

#### Introducció

¿Qué es Promethe

> Cómo se monitori: in nuevo target?

#### Motivacione y Objetivos

Motivaciones:
Problemas con la
monitorización actu
Objetivos: ¿Qué se
pretende conseguir

### Proceso de

Ansible
API en Python

Resultados y Conclusiones

Funcionamiento de l aplicación

Trabajo futuro

Conclusiones: App v

# Grafana, Alert-Manager and Prometheus Manager

Vidal Peña, Arturo <sup>1</sup> Pérez Costoya, Fernando <sup>2</sup>

<sup>1</sup>Escuela Técnica Superior de Ingeniería Informática Universidad Politécnica de Madrid

<sup>2</sup>Departamento de Arquitectura y Tecnología de Sistemas Informáticos Escuela Técnica Superior de Ingeniería Informática Universidad Politécnica de Madrid

En colaboración con Accenture Technologies S.L.

**Junio 2022** 



### Introducción

#### ¿Qué es Prometheus?

¿Cómo se monitori un nuevo target?

#### Motivacion y Objetivos

Problemas con la monitorización actu Objetivos: ¿Qué se pretende conseguir con la app?

### Ansible API en Pytho

Resultados y

Conclusiones
Resultados:

Trabajo futuro

Conclusiones: App

1 Introducción

- ¿Qué es Prometheus?
- ¿Cómo se monitoriza un nuevo target?
- 2 Motivaciones y Objetivos
  - Motivaciones: Problemas con la monitorización actual
  - Objetivos: ¿Qué se pretende conseguir con la app?
- 3 Proceso de desarrollo
  - Ansible
  - API en Python
  - Interfaz
- 4 Resultados y Conclusiones
  - Resultados: Funcionamiento de la aplicación
  - Trabajo futuro



### ¿Qué es Prometheus?

#### Introducció

#### ¿Qué es Prometheus?

¿Cómo se monit

#### Motivacion y Objetivos

Problemas con la monitorización actu Objetivos: ¿Qué se pretende conseguir con la app?

Proceso de

Ansible API en Pyth

#### esultados y onclusiones

Funcionamiento de la aplicación Trabajo futuro Conclusiones: App ve

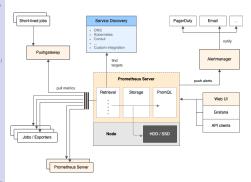


Figure 1: Arquitectura de funcionamiento de Prometheus

- ¿Qué se necesita para monitorizar con Prometheus?
- Servidor de prometheus
- Targets con etiquetas, agrupados por proyectos
- Reglas de alertado



### ¿Cómo se monitoriza un nuevo target?

¿Cómo se monitoriza un nuevo target?

# Motivaciones:

### Forma estática:

- job: node exporter static\_configs:
  - targets:
    - "localhost:9091" labels:
      - jobname: node os: linux

### Forma dinámica:

- job: node exporter file\_sd\_configs: - files: - "targets/\*.json" refresh interval: 2m

```
{
  "targets": [
    "localhost:9091"
  "labels": {
    "jobname": "node",
    "os": "linux"
```



# Motivaciones: Problemas con la monitorización actual

#### Introducción

¿Qué es Promethe

> Cómo se monitor n nuevo target?

### Motivacion y Objetivos

Motivaciones: Problemas con la monitorización actual

monitorización a Objetivos: ¿Qué

con la app?

Proceso de desarrollo

Ansible API en Pytho Interfaz

Resultados y Conclusiones

Funcionamiento de l aplicación

Trabajo futuro Conclusiones: App v

- Proceso lento y repetitivo
- Con etiquetas diferentes, hay que separar los targets
- 3 No hay automatizaciones oficiales
- 4 Alta probabilidad de introducir errores humanos
- 5 Con ficheros grandes, los errores son difíciles de detectar



## Objetivos: ¿Qué se pretende conseguir con la app?

#### Introducción

¿Qué es Promethe

> ¿Cómo se monitori un nuevo target?

### Motivacion y Objetivos

Motivaciones: Problemas con la monitorización ac

Objetivos: ¿Qué se pretende conseguir

con la app?

Proceso de

## desarrollo

API en Python Interfaz

Resultados y Conclusiones

Funcionamiento de aplicación

Trabajo futur

Conclusiones: App v

- Automatizar el proceso de añadir targets
- Aumentar la rapidez y eficacia del proceso
- Disminuir el número de errores introducidos



### ¿Cómo funciona Ansible en este proyecto?

#### Introducció

#### ¿Qué es

Cómo se monitori

### Motivacion

Motivaciones:
Problemas con la
monitorización act
Objetivos: ¿Qué se
pretende conseguir

### Proceso de desarrollo

### Ansible

API en Pythor Interfaz

Conclusiones

Funcionamiento de aplicación
Trabajo futuro

Trabajo futuro Conclusiones: App vs tradicional

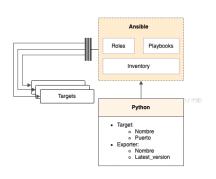


Figure 2: Diagrama ansible-node\_exporter

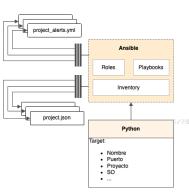


Figure 3: Diagrama ansible-ficheros\_prometheus



# API en Python

Introducció

Promethe

Cómo se monitor n nuevo target?

#### Motivacione y Objetivos

Motivaciones: Problemas con la monitorización act Objetivos: ¿Qué s pretende conseguir

con la app?

Proceso de

Ansible

API en Python Interfaz

Conclusiones

aplicación

Trabajo futuro Conclusiones: App v

- Establece el backend
- Genera endpoints para conectar con frontend
- Ejecuta ansible para instalar node exporter en los targets
- Ejecuta ansible para añadir target a Prometheus
- Crea el exporter de versiones en el target
- Añade un fichero de targets y de reglas de alertado para cada proyecto



### Interfaz

API en Python

Interfaz

- Hecha con ElectronJS
- Simple y práctica
- Permite:
  - 1 Crear proyectos
  - 2 Añadir targets
  - Buscar targets por projecto

### Resultados: Funcionamiento de la aplicación

Mejor poner fotos, el vídeo puede salirse de tiempo



# Trabajo futuro

#### Introducció

Promethe

Cómo se monitori un nuevo target?

### Motivacion y Objetivos

Motivaciones:
Problemas con la
monitorización actu
Objetivos: ¿Qué se
pretende conseguir

Proceso de

Ansible
API en Pythor
Interfaz

Resultados y Conclusiones

Funcionamiento de

#### Trabajo futuro

Conclusiones: App

- Trabajar con más exporters
- Añadir más servidores de Prometheus, Alertmanager y Grafana
- Monitorización en la nube
- Creación de dashboards en Grafana mediante la app



### Conclusiones: App vs tradicional

Introducció

Prometheus?

Cómo se monitori un nuevo target?

Motivacion y Objetivos

Motivaciones: Problemas con la monitorización ac

con la app?

Ansible
API en Pythor

Resultados y

Resultados: Funcionamiento de

Trabajo futuro

Conclusiones: App vs tradicional

### Objetivos cumplidos:

- √ La opción de introducir errores se ve reducida a la insertada por el usuario
- √ Los targets se añaden automáticamente
- √ La velocidad en el proceso aumenta considerablemente