

Zabbix Easy guide

Introducción	1
Zabbix	2
Instalación	2
Instalación de la pila LAMP	2
Obtención e instalación de los repositorios de Zabbix	2
Ubuntu-Server (Agente + Servidor + Frontend)	3
Rocky-Server (Agente)	4
Configuración Base de datos	5
Configuración de Zabbix	6
Server (Ubuntu-Server)	6
Configuración de la contraseña:	6
Configuración time para php	7
Habilitación del servicio Zabbix:	8
Agente (Rocky-Server)	8
Setup frontend de Zabbix	10
Uso de Zabbix:	12
Añadir Hosts	12
Items:	13
Triggers:	16
El Dashboard	17

Introducción

En este documento trataremos mecanismos de monitorización de servidores de alto nivel. En esta ocasión usaremos Zabbix. Un sistema diseñado para monitorizar y registrar servicios, servidores y hardware de red.

Para realizar la instalación tendremos que obtener los archivos del repositorio oficial de Zabbix, para ello tendremos que copiar enlaces y será común editar archivos, de modo que para facilitar este trabajo, recomendación es la conexión desde una máquina principal (host si estás virtualizando) y el uso de algún terminal flexible, como terminator para facilitar nuestro trabajo. Para más información al respecto puedes consultar mi [guía fácil sobre ssh](#).

Zabbix

Instalación

Respecto a la instalación de Zabbix, cabe destacar la diferencia entre:

- **Servidor:** Es el que gestiona el servicio de monitorización de Zabbix.
- **Agente:** Es un cliente del servidor que monitoriza.

De esta manera podríamos tener, por ejemplo, un computador destinado a ser un servidor Zabbix que monitorice otros servidores que ofrecen servicios a los clientes, y que son monitorizados por el servidor a través del agente. Aunque ser servidor Zabbix no es excluyente con ofrecer otros servicios como veremos más adelante.

Instalación de la pila LAMP

Para realizar la monitorización con Zabbix es necesario tener instalada la pila LAMP, para ello:

```
sudo apt install php
```

```
sudo apt install mariadb-server mariadb-client -y
```

```
sudo apt install apache2
```

Nota: recordar habilitar e iniciar todos los servicios con *sudo systemctl enable/start <servicio>*.

Obtención e instalación de los repositorios de Zabbix

Obtendremos los enlaces pertinentes a partir de la [página de descarga oficial de Zabbix](#).

Una vez en la página oficial de Zabbix, podremos escoger mediante un menú las distintas opciones que se ajusten a nuestras características. Más abajo, nos da los comandos que tendremos que ejecutar para obtener el paquete y realizar la instalación.

VERSIÓN ZABBIX	DISTRIBUCIÓN DE SO	VERSIÓN DEL SISTEMA OPERATIVO	ZABBIX COMPONENT	BASE DE DATOS	SERVIDOR WEB
6.2	Alma Linux	22.04 (Jammy)	Server, Frontend, Agent	MySQL	Apache
6.0 LTS	CentOS	20.04 (Focal)	Proxy	PostgreSQL	Nginx
5.0 LTS	Debian	18.04 (Bionic)	Agent		
4.0 LTS	Oracle Linux	16.04 (Xenial)	Agent 2		
6.3 PRE-RELEASE	Raspberry Pi OS	14.04 (Trusty)	Java Gateway		
	Red Hat Enterprise Linux				
	Rocky Linux				
	SUSE Linux Enterprise Server				
	Ubuntu				
	Ubuntu (arm64)				

[Notas de la versión 5.0](#)

Ubuntu-Server (Agente + Servidor + Frontend)

Estos son los comandos que nos ofrece Zabbix para instalar en ubuntu:

-Obtención de repositorios: *wget*

https://repo.zabbix.com/zabbix/5.0/ubuntu/pool/main/z/zabbix-release/zabbix-release_5.0-1%2Bfocal_all.deb

-Manipular paquete: *dpkg -i zabbix-release_5.0-1+focal_all.deb*

apt update

- Instalar la interfaz, el cliente y el agente: *sudo apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-agent*

```
albertopm@ubuntu:~$ sudo apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-agent
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  fontconfig-config fonts-dejavu fonts-dejavu-core fonts-dejavu-extra fping libfontconfig1 libgd3
  libjpeg-turbo8 libjpeg8 libodbc1 libpng5 libopenjp2-7 libsensors-config libsensors5
  libsnmp-base libsnmp35 libtiff5 libwebp6 libxpm4 php-bcmath php-gd php-ldap php-mbstring
  php-mysql php-xml php7.4-bcmath php7.4-gd php7.4-ldap php7.4-mbstring php7.4-mysql php7.4-xml
  snmpd
Paquetes sugeridos:
  libgd-tools libmyodbc odbc-postgresql tdsodbc unixodbc-bin lm-sensors snmp-mibs-downloader
  snmptrapd zabbix-nginx-conf
Se instalarán los siguientes paquetes NUEVOS:
  fontconfig-config fonts-dejavu fonts-dejavu-core fonts-dejavu-extra fping libfontconfig1 libgd3
  libjpeg-turbo8 libjpeg8 libodbc1 libpng5 libopenjp2-7 libsensors-config libsensors5
  libsnmp-base libsnmp35 libtiff5 libwebp6 libxpm4 php-bcmath php-gd php-ldap php-mbstring
  php-mysql php-xml php7.4-bcmath php7.4-gd php7.4-ldap php7.4-mbstring php7.4-mysql php7.4-xml
  snmpd zabbix-agent zabbix-apache-conf zabbix-frontend-php zabbix-server-mysql
0 actualizados, 37 nuevos se instalarán, 0 para eliminar y 45 no actualizados.
Se necesita descargar 12,2 MB de archivos.
Se utilizarán 51,1 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libsensors-config all 1:3.6.0-2ubuntu1.1 [6.052 B]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libsensors5 amd64 1:3.6.0-2ubuntu1.1 [27,2 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libsnmp-base all 5.8+dfsg-2ubuntu2.5 [206 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libsnmp35 amd64 5.8+dfsg-2ubuntu2.5 [980 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 snmpd amd64 5.8+dfsg-2ubuntu2.5 [56,4 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libodbc1 amd64 2.3.6-0.1build1 [189 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libopenjp2-7 amd64 2.0.27-0ubuntu2 [456 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 fping amd64 4.2-1 [32,4 kB]
Des:9 http://es.archive.ubuntu.com/ubuntu focal/main amd64 fonts-dejavu-core all 2.37-1 [1.041 kB]
Des:10 http://es.archive.ubuntu.com/ubuntu focal/main amd64 fontconfig-config all 2.13.1-2ubuntu3 [28,8 kB]
Des:11 http://es.archive.ubuntu.com/ubuntu focal/main amd64 fonts-dejavu-extra all 2.37-1 [1.953 kB]
Des:12 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 fonts-dejavu all 2.37-1 [3.130 B]
Des:13 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libfontconfig1 amd64 2.13.1-2ubuntu3 [114 kB]
Des:14 http://repo.zabbix.com/zabbix/5.0/ubuntu focal/main amd64 zabbix-server-mysql amd64 1:5.0.29-1+focal [2.881 kB]
Des:15 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libjpeg-turbo8 amd64 2.0.3-0ubuntu1.20.04.3 [118 kB]
Des:16 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libjpeg8 amd64 8c-2ubuntu8 [2.194 B]
Des:17 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libjpeg-turbo8 amd64 2.1.3-1build1 [26,7 kB]
Des:18 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libwebp6 amd64 0.6.1-2ubuntu0.20.04.1 [185 kB]
Des:19 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libtiff5 amd64 4.1.0+git191117-2ubuntu0.20.04.6 [163 kB]
Des:20 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libxpm4 amd64 1:3.5.12-1 [34,0 kB]
Des:21 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libgd3 amd64 2.2.5-5.2ubuntu2.1 [118 kB]
Des:22 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libpng5 amd64 6.9.4-1 [142 kB]
Des:23 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 php7.4-bcmath amd64 7.4.3-4ubuntu2.15 [15,1 kB]
Des:24 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 php-bcmath all 2:7.4+75 [2.004 B]
Des:25 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-gd amd64 7.4.3-4ubuntu2.15 [28,1 kB]
Des:26 http://es.archive.ubuntu.com/ubuntu focal/main amd64 php-gd all 2:7.4+75 [2.000 B]
Des:27 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-ldap amd64 7.4.3-4ubuntu2.15 [29,5 kB]
Des:28 http://es.archive.ubuntu.com/ubuntu focal/main amd64 php-ldap all 2:7.4+75 [2.004 B]
Des:29 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 php7.4-mbstring amd64 7.4.3-4ubuntu2.15 [392 kB]
Des:30 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 php-mbstring all 2:7.4+75 [2.012 B]
```

Rocky-Server (Agente)

Estos son los comandos que nos ofrece Zabbix para instalarlo en Rocky-Server:

-Obtención de repositorios: *rpm -Uvh*

https://repo.zabbix.com/zabbix/5.0/rhel/9/x86_64/zabbix-release-5.0-3.el9.noarch.rpm

dnf clean all

- Instalar el agente: *dnf install zabbix-agent*

```
albertopm@localhost:~$ sudo rpm -Uvh https://repo.zabbix.com/zabbix/5.0/rhel/9/x86_64/zabbix-release-5.0-3.el9.noarch.rpm
[albertopm@localhost ~]$ sudo rpm -Uvh https://repo.zabbix.com/zabbix/5.0/rhel/9/x86_64/zabbix-release-5.0-3.el9.noarch.rpm
[sudo] password for albertopm:
Retrieving https://repo.zabbix.com/zabbix/5.0/rhel/9/x86_64/zabbix-release-5.0-3.el9.noarch.rpm
warning: /var/tmp/rpm-tmp.c5x0qq: Header V4 RSA/SHA512 Signature, key ID 08efa7dd: NOKEY
Verifying... ##### [100%]
Preparing... ##### [100%]
Updating / installing...
 1:zabbix-release-5.0-3.el9 ##### [100%]
[albertopm@localhost ~]$ dnf clean all
0 files removed
[albertopm@localhost ~]$
```

Nota: Recordar habilitar e iniciar los servicios una vez instalados.

`sudo systemctl enable <servicio>`

`sudo systemctl start <servicio>`

Configuración Base de datos

Referencia: Continuación de la guía que ofrece la documentación oficial de Zabbix tras la instalación, es decir, sigue siendo exactamente la misma página de antes.

Primero configuraremos la base de datos con la instalación segura. Para ello ejecutaremos `sudo mysql_secure_installation` Configuraremos la seguridad básica de la base de datos según nuestro criterio.

Nota: la contraseña por defecto, nada más instalar mariadb es la cadena vacía, es decir, pulsar enter.

Ahora crearemos una base de datos para Zabbix, para ello seguiremos las instrucciones de la página oficial:

Entraremos en maria db: `sudo mariadb`

Creación de la base de datos:

```
mysql> create database zabbix character set utf8 collate utf8_bin;
```

Creación de usuario y contraseña (en password introduciremos la contraseña correspondiente):

```
mysql> create user zabbix@localhost identified by 'password';
```

Asignación de privilegios:

```
mysql> grant all privileges on zabbix.* to zabbix@localhost;
```

```
mysql> quit;
```

Y ahora importamos al servidor de Zabbix la información:

```
zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p zabbix
```

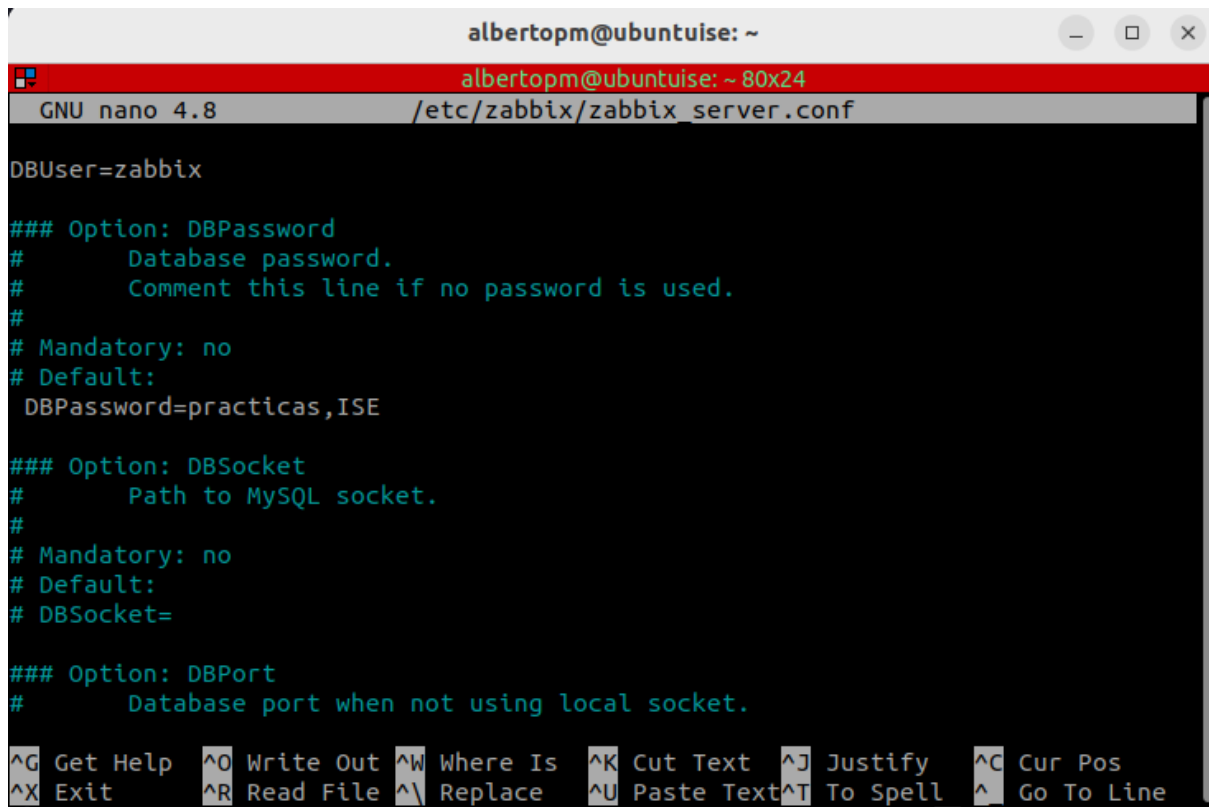
```
albertopm@ubuntu15e:~$ zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p zabbix
Enter password:
albertopm@ubuntu15e:~$
```

Configuración de Zabbix

Server (Ubuntu-Server)

Configuración de la contraseña:

Para ello, en el fichero `/etc/zabbix/zabbix_server.conf` descomentamos el apartado correspondiente a la contraseña (DBPassword) e introduciremos nuestra contraseña correspondiente.



The screenshot shows a terminal window with the title bar `albertopm@ubuntuise: ~`. The terminal is running the `nano` text editor, editing the file `/etc/zabbix/zabbix_server.conf`. The editor's status bar at the top indicates `GNU nano 4.8` and the file path. The configuration file content is as follows:

```
DBUser=zabbix

### Option: DBPassword
# Database password.
# Comment this line if no password is used.
#
# Mandatory: no
# Default:
DBPassword=practicas,ISE

### Option: DBSocket
# Path to MySQL socket.
#
# Mandatory: no
# Default:
# DBSocket=

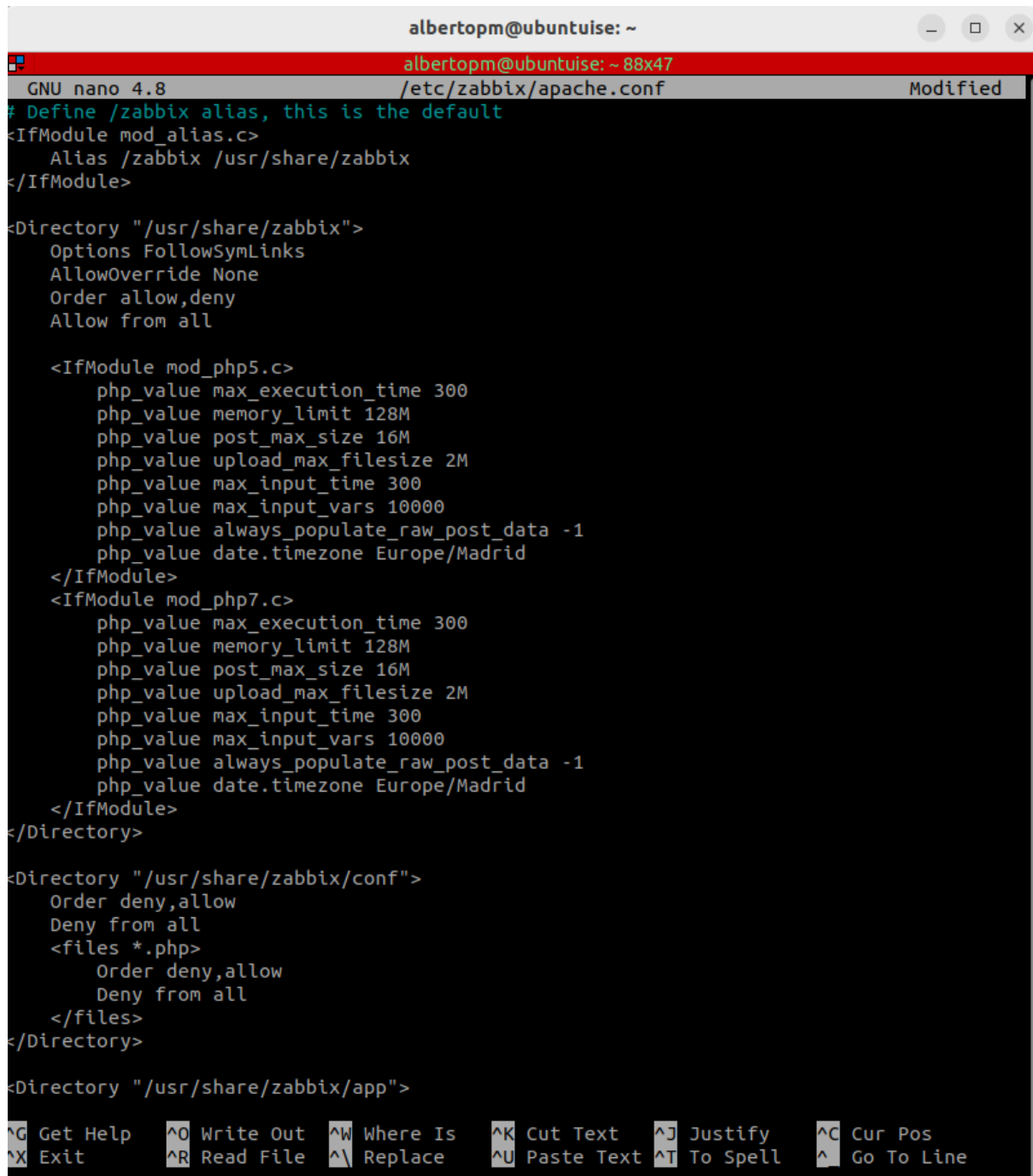
### Option: DBPort
# Database port when not using local socket.
```

At the bottom of the terminal, a help menu is visible with the following options:

<code>^G</code> Get Help	<code>^O</code> Write Out	<code>^W</code> Where Is	<code>^K</code> Cut Text	<code>^J</code> Justify	<code>^C</code> Cur Pos
<code>^X</code> Exit	<code>^R</code> Read File	<code>^_</code> Replace	<code>^U</code> Paste Text	<code>^T</code> To Spell	<code>^_</code> Go To Line

Configuración time para php

Para ello modificaremos el archivo `/etc/zabbix/apache.conf`, descomentamos las dos líneas `date.timezone`, y cambiaremos Riga por Madrid.



The screenshot shows a terminal window with the title `albertopm@ubuntuise: ~`. The terminal is running the `nano` text editor to edit the file `/etc/zabbix/apache.conf`. The editor's status bar at the top indicates `GNU nano 4.8`, the file path `/etc/zabbix/apache.conf`, and the word `Modified`. The content of the file is as follows:

```
# Define /zabbix alias, this is the default
<IfModule mod_alias.c>
    Alias /zabbix /usr/share/zabbix
</IfModule>

<Directory "/usr/share/zabbix">
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all

    <IfModule mod_php5.c>
        php_value max_execution_time 300
        php_value memory_limit 128M
        php_value post_max_size 16M
        php_value upload_max_filesize 2M
        php_value max_input_time 300
        php_value max_input_vars 10000
        php_value always_populate_raw_post_data -1
        php_value date.timezone Europe/Madrid
    </IfModule>
    <IfModule mod_php7.c>
        php_value max_execution_time 300
        php_value memory_limit 128M
        php_value post_max_size 16M
        php_value upload_max_filesize 2M
        php_value max_input_time 300
        php_value max_input_vars 10000
        php_value always_populate_raw_post_data -1
        php_value date.timezone Europe/Madrid
    </IfModule>
</Directory>

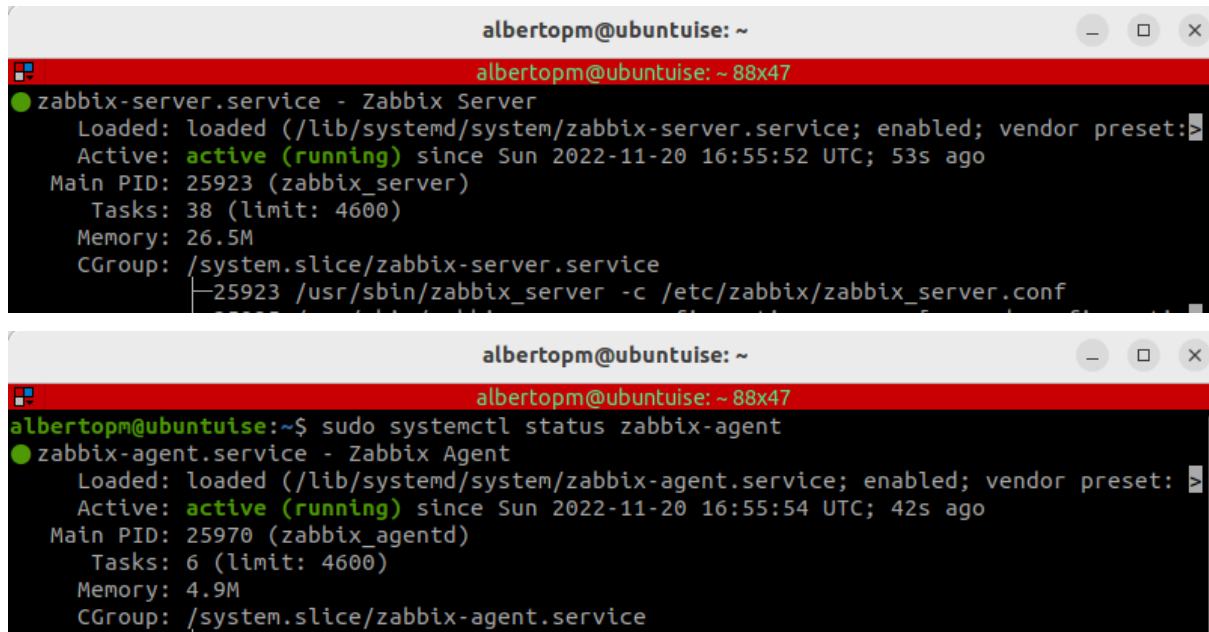
<Directory "/usr/share/zabbix/conf">
    Order deny,allow
    Deny from all
    <files *.php>
        Order deny,allow
        Deny from all
    </files>
</Directory>

<Directory "/usr/share/zabbix/app">
```

At the bottom of the terminal, the `nano` editor's help menu is visible, showing various keyboard shortcuts for navigation and editing.

Habilitación del servicio Zabbix:

```
systemctl restart zabbix-server zabbix-agent apache2  
systemctl enable zabbix-server zabbix-agent apache2
```



The first terminal screenshot shows the status of the `zabbix-server.service`. It is loaded, enabled, and active (running) since Sun 2022-11-20 16:55:52 UTC, 53s ago. The main PID is 25923 (zabbix_server). The tasks are 38 (limit: 4600) and memory usage is 26.5M. The CGroup is `/system.slice/zabbix-server.service`. The second terminal screenshot shows the status of the `zabbix-agent.service`. It is also loaded, enabled, and active (running) since Sun 2022-11-20 16:55:54 UTC, 42s ago. The main PID is 25970 (zabbix_agentd). The tasks are 6 (limit: 4600) and memory usage is 4.9M. The CGroup is `/system.slice/zabbix-agent.service`.

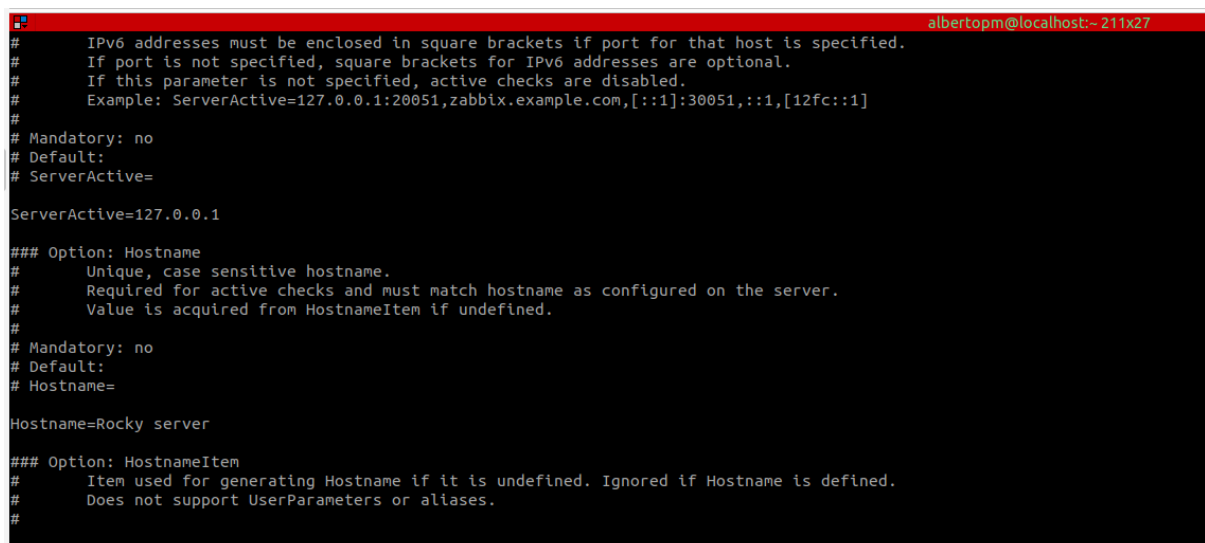
```
albertopm@ubuntuise: ~  
albertopm@ubuntuise: ~ 88x47  
● zabbix-server.service - Zabbix Server  
   Loaded: loaded (/lib/systemd/system/zabbix-server.service; enabled; vendor preset: >  
   Active: active (running) since Sun 2022-11-20 16:55:52 UTC; 53s ago  
     Main PID: 25923 (zabbix_server)  
        Tasks: 38 (limit: 4600)  
       Memory: 26.5M  
      CGroup: /system.slice/zabbix-server.service  
             └─25923 /usr/sbin/zabbix_server -c /etc/zabbix/zabbix_server.conf  
  
albertopm@ubuntuise: ~  
albertopm@ubuntuise: ~ 88x47  
albertopm@ubuntuise:~$ sudo systemctl status zabbix-agent  
● zabbix-agent.service - Zabbix Agent  
   Loaded: loaded (/lib/systemd/system/zabbix-agent.service; enabled; vendor preset: >  
   Active: active (running) since Sun 2022-11-20 16:55:54 UTC; 42s ago  
     Main PID: 25970 (zabbix_agentd)  
        Tasks: 6 (limit: 4600)  
       Memory: 4.9M  
      CGroup: /system.slice/zabbix-agent.service
```

Agente (Rocky-Server)

Para configurar el agente modificaremos el archivo `/etc/zabbix/zabbix_agentd.conf`

En el apartado `Hostname` colocaremos el nombre que añadiremos más adelante como host en la página de configuración de nuestro servidor Zabbix.

`Hostname=<Nombre Host>`



The terminal screenshot shows the configuration of `zabbix_agentd.conf`. It displays comments about IPv6 addresses and the `ServerActive` parameter. The `ServerActive` parameter is set to `127.0.0.1`. The `Hostname` parameter is set to `Rocky server`. The `HostnameItem` parameter is also shown.

```
albertopm@localhost:~ 211x27  
# IPv6 addresses must be enclosed in square brackets if port for that host is specified.  
# If port is not specified, square brackets for IPv6 addresses are optional.  
# If this parameter is not specified, active checks are disabled.  
# Example: ServerActive=127.0.0.1:20051,zabbix.example.com,[::1]:30051,::1,[12fc::1]  
#  
# Mandatory: no  
# Default:  
# ServerActive=  
  
ServerActive=127.0.0.1  
  
### Option: Hostname  
# Unique, case sensitive hostname.  
# Required for active checks and must match hostname as configured on the server.  
# Value is acquired from HostnameItem if undefined.  
#  
# Mandatory: no  
# Default:  
# Hostname=  
  
Hostname=Rocky server  
  
### Option: HostnameItem  
# Item used for generating Hostname if it is undefined. Ignored if Hostname is defined.  
# Does not support UserParameters or aliases.  
#
```


En el apartado Server colocaremos la ip del servidor Zabbix al que nos vayamos a conectar, en este caso la ip de Ubuntu (la ip de la interfaz enp0s8 en nuestro caso).

```
albertopm@localhost:/etc/zabbix
albertopm@localhost:/etc/zabbix 161x56
# Parameters are processed one by one according their appearance order.
# If no AllowKey or DenyKey rules defined, all keys are allowed.
# Unless another system.run[*] rule is specified DenyKey=system.run[*] is added by default.
#
# Mandatory: no
# Default:
# DenyKey=system.run[*]
### Option: EnableRemoteCommands - Deprecated, use AllowKey=system.run[*] or DenyKey=system.run[*] instead
# Internal alias for AllowKey/DenyKey parameters depending on value:
# 0 - DenyKey=system.run[*]
# 1 - AllowKey=system.run[*]
#
# Mandatory: no
### Option: LogRemoteCommands
# Enable logging of executed shell commands as warnings.
# 0 - disabled
# 1 - enabled
#
# Mandatory: no
# Default:
# LogRemoteCommands=0
##### Passive checks related
### Option: Server
# List of comma delimited IP addresses, optionally in CIDR notation, or DNS names of Zabbix servers and Zabbix proxies.
# Incoming connections will be accepted only from the hosts listed here.
# If IPv6 support is enabled then '127.0.0.1', '::127.0.0.1', '::ffff:127.0.0.1' are treated equally
# and '::/0' will allow any IPv4 or IPv6 address.
# '0.0.0.0/0' can be used to allow any IPv4 address.
# Example: Server=127.0.0.1,192.168.1.0/24,::1,2001:db8::/32,zabbix.example.com
#
# Mandatory: yes, if StartAgents is not explicitly set to 0
# Default:
# Server=
Server=192.168.56.110
```

Nota: La configuración del agente en Ubuntu-Server no es necesaria ya que es el localhost el que gestiona el servidor Zabbix. En caso de configurarlo, podríamos mantener la ip por defecto en Server=<ip> ya que es la correspondiente al localhost.

Importante:

la conexión Servidor agente se hará mediante el puerto 10050/tcp, por lo que será necesario añadirlo al firewall, especialmente en Rocky ya que este sí que tiene activado el firewall por defecto, en Ubuntu no es necesario ya que ufw está desactivado, si estuviese activado también habría que abrir el puerto.

```
sudo firewall-cmd --add-port=10050/tcp --permanent
```

Además es necesario tener fingerprint ssh entre servidor y agente (de manera bidireccional). Puedes consultar cómo hacerlo en mi [guía fácil sobre ssh](#).

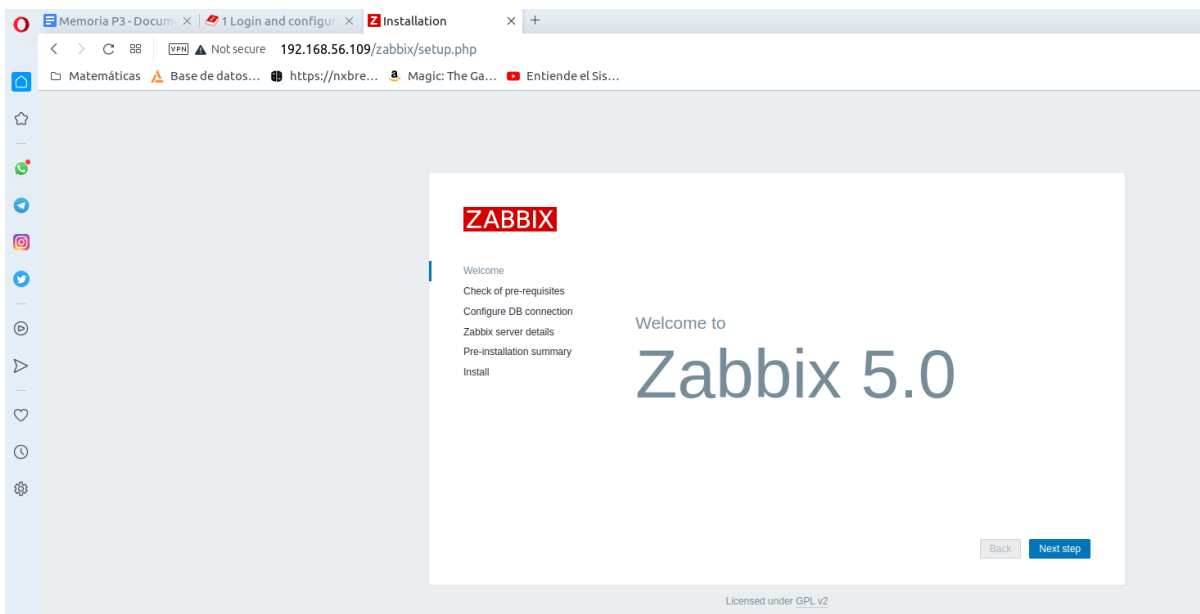
También es importante abrir los puertos que vayan a usarse en los ítems, por ejemplo, si Rocky va a usar HTTP y vamos a monitorizarlo, tenemos que abrir el puerto 80, correspondiente al servicio.

```
sudo firewall-cmd --add-port=80/tcp --permanent  
sudo firewall-cmd --reload
```

Setup frontend de Zabbix

Continuaremos con el uso de Zabbix, para el que usaremos la [documentación oficial](#).

Entraremos en la página de configuración de Zabbix mediante nuestra ip en la interfaz



En caso de que salte incompatibilidad respecto a la zona horaria de Zabbix y php, podemos configurar la de php en el archivo `/etc/php/<version>/apache2/php.ini` Buscando el apartado `[Date]` y descontando el `timezone` y colocando el que nos corresponda.

```
[Date]  
; Defines the default timezone used by the date functions  
; http://php.net/date.timezone  
date.timezone = "Europe/Madrid"
```

Nota: recordar reiniciar el servicio `apache2`.

Ahora configuraremos el acceso a la base de datos rellenando los campos pertinentes.

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

Pre-installation summary

Install

Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database. Press "Next step" button when done.

Database type

MySQL

Database host

localhost

Database port

0

0 - use default port

Database name

zabbix

User

zabbix

Password

Database TLS encryption

Connection will not be encrypted because it uses a socket file (on Unix) or shared memory (Windows).

Back

Next step

Licensed under GPL v2

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

Pre-installation summary

Install

Zabbix server details

Please enter the host name or host IP address and port number of the Zabbix server, as well as the name of the installation (optional).

Host

localhost

Port

10051

Name

Bench

Back

Next step

Licensed under GPL v2

Y obtendremos la pestaña de instalación del frontend

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

Pre-installation summary

Install

Install

Congratulations! You have successfully installed Zabbix frontend.

Configuration file "conf/zabbix.conf.php" created.

Back

Finish

Licensed under GPL v2

Ahora iniciaremos sesión como superusuario Zabbix, para ello:

Usuario: Admin

Contraseña: Zabbix

Uso de Zabbix:

Para enseñar cómo funciona Zabbix, configuraremos la monitorización de los servicios HTTP y SSH a modo de ejemplo.

Añadir Hosts

Cada host es cada servidor que estamos monitorizando, en el caso del servidor que gestiona el frontend de Zabbix, no es necesario añadirlo, ya que este ya está configurado por defecto.

Para añadir un nuevo host:

Configuration → hosts → create host

The screenshot shows the Zabbix web interface for creating a new host. The left sidebar contains the navigation menu with 'Configuration' expanded and 'Hosts' selected. The main content area is titled 'Hosts' and includes tabs for Host, Templates, IPMI, Tags, Macros, Inventory, and Encryption. The form fields are as follows:

- Host name:** Rocky server
- Visible name:** (empty field)
- Groups:** Linux servers (selected from a dropdown with a search box)
- Interfaces:** A table with columns: Type, IP address, DNS name, Connect to, Port, and Default.

Type	IP address	DNS name	Connect to	Port	Default
Agent	192.168.56.103		IP	10050	<input checked="" type="radio"/> Remove
- Description:** (empty text area)
- Monitored by proxy:** (no proxy) (dropdown menu)
- Enabled:** ☒
- Buttons:** Add, Cancel

En esta pantalla haremos coincidir los campos Host name e Interfaces con el hostname que colocamos en el documento de configuración del agente que estamos añadiendo, tal y como vimos en el apartado Configuración de Zabbix → Agente (Rocky-Server).

Una vez hecho esto, podremos observar una lista de los distintos hosts, mostrando su disponibilidad (cuando esté disponible se pondrá ZBX en verde).

<input type="checkbox"/>	Name	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Proxy	Templates	Status ▾	Availability	Agent encryption	Info	Tags	
<input type="checkbox"/>	Rocky server	Applications	Items 1	Triggers 1	Graphs	Discovery	Web	192.168.56.103:10050			Enabled	ZBX	SNMP	JMX	IPMI	NONE
<input type="checkbox"/>	Zabbix server	Applications 17	Items 126	Triggers 67	Graphs 25	Discovery 3	Web	127.0.0.1: 10050		Template App Zabbix Server, Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent, Template Module Linux memory by Zabbix agent, Template Module Linux network interfaces by Zabbix agent, Template Module Zabbix agent)	Enabled	ZBX	SNMP	JMX	IPMI	NONE

Displaying 2 of 2 found

Nota: si sale en rojo es porque no puede conectarse correctamente al host. Esto puede ser debido a:

- No se ha configurado el fingerprint ssh correctamente.
- No se ha abierto el puerto para intercambio de datos (10050/tcp por defecto).
- Lo escrito en el archivo de configuración de agente y en la página de host no coincide.

Items:

Sirven para monitorizar distintas características de servicios, recursos hardware, etc. Van asociados a un host.

Para crearlos:

Configuration → hosts → <host en el que queremos añadir un item> → items → create item.

- Nombre: Sirve para identificar el ítem correcto.
- Key: Es el código que asigna el ítem a lo que queremos monitorizar. En la [documentación oficial de Zabbix](#) podemos ver una lista completa de las keys disponibles.
- Type of information: colocaremos la salida de dato que tendrá el ítem; la documentación lo especifica.

De este modo, la creación de ítems es mecánica. Revisando en la documentación la key de lo que queremos monitorizar, y añadiendo sus salidas.

Ejemplo HTTP y SSH para el host Ubuntu, el procedimiento para hacerlo en Rocky sería exactamente el mismo solo que entrando en el host correspondiente.

Usaremos las key `net.tcp.service.perf[service,<ip>,<port>]`, la cual puede usarse tanto en los servicios ssh como http, con sus respectivos puertos. Con esta key podremos medir la velocidad de los servicios.

Ejemplos del SSH en Rocky y el HTTP en Ubuntu:

ZABBIX << >> Bench

All hosts / Zabbix server Enabled **ZBX** SNMP JMX IPMI Applications 13 Items 100 Triggers 50 Graphs 21 Discovery rules 3 Web scenarios

Item Preprocessing

* Name **HTTP-Ubuntu**

Type Zabbix agent

* Key **net.tcp.service.perf[http,,80]** Select

* Host interface 127.0.0.1 : 10050

Type of information Numeric (float)

Units

* Update interval 1m

Custom intervals

Type	Interval	Period	Action
Flexible	Scheduling	50s	1-7,00:00-24:00 Remove

Add

* History storage period Do not keep history Storage period 90d

* Trend storage period Do not keep trends Storage period 365d

Show value As is show value mappings

New application

Applications

- None-
- CPU
- Disk sda
- Filesystems
- General
- Inventory
- Memory
- Monitoring agent
- Network interfaces
- Security

Populates host inventory field -None-

Description

Enabled ☒

Update Clone Execute now Test Clear history and trends Delete Cancel

ZABBIX << >> Bench

All hosts / Zabbix server Enabled **ZBX** SNMP JMX IPMI Applications 13 Items 100 Triggers 50 Graphs 21 Discovery rules 3 Web scenarios

Item Preprocessing

* Name **ssh-ubuntu**

Type Zabbix agent

* Key **net.tcp.service.perf[ssh,,22]** Select

* Host interface 127.0.0.1 : 10050

Type of information Numeric (unsigned)

Units

* Update interval 1m

Custom intervals

Type	Interval	Period	Action
Flexible	Scheduling	50s	1-7,00:00-24:00 Remove

Add

* History storage period Do not keep history Storage period 90d

* Trend storage period Do not keep trends Storage period 365d

Show value As is show value mappings

New application

Applications

- None-
- CPU
- Disk sda
- Filesystems
- General
- Inventory
- Memory
- Monitoring agent
- Network interfaces
- Security

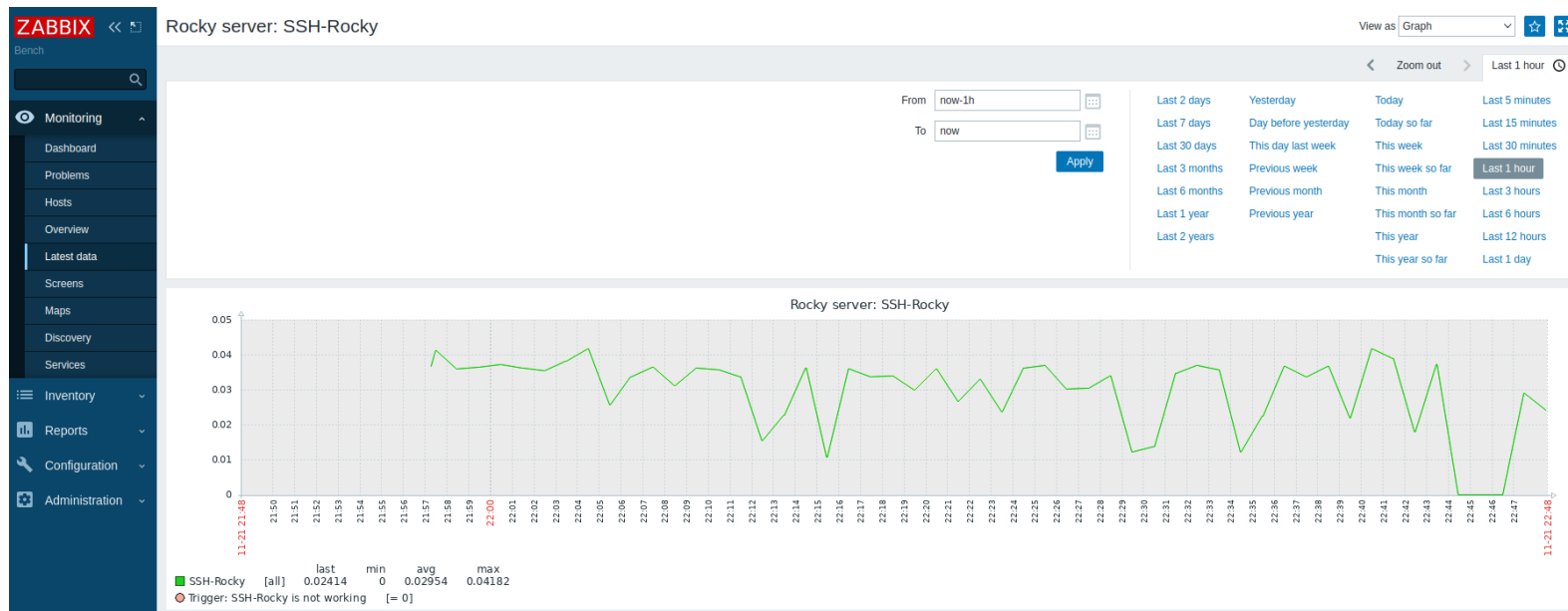
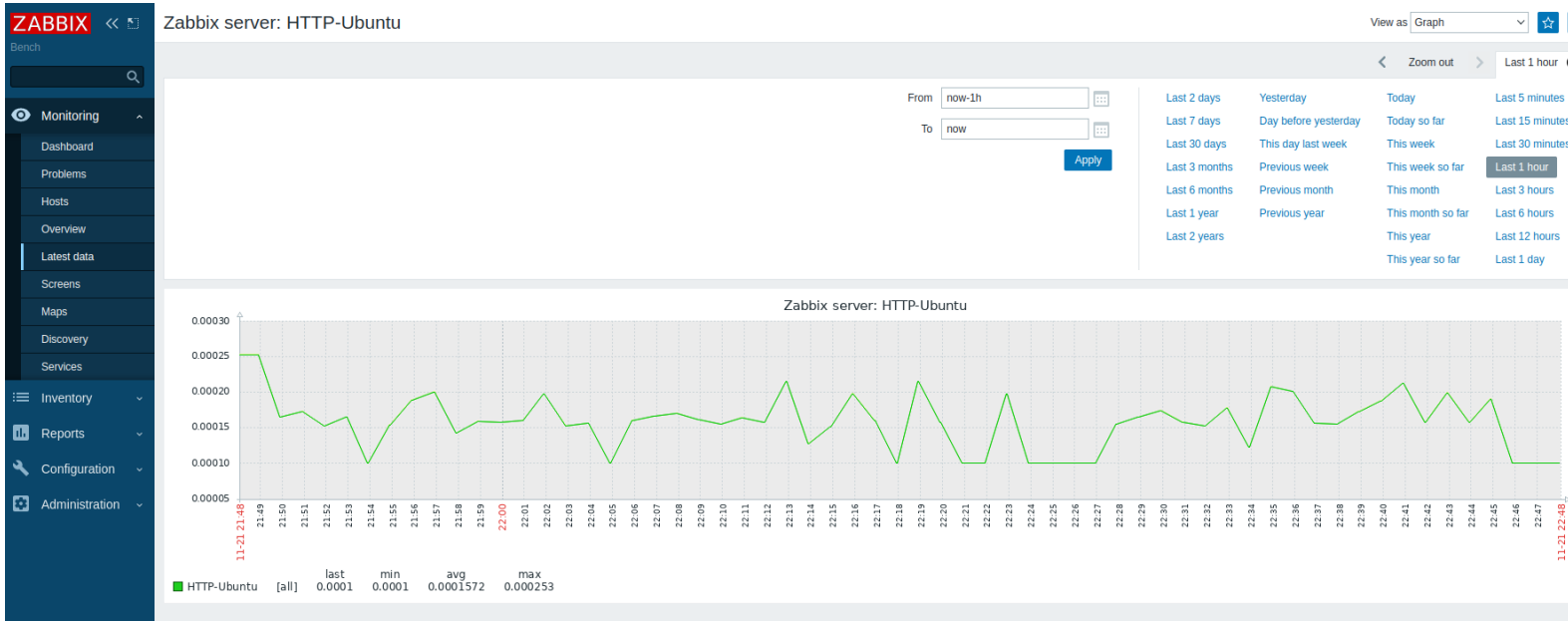
Populates host inventory field -None-

Description

Enabled ☒

Ingeniería informática (2022-2023)
Ingeniería de Servidores
Plaza Montes Alberto

De esta forma, podemos monitorizar esos tiempos. Si entramos en Monitoring → Latest Data, y ahora seleccionamos el ítem que queremos monitorizar, obtendremos un gráfico:



Triggers:

Son avisos dados ciertos parámetros configurados.

Para crearlos:

Configuration → hosts → <host en el que queremos añadir un trigger> → Triggers → Create Trigger

- Name: Nombre para identificarlo.
- Severity: Gravedad del asunto.
- Expression: Pulsamos add, y en ítem, seleccionamos el que queremos gestionar como trigger.
- Resto de parámetros: cuando se activa el trigger

Ejemplo, creamos un trigger para el host Rocky Server, con el ítem SSH-Rocky, el cual hemos creado anteriormente, en el que si obtenemos un 0 (caída de servicio) obtendremos un trigger en la página principal.

ZABBIX << Bench

All hosts / Rocky server Enabled ZBX SNMP JMX IPMI Applications Items 1 Triggers Graphs Discovery rules Web scenarios

Trigger Tags Dependencies

Name SSH-Rocky is not working

Operational data

Severity Not classified Information Warning Average High Disaster

Expression {Rocky server:net.tcp.service.perf[ssh,,22].last()}=0 Add

Expression constructor

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Allow manual close ☐

URL

Description

Enabled ☒

Add Cancel

De modo que si ahora, forzamos un error, tirando a propósito el servicio ssh en Rocky,

`sudo systemctl stop sshd`

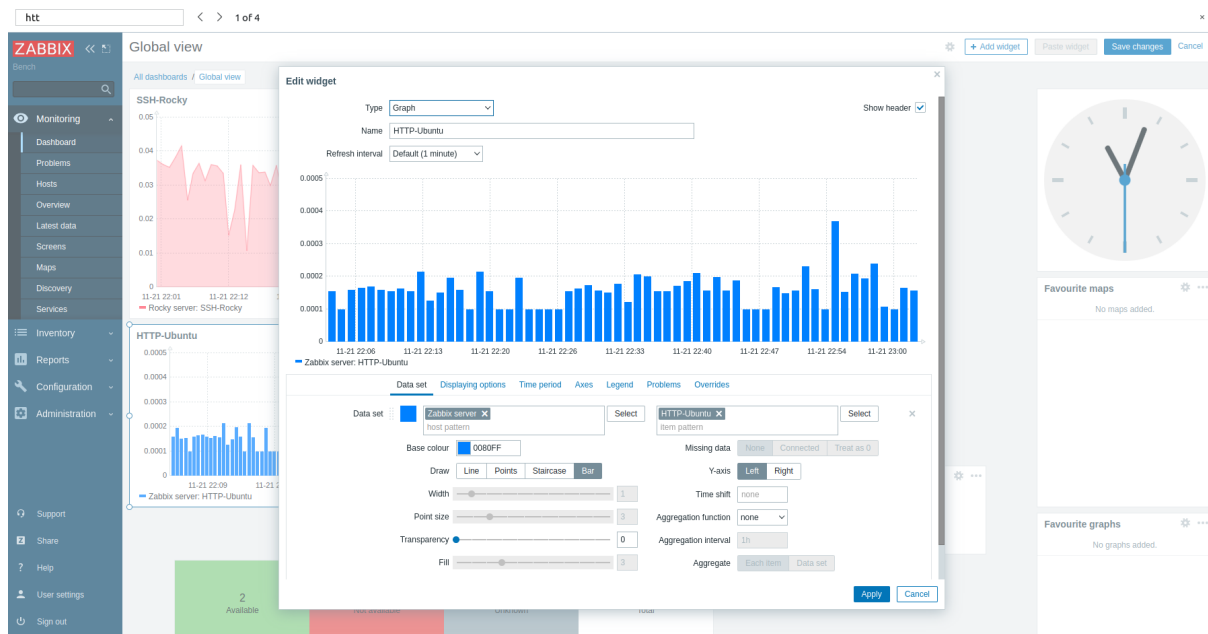
obtenemos un trigger en la página principal de monitorización:



El Dashboard

Zabbix tiene múltiples parámetros configurables, ya solo quedaría explorar [el manual](#) para investigar las distintas opciones que tiene este servicio de monitorización. Configurar el Dashboard de acuerdo a las principales necesidades de monitorización de nuestro servidor configurando los distintos widgets.

Ejemplo de configuración de Widget:



Ejemplo de posible panel dashboard:

