_iam_role.this[0]: Destruction complete after 1s
module.eks.aws_security_group_rule.node["ingress_cluster_kubelet"]: Destruction complete after 4s
module.eks.aws_security_group_rule.node["egress_ntp_udp"]: Destruction complete after 4s
module.eks.aws_security_group_rule.node["egress_cluster_443"]: Destruction complete after 5s
module.eks.aws_security_group_rule.node["egress_self_coredns_udp"]: Destruction complete after 6s
module.eks.aws_security_group_rule.node["egress_self_coredns_tcp"]: Destruction complete after 6s
module.eks.aws_security_group_rule.node["ingress_self_coredns_udp"]: Destruction complete after 7s
module.eks.aws_security_group_rule.node["egress_ntp_tcp"]: Destruction complete after 7s
module.eks.aws_security_group.cluster[0]: Destroying... [id=sg-086bda5dd4016438]
module.eks.aws_security_group.node[0]: Destroying... [id=sg-086bda5dd4016438]
module.eks.aws_security_group.cluster[0]: Destruction complete after 1s
module.eks.aws_security_group.node[0]: Destruction complete after 1s
module.vpc.aws_vpc.this[0]: Destroying... [id=vpc-0f25bfceb375d2c86]
module.vpc.aws_vpc.this[0]: Destruction complete after 1s

Destroy complete! Resources: 49 destroyed.
nanzana@devops:~/pin-final-tfs
```

Análisis y Conclusiones:

Para la región **us-east-1**, utilizamos la versión de Kubernetes **1.32**, dado que la versión 1.22 no era compatible.

La instancia al ser **t2.micro**, le agregamos un timeout a Prometheus y Grafana de 600 segundos. Dado al tiempo de demora en el deploy Prometheus, modificamos el **eks.tf** para incorporar una instancia de **T3.Medium**, y, además, limitamos los recursos de los pods en cuanto a cpu y memoria, **cpu = "500m"** y **memoria = "512Mi"**.