



UNIVERSIDAD DE GRANADA

3ºA - A2

GRADO EN INGENIERÍA INFORMÁTICA

Ingeniería de Servidores: Práctica 3.

Autor:

Antonio David Villegas Yeguas

Asignatura:

Ingeniería de Servidores

18 de noviembre de 2019

<https://github.com/advy99/ISE/>

Esta obra está bajo una licencia [Creative Commons](#) “Reconocimiento-NoCommercial-CompartirIgual 4.0 Internacional”.



Índice

1. Instalación de Zabbix	2
1.1. Instalación en Ubuntu	2
1.2. Creación de la BD	4
1.3. Configuración de Zabbix Server	6
1.4. Configuración del front-end	6
1.5. Configuración de Zabbix Agent en Ubuntu	6
1.6. Configuración de Zabbix a través del navegador	7
1.6.1. Configuración inicial:	7
1.6.2. Crear un usuario:	8
1.6.3. Gestionar permisos:	8
2. Configurar Zabbix Agent en CentOS	9
2.1. Instalación de zabbix-agent	9
2.2. Configuración del agente	9
3. Monitorizar máquinas	11
3.1. Añadir máquinas a monitorizar:	11
3.2. Añadir items a una máquina:	11
3.3. Revisar los datos	12
3.4. Triggers	14
4. Conclusiones	16
Bibliografía	17

1. Instalación de Zabbix

Tras configurar el nuevo disco en el RAID1 y monitorizar los discos disponibles en nuestro RAID1 con systemd, llegamos al punto de la instalación de Zabbix.

Para esto, primero es conveniente leer el manual de Zabbix[1] dado en el guión de prácticas de esta práctica.

Tras leer la introducción del manual, en la que nos explican la estructura del manual y una breve descripción de cada apartado así como los distintos softwares que tiene Zabbix (cliente, servidor, proxy, etc) , pasamos a la instalación.

En nuestro caso, instalaremos Zabbix Server en Ubuntu y Zabbix Agent en Ubuntu y CentOS, así que nos centraremos en estos SO, cabe decir el manual de Zabbix contempla la instalación del software para Ubuntu, RHEL/CentOS y cualquier distribución a través del código fuente, ya que Zabbix es libre.

1.1. Instalación en Ubuntu

Lo primero que encontramos en el manual de instalación son las formas de obtener Zabbix, que son las siguientes:

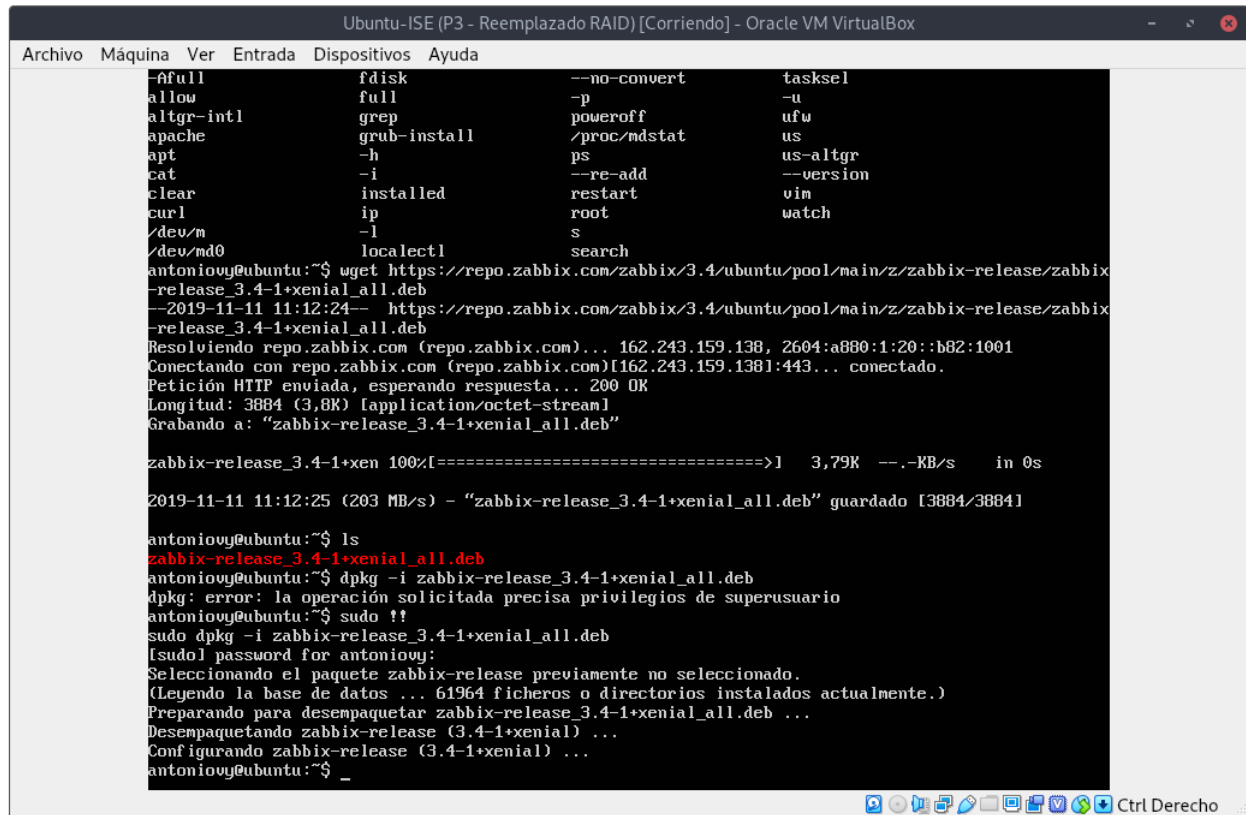
1. Desde el código fuente
2. Desde paquetes precompilados

En nuestro caso, utilizaremos la instalación desde paquetes precompilados para que APT se encargue de resolver los problemas de dependencias y requisitos al instalar Zabbix.

Lo instalamos de la siguiente forma:

```
1 # wget https://repo.zabbix.com/zabbix/3.4/ubuntu/pool/main/z/zabbix-release/\n2 zabbix-release_3.4-1+xenial_all.deb\n3 # dpkg -i zabbix-release_3.4-1+xenial_all.deb\n4 # apt update
```

Vemos como se ha instalado correctamente:



```
Ubuntu-ISE (P3 - Reemplazado RAID) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

--full      fdisk      --no-convert  tasksetl
allow       full       -p            -u
altgr-intl  grep       poweroff      ufw
apache      grub-install /proc/mdstat  us
apt         -h         ps            us-altgr
cat         -i         --re-add      --version
clear       installed  restart       vim
curl        ip         root          watch
/dev/m      -l         s
/dev/md0    localectl search

antoniovy@ubuntu:~$ wget https://repo.zabbix.com/zabbix/3.4/ubuntu/pool/main/z/zabbix-release/zabbix
-release_3.4-1+xenial_all.deb
--2019-11-11 11:12:24-- https://repo.zabbix.com/zabbix/3.4/ubuntu/pool/main/z/zabbix-release/zabbix
-release_3.4-1+xenial_all.deb
Resolviendo repo.zabbix.com (repo.zabbix.com)... 162.243.159.138, 2604:a880:1:20::b82:1001
Conectando con repo.zabbix.com (repo.zabbix.com)[162.243.159.138]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 3884 (3,8K) [application/octet-stream]
Grabando a: "zabbix-release_3.4-1+xenial_all.deb"

zabbix-release_3.4-1+xen 100%[=====>] 3,79K --KB/s in 0s

2019-11-11 11:12:25 (203 MB/s) - "zabbix-release_3.4-1+xenial_all.deb" guardado [3884/3884]

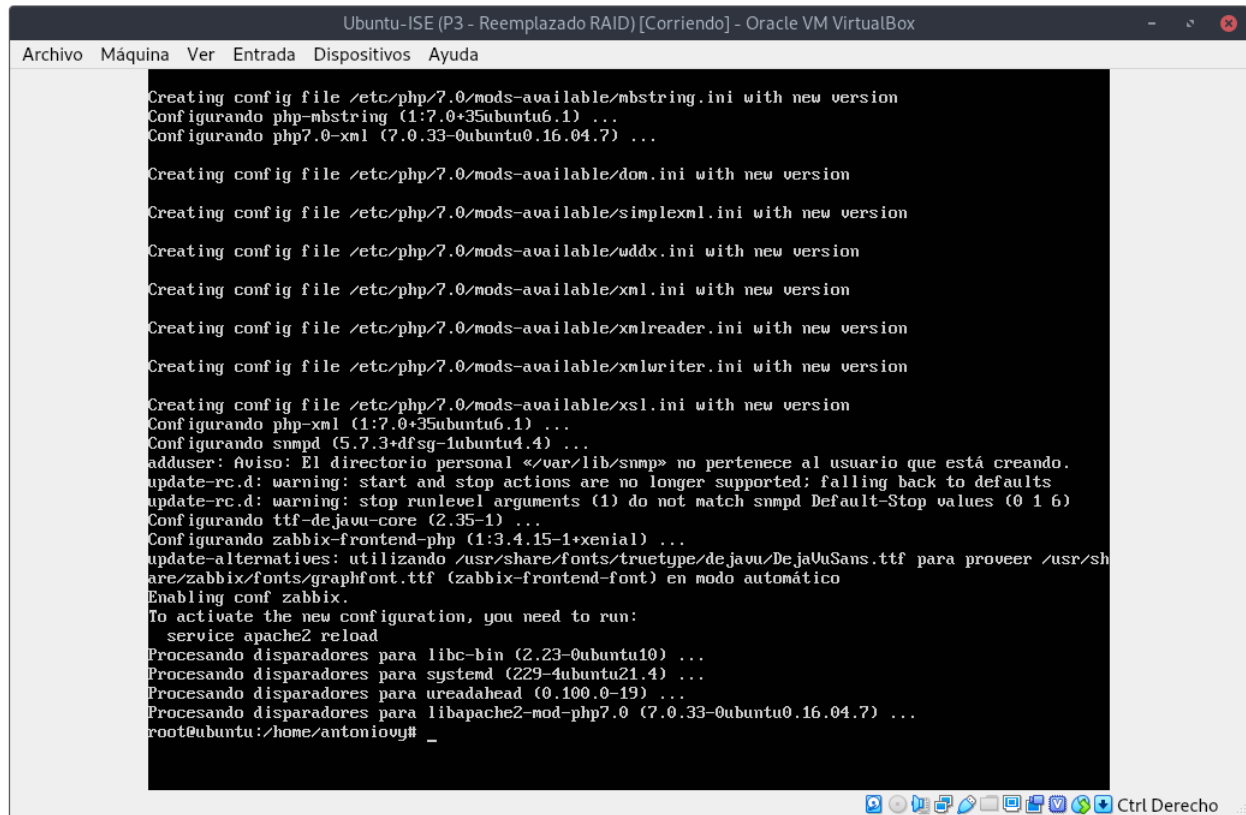
antoniovy@ubuntu:~$ ls
zabbix-release_3.4-1+xenial_all.deb
antoniovy@ubuntu:~$ dpkg -i zabbix-release_3.4-1+xenial_all.deb
dpkg: error: la operación solicitada precisa privilegios de superusuario
antoniovy@ubuntu:~$ sudo !!
sudo dpkg -i zabbix-release_3.4-1+xenial_all.deb
[sudo] password for antoniovy:
Seleccionando el paquete zabbix-release previamente no seleccionado.
(Leyendo la base de datos ... 61964 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar zabbix-release_3.4-1+xenial_all.deb ...
Desempaquetando zabbix-release (3.4-1+xenial) ...
Configurando zabbix-release (3.4-1+xenial) ...
antoniovy@ubuntu:~$ _
```

Aun así esto solo ha instalado los repositorios de Zabbix en Ubuntu, ahora pasamos a instalar el servidor de Zabbix con MySQL, el frontend de Zabbix Server y el cliente de Zabbix:

- 1 # apt install zabbix-server-mysql
- 2 # apt install zabbix-frontend-php
- 3 # apt install zabbix-agent

Y con esto, tenemos el servidor y cliente de Zabbix instalado, pasamos a la configuración.

Por último, comentar un detalle bastante importante, el manual de Zabbix nos indica que para activar y habilitar Zabbix hay que utilizar el comando `service` y el sistema de servicio `update-rc`, sin embargo, como nos avisa la instalación, este software ya no está en uso, ya que ha sido sustituido por SystemD, por este motivo yo usaré `systemctl` para gestionar los distintos procesos de Zabbix.



```
Creating config file /etc/php/7.0/mods-available/mbstring.ini with new version
Configurando php-mbstring (1:7.0+35ubuntu6.1) ...
Configurando php7.0-xm1 (7.0.33-0ubuntu0.16.04.7) ...

Creating config file /etc/php/7.0/mods-available/dom.ini with new version

Creating config file /etc/php/7.0/mods-available/simplexml.ini with new version

Creating config file /etc/php/7.0/mods-available/xdm.ini with new version

Creating config file /etc/php/7.0/mods-available/xml.ini with new version

Creating config file /etc/php/7.0/mods-available/xmlreader.ini with new version

Creating config file /etc/php/7.0/mods-available/xmlwriter.ini with new version

Creating config file /etc/php/7.0/mods-available/xsl.ini with new version
Configurando php-xml (1:7.0+35ubuntu6.1) ...
Configurando snmpd (5.7.3+dfsg-1ubuntu4.4) ...
adduser: Aviso: El directorio personal «/var/lib/snmp» no pertenece al usuario que está creando.
update-rc.d: warning: start and stop actions are no longer supported; falling back to defaults
update-rc.d: warning: stop runlevel arguments (1) do not match snmpd Default-Stop values (0 1 6)
Configurando ttf-dejavu-core (2.35-1) ...
Configurando zabbix-frontend-php (1:3.4.15-1+xenial) ...
update-alternatives: utilizando /usr/share/fonts/truetype/dejavu/DejaVuSans.ttf para proveer /usr/sh
are/zabbix/fonts/graphfont.ttf (zabbix-frontend-font) en modo automático
Enabling conf zabbix.
To activate the new configuration, you need to run:
service apache2 reload
Procesando disparadores para libc-bin (2.23-0ubuntu10) ...
Procesando disparadores para systemd (229-4ubuntu21.4) ...
Procesando disparadores para ureadahead (0.100.0-19) ...
Procesando disparadores para libapache2-mod-php7.0 (7.0.33-0ubuntu0.16.04.7) ...
root@ubuntu:/home/antoniovg# _
```

1.2. Creación de la BD

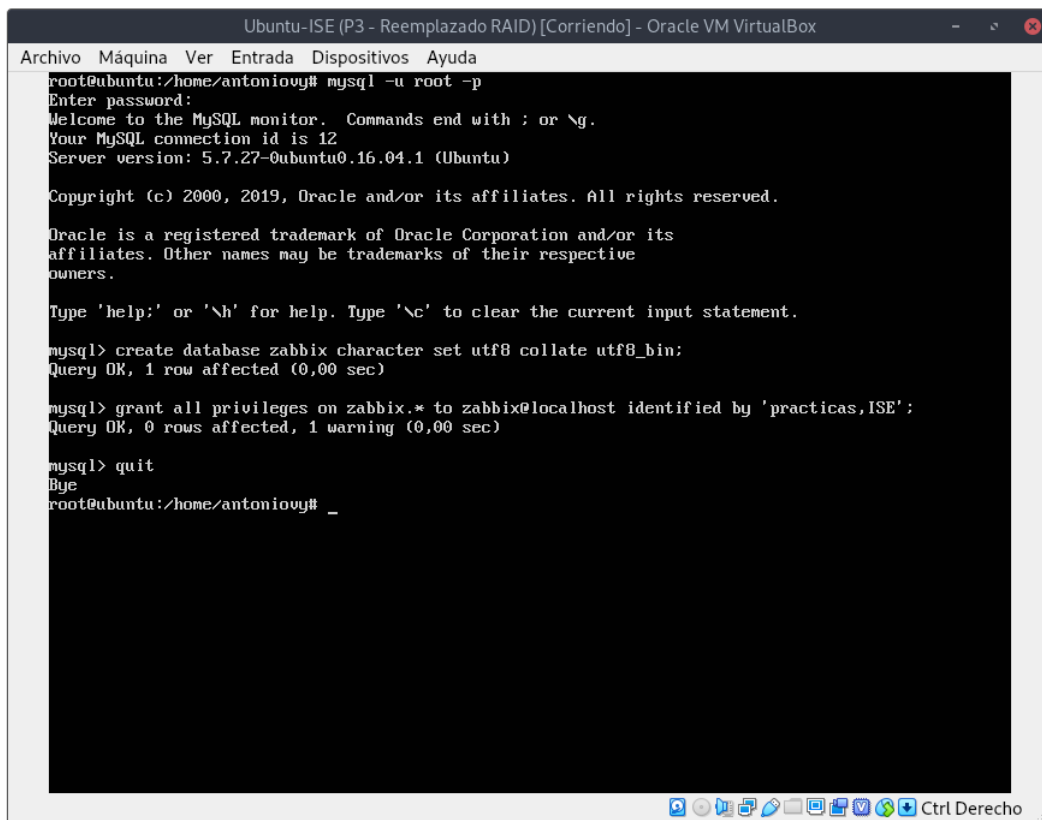
Una vez instalado Zabbix, es necesario crear una base de datos, para realizar este paso consultaremos el apéndice del manual de Zabbix, en concreto la sección en la que nos enseñan a crear una BD[2].

El primer paso, como indican, es localizar los archivos `schema.sql`, `images.sql` y `data.sql`, que al instalarlo por los paquetes, tenemos esos archivos en:

`/usr/share/doc/zabbix-server-mysql/create.sql.gz`, y solo será necesario realizar los primeros pasos de la creación de la BD, ya que los siguientes están en la instalación específica de Ubuntu.

```
1 shell> mysql -u root -p
2 mysql> create database zabbix character set utf8 collate utf8_bin;
3 mysql> grant all privileges on zabbix.* to zabbix@localhost identified by
   ↪ '<password>';
4 mysql> quit;
```

Donde para nosotros, <password> será practicas, ISE



The screenshot shows a terminal window titled 'Ubuntu-ISE (P3 - Reemplazado RAID) [Corriendo] - Oracle VM VirtualBox'. The terminal displays the following commands and output:

```
root@ubuntu:/home/antoniovy# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.7.27-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database zabbix character set utf8 collate utf8_bin;
Query OK, 1 row affected (0,00 sec)

mysql> grant all privileges on zabbix.* to zabbix@localhost identified by 'practicas,ISE';
Query OK, 0 rows affected, 1 warning (0,00 sec)

mysql> quit
Bye
root@ubuntu:/home/antoniovy# _
```

Y pasamos a exportar el esquema inicial de la base de datos:

- 1 `# zcat /usr/share/doc/zabbix-server-mysql/create.sql.gz | mysql -u zabbix -p zabbix`

Podemos comprobar que se ha realizado correctamente entrando en MySQL y comprobando las tablas de la BD zabbix:

- 1 `shell> mysql -u zabbix -p zabbix`
- 2 `mysql> show tables;`
- 3 `mysql> quit;`

1.3. Configuración de Zabbix Server

Ahora pasamos a la configuración del servidor de Zabbix, para esto pasamos a editar el archivo de configuración, ubicado en `/etc/zabbix/zabbix_server.conf`

```
1 # vi /etc/zabbix/zabbix_server.conf
2 DBHost=localhost
3 DBName=zabbix
4 DBUser=zabbix
5 DBPassword=practicass,ISE
```

Es necesario establecer `DBHost` a `localhost`, para que use MySQL y no PostgreSQL y establecer `DBPassword` a `practicass,ISE`, mientras que `DBName` y `DBUser` ya vienen con esos valores por defecto.

Por último, activamos y habilitamos el servicio de zabbix:

```
1 # systemctl start zabbix-server
2 # systemctl enable zabbix-server
```

Un detalle importante, es que aunque actualmente estemos en Ubuntu, pero si el servidor lo instalamos en CentOS debemos activar las distintas flags de SELinux para permitir que un servicio http se conecte a las bases de datos.

1.4. Configuración del front-end

La configuración del cliente de Zabbix la podemos encontrar en:

`/etc/apache2/conf-enabled/zabbix.conf`

La configuración ya está activada, sin embargo hay que quitar el comentario de la línea con el valor para la zona horaria.

Por último, reiniciamos apache2:

```
1 # systemctl restart apache2
```

1.5. Configuración de Zabbix Agent en Ubuntu

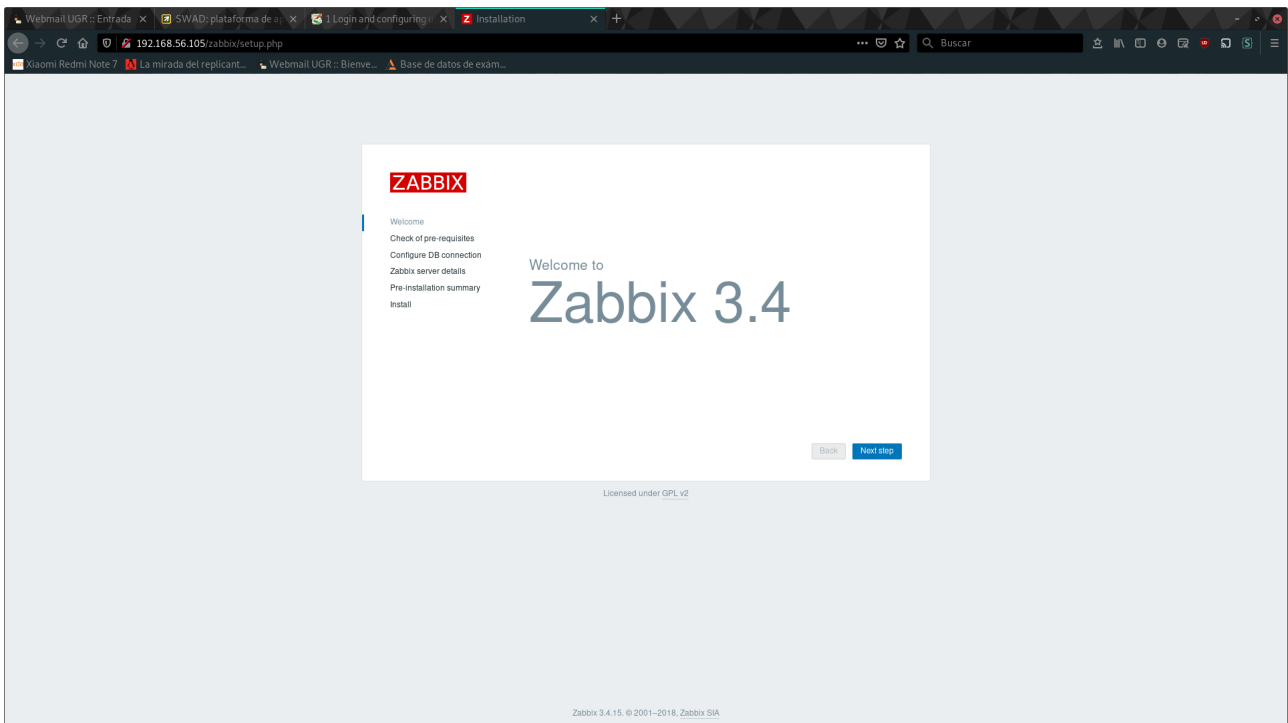
Ubuntu al tener instalado el servidor en el mismo sistema operativo no necesita configurar zabbix-agent, con iniciar el proceso basta, además de añadir en ufw el puerto usado por Zabbix, el 10050. Esto último no es obligatorio ya que se va a conectar con la propia máquina, pero en caso de tener el servidor Zabbix instalado en otra máquina si es necesario.

```
1 # systemctl start zabbix-agent
2 # systemctl enable zabbix-agent
3 # ufw allow 10050
```

1.6. Configuración de Zabbix a través del navegador

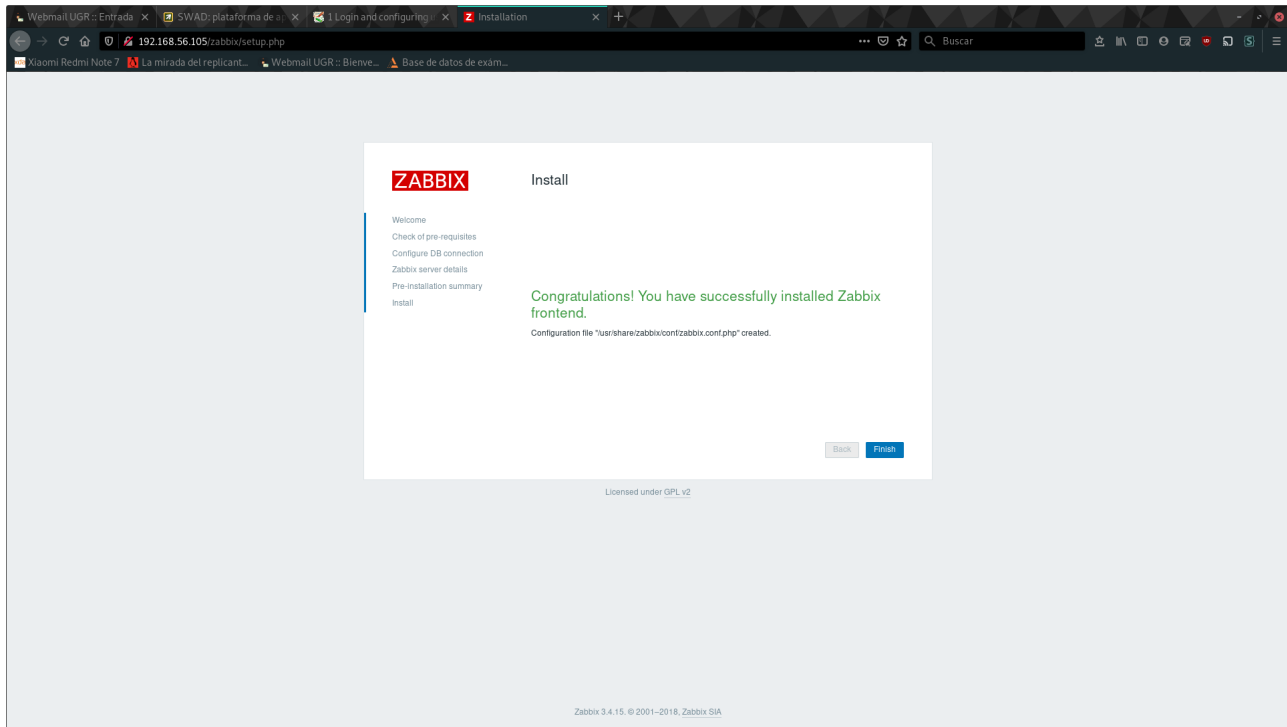
Con todo esto, ya tenemos Zabbix disponible en el navegador y pasamos a su configuración:

1.6.1. Configuración inicial:



En la siguiente pestaña veremos como todos los paquetes necesarios son correctos, y en la siguiente nos pedirá conectarnos a la base de datos creada para Zabbix, cuya configuración es la por defecto, añadiendo la clave usada, es decir, practicas, ISE, manteniendo el host en localhost, y el nombre de la BD y el nombre de usuario en zabbix.

Ahora pasamos a poner el host e IP del servidor de Zabbix, los cuales dejaremos por defecto, además de el nombre de la instalación, que en mi caso pondré Zabbix-Server19, y con esto acabamos la configuración inicial:



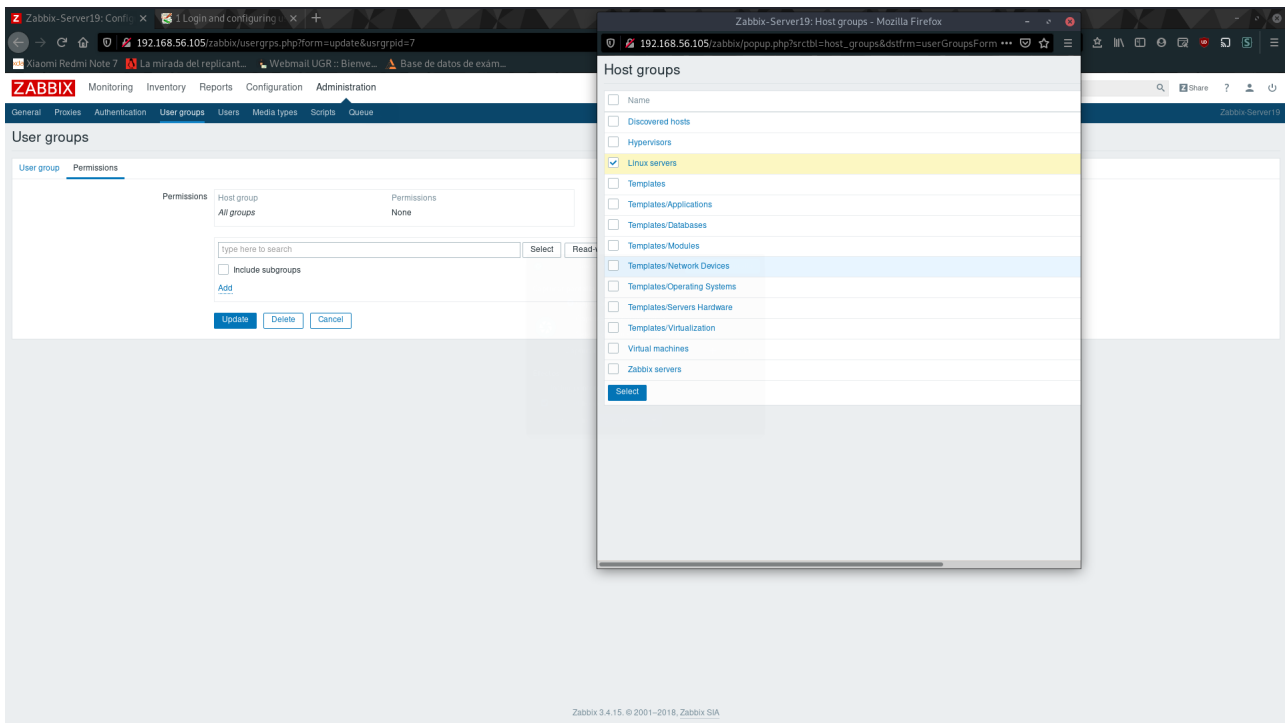
1.6.2. Crear un usuario:

Con esto podremos acceder a Zabbix a través del navegador usando el usuario Admin y contraseña zabbix. Lo primero que haremos será cambiar este usuario y contraseña como nos explican en su manual de configuración rápida[3], en concreto en el menú Administración ->Usuarios, desde este menú podremos gestionar todos los usuarios de Zabbix. Vemos como por defecto existen dos usuarios, el administrador y el invitado.

1.6.3. Gestionar permisos:

En mi caso, crearé un usuario con permisos de lectura a Linux servers para que sea capaz de monitorizar el sistema.

En el caso de Zabbix, los permisos se gestionan por grupos, es decir, nosotros no asignaremos los permisos a un usuario, si no a un grupo, así que para que un usuario tenga ciertos permisos (asociados a un grupo) este usuario debe pertenecer al grupo que tenga esos permisos.



2. Configurar Zabbix Agent en CentOS

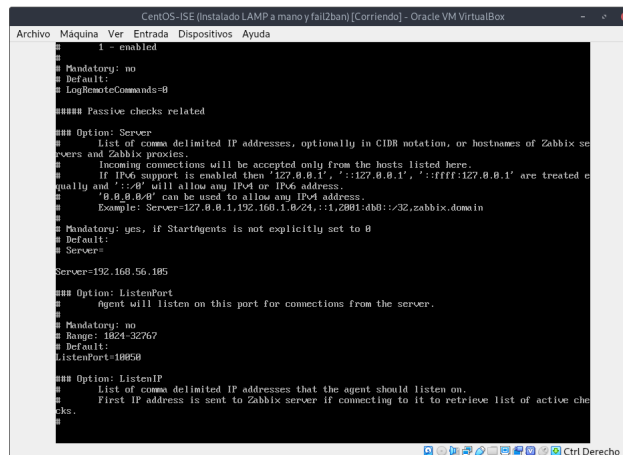
2.1. Instalación de zabbix-agent

Para monitorizar CentOS lo primero que tenemos que hacer es instalar zabbix-agent y permitir el puerto que usaremos para conectarnos al servidor:

```
1 # rpm -ivh
   ↪ https://repo.zabbix.com/zabbix/3.4/rhel/7/x86_64/zabbix-release-3.4-2.el7.noarch.rpm
2 # yum install zabbix-agent
3 # firewall-cmd --add-port=10050/tcp
4 # firewall-cmd --add-port=10050/tcp --permanent
```

2.2. Configuración del agente

Una vez instalado, modificamos el archivo de configuración `/etc/zabbix/zabbix_agentd.conf` y establecemos el Server a 192.168.56.105, que es donde se encuentra el servidor de Zabbix

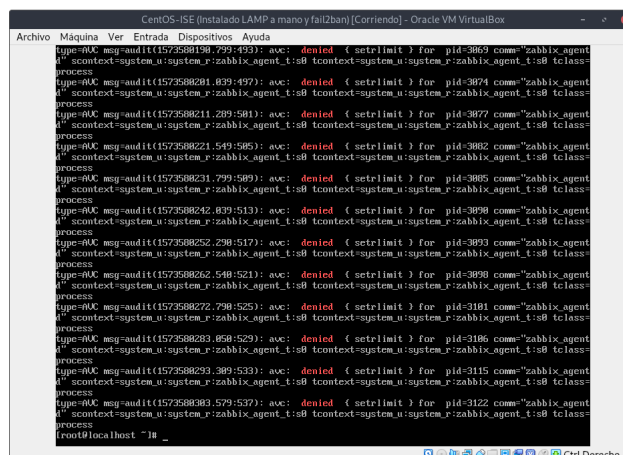


```
CentOS-ISE (Instalado LAMP a mano y fail2ban) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
#
# 1 - enabled
#
# Mandatory: no
# Default:
# LogRemoteCommands=0
##### Passive checks related
### Option: Server
#
# List of comma delimited IP addresses, optionally in CIDR notation, or hostnames of Zabbix se
#
# 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 e
#
# '0.0.0.0/0' 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.domain
#
# Mandatory: yes, if StartAgents is not explicitly set to 0
# Default:
# Server=
Server=192.168.56.185
### Option: ListenPort
#
# Agent will listen on this port for connections from the server.
#
# Mandatory: no
# Range: 1024-32767
# Default:
# ListenPort=10050
### Option: ListenIP
#
# List of comma delimited IP addresses that the agent should listen on.
# First IP address is sent to Zabbix server if connecting to it to retrieve list of active che
#
#
```

Pasamos a activar el servicio

```
1 # systemctl start zabbix-agent
```

Sin embargo, vemos que falla, y `journalctl -xe` junto con `/var/log/audit/audit.log` nos da-
mos cuenta de que el problema es SELinux, que no nos permite modificar el archivo de configura-
ción



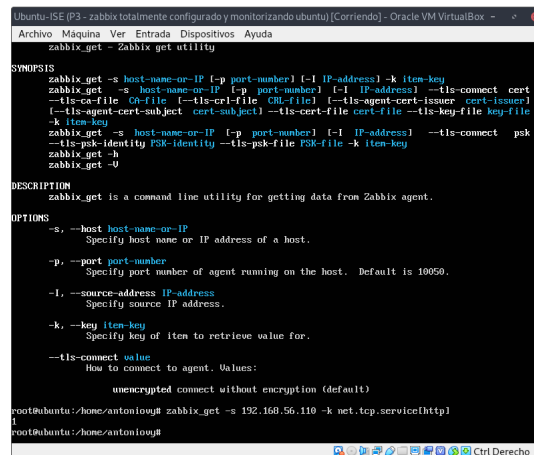
```
CentOS-ISE (Instalado LAMP a mano y fail2ban) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
type=AVC msg=audit(157259021:839:437): avc: denied (setrlimit) for pid=3674 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259021:839:437): avc: denied (setrlimit) for pid=3674 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259021:209:581): avc: denied (setrlimit) for pid=3677 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259021:549:585): avc: denied (setrlimit) for pid=3682 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259021:729:589): avc: denied (setrlimit) for pid=3685 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259021:839:513): avc: denied (setrlimit) for pid=3690 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259025:220:517): avc: denied (setrlimit) for pid=3693 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259025:549:521): avc: denied (setrlimit) for pid=3690 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259027:729:525): avc: denied (setrlimit) for pid=3181 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259030:659:529): avc: denied (setrlimit) for pid=3186 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259033:369:533): avc: denied (setrlimit) for pid=3115 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
type=AVC msg=audit(157259033:579:537): avc: denied (setrlimit) for pid=3122 comm="zabbix_agent
d" scontext=system_u:system_r:zabbix_agent_t:s0 tcontext=system_u:system_r:zabbix_agent_t:s0 tclass=
process
[root@localhost ~]#
```

Este es un problema común que no encontraremos en el manual, pero rápidamente podemos en-
contrar la solución en los foros de Zabbix[4].

Para resolver este problema usaremos la herramienta `audit2allow`[5], que nos permite crear una
política de SELinux a través del archivo `/var/log/audit/audit.log`

```
1 # grep zabbix_agent_t /var/log/audit/audit.log | audit2allow -M
  ↪ zabbix_agentd_custom
2 # semodule -i zabbix_agentd_custom.pp
3 # systemctl restart zabbix-agent
```

Y con esto, `zabbix-agent` ya está activado y funcionando, como podemos ver usando `zabbix_get`,
podemos acceder desde Ubuntu.



```
Ubuntu-ISE (P3 - zabbix totalmente configurado y monitorizando ubuntu) [Corriendo] - Oracle VM VirtualBox -
Archivo Máquina Ver Entrada Dispositivos Ayuda
zabbix_get - Zabbix get utility

SYNOPSIS
zabbix_get -s host-name-or-IP [-p port-number] [-I IP-address] -k item-key
zabbix_get -s host-name-or-IP [-p port-number] [-I IP-address] --tls-connect cert
--tls-ca-file Ca-file [--tls-crl-file CRL-file] [--tls-agent-cert-issuer cert-issuer]
[--tls-agent-cert-subject cert-subject] --tls-cert-file cert-file --tls-key-file key-file
-k item-key
zabbix_get -s host-name-or-IP [-p port-number] [-I IP-address] --tls-connect psk
--tls-psk-identity PSK-identity --tls-psk-file PSK-file -k item-key
zabbix_get -h
zabbix_get -V

DESCRIPTION
zabbix_get is a command line utility for getting data from Zabbix agent.

OPTIONS
-s, --host host-name-or-IP
    Specify host name or IP address of a host.
-p, --port port-number
    Specify port number of agent running on the host. Default is 10050.
-I, --source-address IP-address
    Specify source IP address.
-k, --key item-key
    Specify key of item to retrieve value for.
--tls-connect value
    How to connect to agent. Values:
    unencrypted connect without encryption (default)
    root@ubuntu:/home/antonio# zabbix_get -s 192.168.56.110 -k net.tcp.service[http]
1
root@ubuntu:/home/antonio#
```

3. Monitorizar máquinas

En esta sección se muestra contenido común a Ubuntu y CentOS, la única diferencia será la IP, que cuando estemos en el host de Ubuntu será 127.0.0.1, es decir localhost, ya que se monitoriza a si mismo, mientras que para referirnos al host de CentOS usaremos 192.168.56.110, ya que es la IP donde se encuentra CentOS en nuestra red.

3.1. Añadir máquinas a monitorizar:

Una vez tenemos configurado el nuevo usuario, pasamos a configurar el host, es decir, añadir en Zabbix la maquina que queremos monitorizar.

Por defecto, Zabbix nos añade la máquina en la que se ejecuta Zabbix, aunque su monitorización está desactivada por defecto. Yo dejare desactivado este y añadir uno nuevo aunque también tenga como host a localhost.

Para esto, desde la cuenta Admin añadimos un host en Administración ->Hosts.

Para crear un host indicamos el nombre que le queremos dar, la IP del host y el puerto (este último lo dejamos por defecto) y en Grupos añadimos Linux servers.

3.2. Añadir items a una máquina:

Ya tenemos la máquina añadida, ahora tenemos que decirle que partes del servidor queremos monitorizar.

Para esto, en la sección de host entramos en el nuevo host, y en la sección items creamos un item con lo que queramos monitorizar.

Por ejemplo, crearemos un item para la carga de CPU, en key tenemos que buscar y poner `system.cpu.load` y crearemos una nueva aplicación a la que llamaremos CPU donde almacenaremos todos los items que traten sobre la CPU.

The screenshot shows the Zabbix web interface for configuring a new item. The item is named 'Carga CPU' and is of type 'Zabbix agent'. The key is 'system.cpu.load'. The host interface is '127.0.0.1:10050'. The type of information is 'Numeric (unsigned)'. The units are empty. The update interval is '30s'. There is a custom interval table with one entry: 'Flexible' type, 'Scheduling' interval, '50s' period, and '1-7,00:00-24:00' action. The history storage period is '90d' and the trend storage period is '365d'. The show value is 'As is'. The new application is 'CPU'. The applications list shows 'None' and 'CPU'. The populates host inventory field is 'None'. The description is empty.

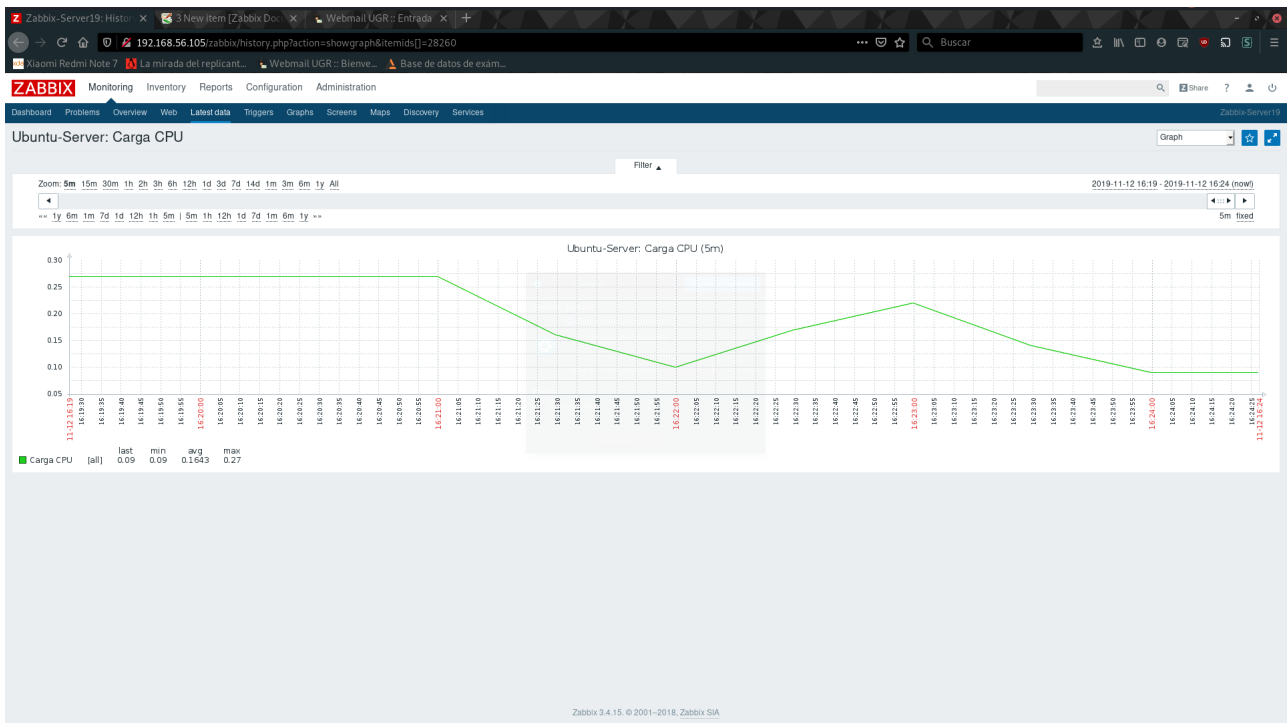
Type	Interval	Period	Action
Flexible	Scheduling	50s	1-7,00:00-24:00

3.3. Revisar los datos

Y con esto, ya vemos que vemos la carga de la CPU de Ubuntu en la sección Monitoring -> Latest data:

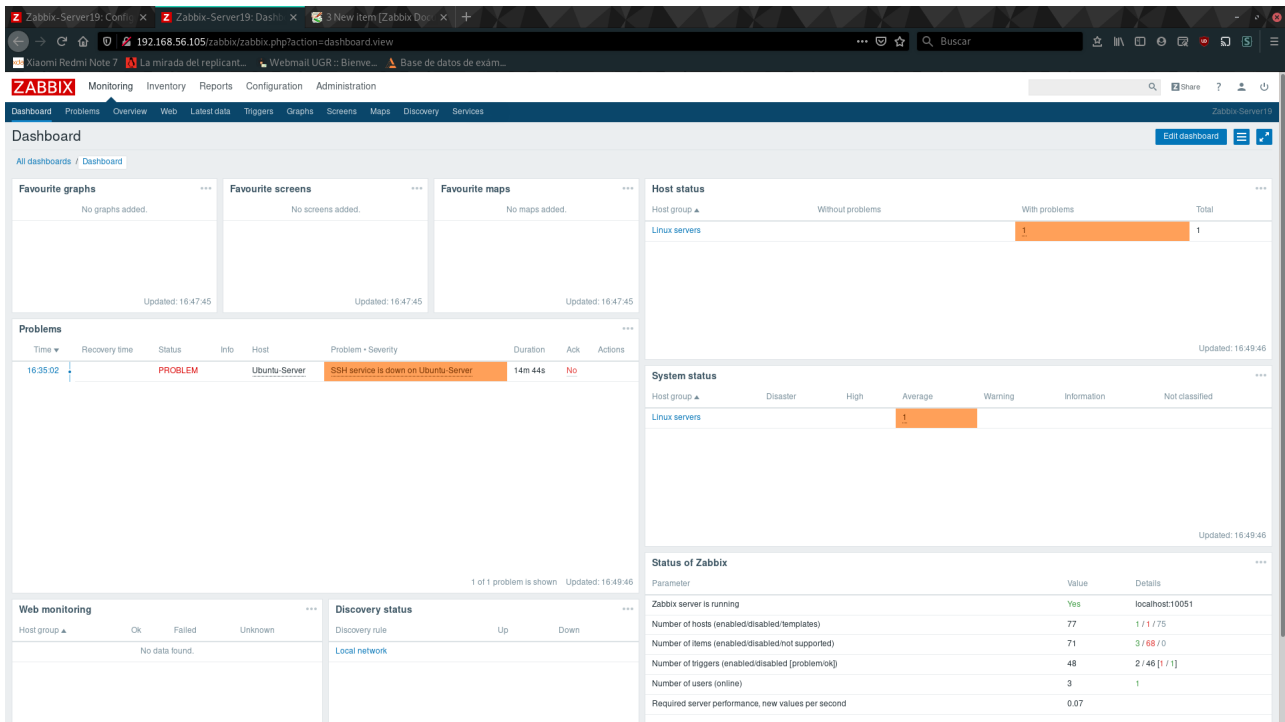
The screenshot shows the Zabbix web interface for the 'Latest data' section. The filters are set to 'Linux servers' for host groups, 'CPU (1 item)' for applications, and 'Carga CPU' for the specific item. The table shows the latest data for the 'Carga CPU' item on the 'Ubuntu-Server' host.

Host	Name	Last check	Last value	Change
Ubuntu-Server	CPU (1 item)			
	Carga CPU	2019-11-12 16:23:00	0.22	+0.05



Yo además he añadido dos templates, una para monitorizar SSH y otra para el servidor HTTP, usando la sección templates.

Vemos como HTTP no da ningun problema, sin embargo, SSH esta caído, esto se debe a que por defecto, el template de SSH tiene el puerto 22, y nosotros usamos el puerto 22022:



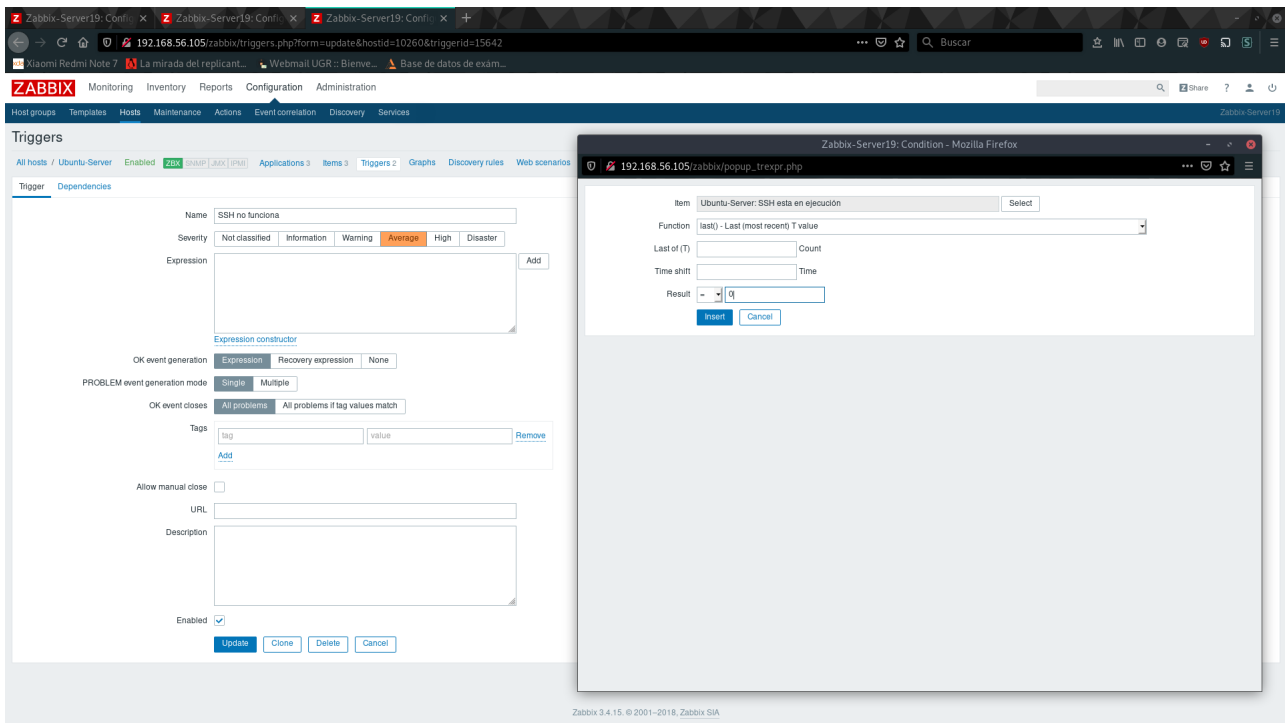
Eliminamos el template de SSH, y lo añadimos manualmente, usando la clave `net.tcp.service[ssh,127.0.0.1,22022]` que nos dice que consulte si el servicio SSH esta activo en la IP 127.0.0.1 con el puerto 22022 (usamos la IP 127.0.0.1 ya que es la IP de la máquina)

Sin embargo, ya no tenemos el aviso que nos añade el template, y que ya lo ha hecho con HTTP.

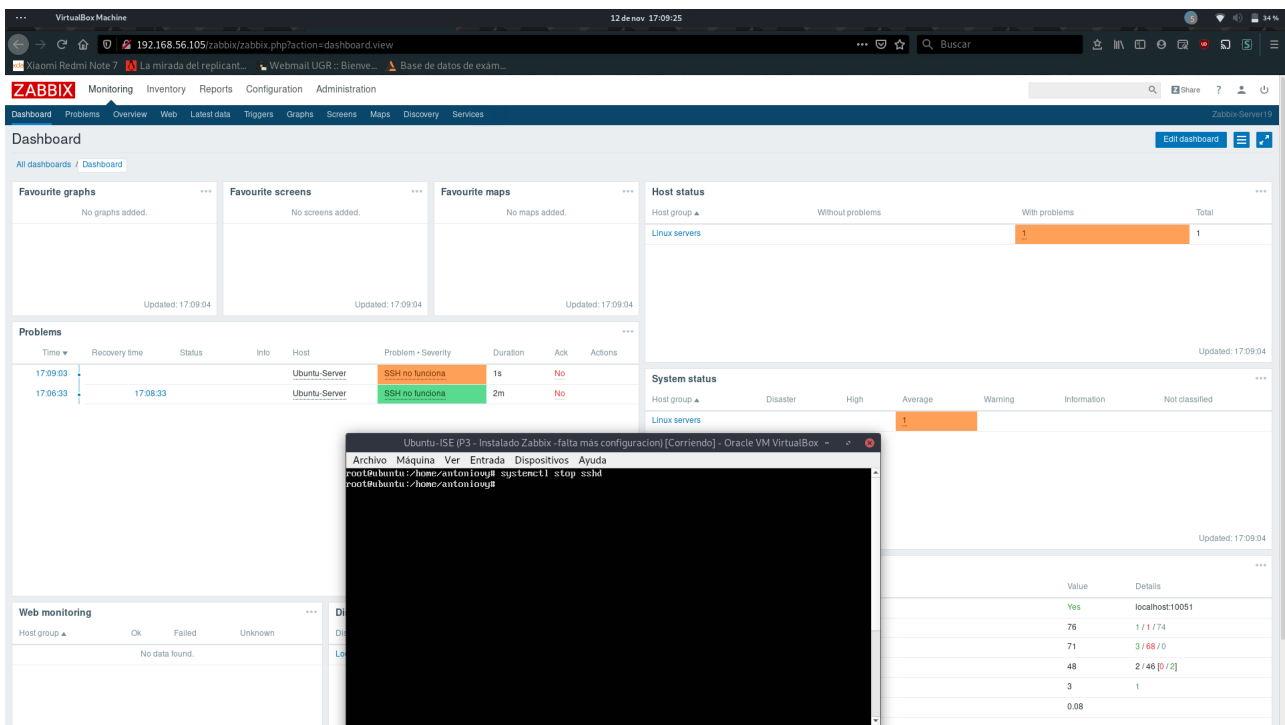
Vamos a añadirlo nosotros añadiendo un trigger al host.

3.4. Triggers

Al añadirlo nos preguntará la expresión, que en nuestro caso diremos que será del item que gestiona SSH, si el último valor obtenido vale 0 (esta caído el servicio).



De esta forma, si paramos el servicio sshd, zabbix nos avisará:



Vemos también como hay un problema resuelto, esto se debe a que primero realicé una prueba parando sshd, al volver a iniciar el sshd Zabbix nos avisa que el problema ha sido resuelto.

4. Conclusiones

Como conclusión, vemos que Zabbix es un programa potente y versátil que permite monitorizar servidores de forma bastante sencilla, y además con una buena documentación y manuales.

Los pocos problemas que hemos encontrado (manejo de SELinux) le hemos podido encontrar rápidamente solución gracias a la comunidad de Zabbix y sus foros, los cuales son bastante útiles y con bastante actividad entre usuarios.

Aunque nosotros hemos configurado Zabbix de una forma simple la complejidad de este software puede llegar a ser alta debido a la gran cantidad de opciones que ofrece, haciendo de este un buen servicio tanto para pequeños servidores como para sistemas más complejos.

Bibliografía

- [1] Manual Zabbix <https://www.zabbix.com/documentation/3.4/manual>
- [2] Crear BD para Zabbix https://www.zabbix.com/documentation/3.4/manual/appendix/install/db_scripts#mysql
- [3] Zabbix quickstart <https://www.zabbix.com/documentation/3.4/manual/quickstart/login>
- [4] Configurar Zabbix con SELinux <https://www.zabbix.com/forum/zabbix-help/367261-selinux-and-zabbix>
- [5] Manual audit2allow <https://linux.die.net/man/1/audit2allow>