

Práctica 3: Monitorización usando Zabbix

Alberto Jesús Durán López

13 de mayo de 2018

Índice

1. Introducción	3
2. Ubuntu Server	3
2.1. Instalación	3
2.2. Base de datos - MySQL	3
2.3. Iniciamos el servidor	5
2.4. Primer problema encontrado	7
2.5. Configuración del frontend	8
2.6. Revisión de errores	11
3. CentOS	13
3.1. Instalación	13

Índice de figuras

2.1. Instalación del paquete	3
2.2. Base de datos configurada	4
2.3. /etc/zabbix/zabbix_server.conf	5
2.4. Servicios	6
2.5. Base de datos configurada	7
2.6. Configuración frontend	8
2.7. Nuevo host CentOS	9
2.8. Templates CentOS	9
2.9. Templates frontend	10
2.10. Error en los servicios	10
2.11. /var/log/zabbix/zabbix_agentd.log	11
2.12. Ubuntu server: Zabbix_get	11
2.13. CentOS: Zabbix_get	11
2.14. Prueba de comandos	12
2.15. Zabbix - Correcto funcionamiento	12

1. Introducción

Zabbix es un Sistema de Monitorización de Redes diseñado para monitorizar y registrar el estado de varios servicios de red, Servidores, y hardware de red.

Usa MySQL, PostgreSQL, SQLite, Oracle o IBM DB2 como base de datos. Su backend está escrito en C y el frontend web está escrito en PHP.

En nuestro problema, tendremos que instalar Zabbix 3.4 en Ubuntu Server 16.04 y lo configuraremos para monitorizarlo a él mismo y a CentOS.

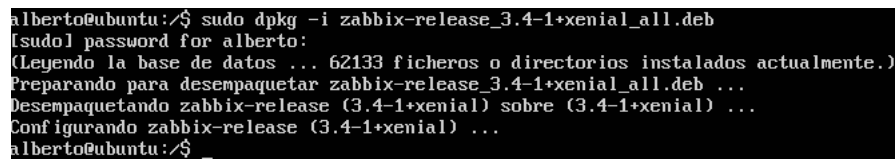
Los servicios a monitorizar serán SSH y HTTP.

2. Ubuntu Server

2.1. Instalación

-Instalamos el paquete de configuración del repositorio:

```
# wget http://repo.zabbix.com/zabbix/3.4/ubuntu/pool/main/z/zabbix-release/zabbix-release_3.4-1+xenial_all.deb
# dpkg -i zabbix-release_3.4-1+xenial_all.deb
```



```
alberto@ubuntu:/$ sudo dpkg -i zabbix-release_3.4-1+xenial_all.deb
[sudo] password for alberto:
(Leyendo la base de datos ... 62133 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar zabbix-release_3.4-1+xenial_all.deb ...
Desempaquetando zabbix-release (3.4-1+xenial) sobre (3.4-1+xenial) ...
Configurando zabbix-release (3.4-1+xenial) ...
alberto@ubuntu:/$ _
```

Figura 2.1: Instalación del paquete

```
# apt-get update
```

-Instalamos el servidor de Zabbix:

```
# apt-get install zabbix-server-mysql
```

-Instalamos la interfaz:

```
# apt-get install zabbix-frontend-php
```

Sin embargo, no tenemos que instalar el proxy ya que únicamente tenemos que monitorizar Ubuntu Server y CentOS y éste se usa cuando se monitorizan muchos nodos

2.2. Base de datos - MySQL

El servidor de Zabbix necesita una base de datos para almacenar toda la monitorización. Usaremos MySQL, realizamos la siguiente configuración:

```
# mysql -uroot -ppracticas,ISE
```

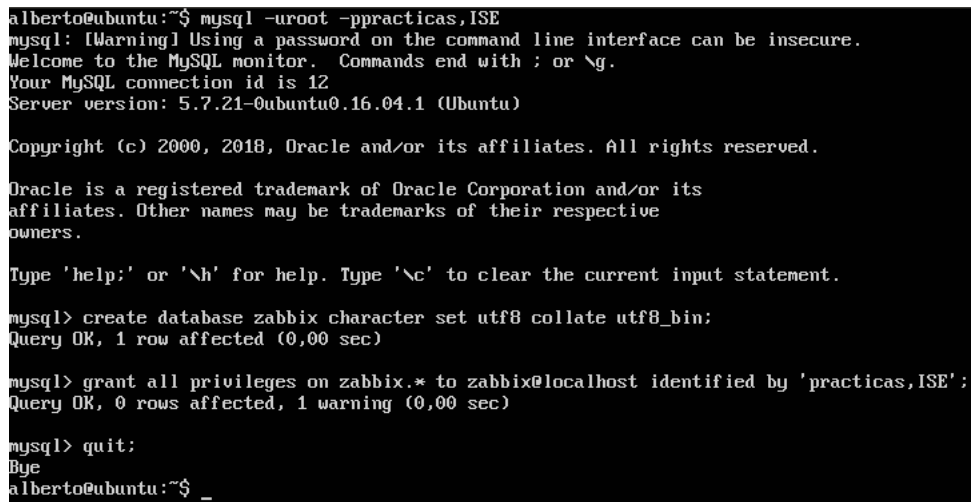
donde **practicasISE** es la contraseña que usaremos.

Una vez abierto el prompt de **MySQL**, creamos y damos permisos a la base de datos:

```
# create database zabbix character set utf8 collate utf8_bin;
```

```
# grant all privileges on zabbix.* to zabbix@localhost identified by 'practicas,ISE';
```

```
# quit;
```



```
alberto@ubuntu:~$ mysql -uroot -ppracticas,ISE
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.7.21-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, 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
alberto@ubuntu:~$ _
```

Figura 2.2: Base de datos configurada

Ahora que la tenemos configurada, importamos los datos:

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

Configuramos la base de datos creada para el servidor. Para ello, debemos editar el archivo de configuración **zabbix_server.conf** ubicado en `/etc/zabbix/`.

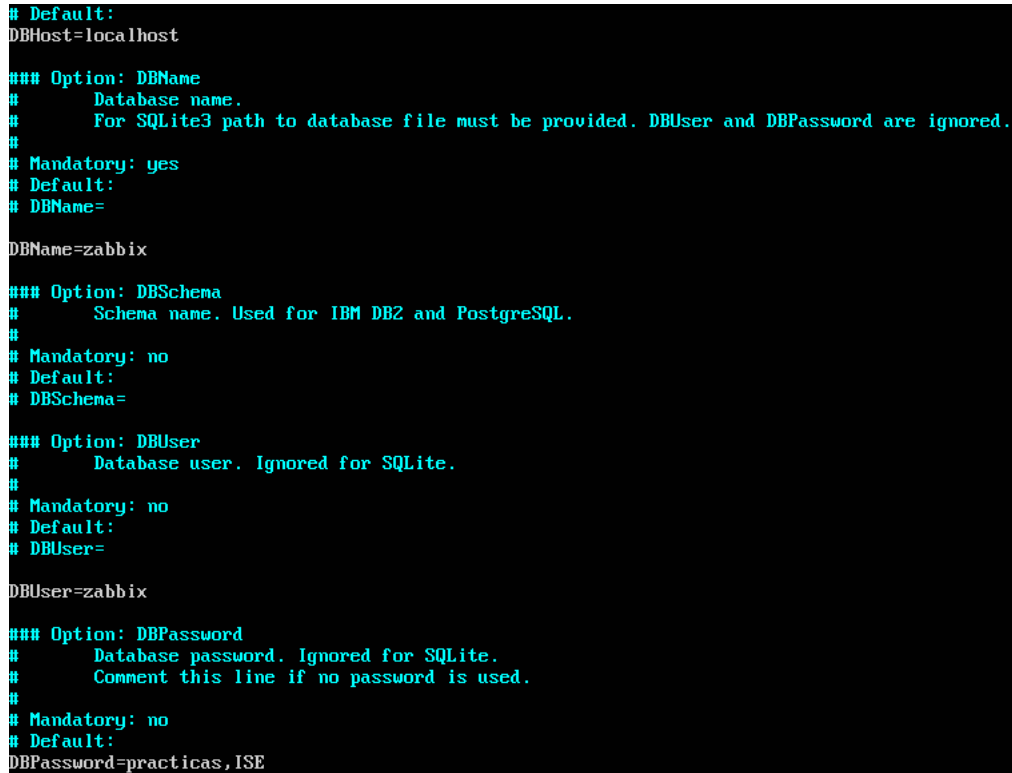
```
# vi /etc/zabbix/zabbix_server.conf
```

```
DBHost=localhost
```

```
DBName=zabbix
```

```
DBUser=zabbix
```

```
DBPassword=practicas,ISE
```



```
# Default:
DBHost=localhost

### Option: DBName
# Database name.
# For SQLite3 path to database file must be provided. DBUser and DBPassword are ignored.
#
# Mandatory: yes
# Default:
# DBName=

DBName=zabbix

### Option: DBSchema
# Schema name. Used for IBM DB2 and PostgreSQL.
#
# Mandatory: no
# Default:
# DBSchema=

### Option: DBUser
# Database user. Ignored for SQLite.
#
# Mandatory: no
# Default:
# DBUser=

DBUser=zabbix

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

Figura 2.3: `/etc/zabbix/zabbix_server.conf`

2.3. Iniciamos el servidor

El siguiente paso a realizar será iniciar el servidor de Zabbix:

```
# service zabbix-server start
```

```
# update-rc.d zabbix-server enable
```

Reiniciamos Apache:

```
# service apache2 restart
```

```
alberto@ubuntu:/$ service zabbix-server start
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to start 'zabbix-server.service'.
Authenticating as: alberto,,, (alberto)
Password:
==== AUTHENTICATION COMPLETE ====
alberto@ubuntu:/$ update-rc.d zabbix-server enable
insserv: fopen(.depend.stop): Permission denied
==== AUTHENTICATING FOR org.freedesktop.systemd1.reload-daemon ====
Se requiere autenticación para recargar el estado de systemd.
Authenticating as: alberto,,, (alberto)
Password:
==== AUTHENTICATION COMPLETE ====
alberto@ubuntu:/$ service apache2 restart
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to restart 'apache2.service'.
Authenticating as: alberto,,, (alberto)
Password:
==== AUTHENTICATION COMPLETE ====
alberto@ubuntu:/$
```

Figura 2.4: Servicios

Es recomendable que modifiquemos el archivo de la configuración de apache `/etc/apache2/conf-enabled/zabbix.conf`, descomentando la línea `'date.timezone'` y establecer nuestra zona horaria, que se puede consultar en <http://php.net/manual/en/timezones.php>

```
# vi /etc/apache2/conf-enabled/zabbix-conf
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 always_populate_raw_post_data -1
# php_value date.timezone Europe/Riga
```

Tenemos instalado el servidor en Ubuntu Server, sin embargo, es necesario instalar también el agente:

```
# apt-get install zabbix-agent
```

Como es un servicio, lo tenemos que habilitar:

```
# service zabbix-agent start
```

Ahora comprobamos: **`http://192.168.56.105/zabbix`**

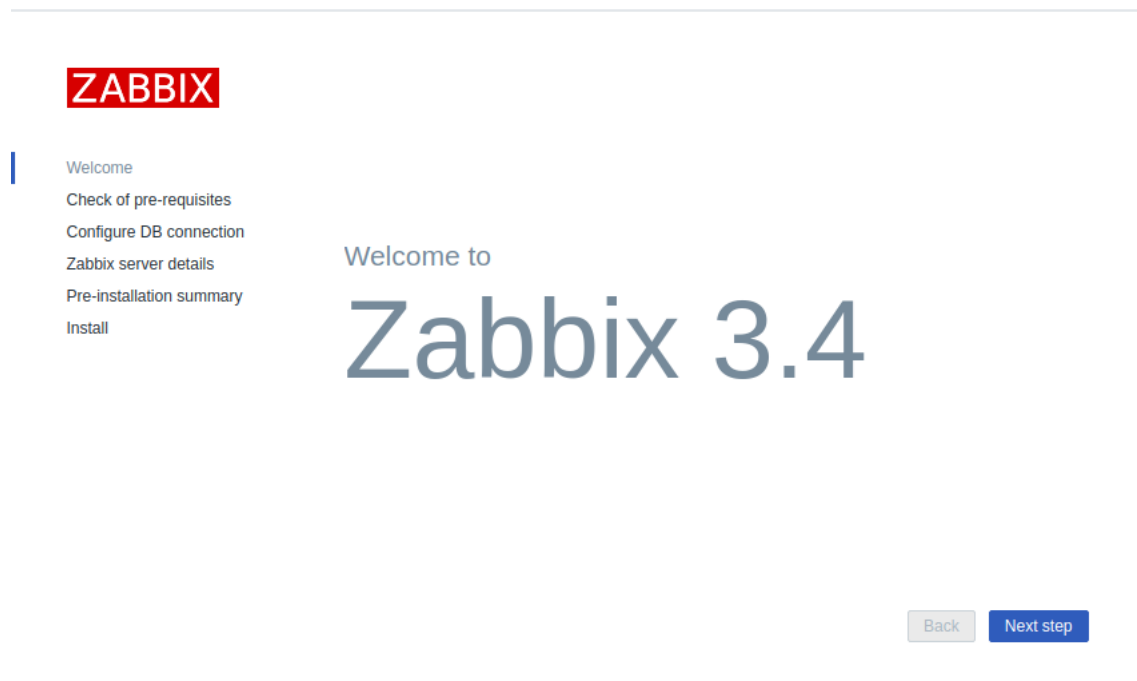


Figura 2.5: Base de datos configurada

2.4. Primer problema encontrado

Cuando intentaba conectarme a la dirección *192.168.56.105/zabbix* me daba un mensaje de error. Para solucionarlo he seguido los siguientes pasos:

- Compruebo que tengo conexión entre el host y ubuntu con:
#ping 192.168.56.1
- Compruebo la conexión de zabbix con el comando:
#curl localhost/zabbix
- Como el protocolo **http** tiene el puerto 80, lo habilitamos:
#ufw allow 80
- Volvemos a comprobar conectarnos a la dirección y vemos que el problema ya ha sido solucionado

2.5. Configuración del frontend

- Comprobamos que todos los prerequisites se cumplen
- Configuramos la conexión a la base de datos, para ello la base de datos debe estar creada
- Rellenamos los detalles del servidor
- Comprobamos que los datos introducidos son correctos:

ZABBIX

Welcome

Check of pre-requisites

Configure DB connection

Zabbix server details

Pre-installation summary

Install

Pre-installation summary

Please check configuration parameters. If all is correct, press "Next step" button, or "Back" button to change configuration parameters.

Database type	MySQL
Database server	localhost
Database port	default
Database name	zabbix
Database user	zabbix
Database password	*****

Zabbix server	localhost
Zabbix server port	10051
Zabbix server name	

Back Next step

Figura 2.6: Configuración frontend

Si hemos configurado correctamente Ubuntu Server y CentOS, realizaremos los siguiente pasos:

- Desde el frontend, nos dirigimos a *Configuration/Host* y añadimos lo siguiente:

The screenshot shows the 'Hosts' configuration page in Nagios XI. The 'Host' tab is selected. The 'Host name' field is set to 'CentOS Agent'. The 'Visible name' field is empty. Under 'Groups', 'Linux servers' is selected in the 'In groups' list. The 'Other groups' list includes 'Discovered hosts', 'Hypervisors', 'Templates', and various template sub-items. A 'New group' field is highlighted with a green border. The 'Agent interfaces' section shows 'IP address' as '192.168.56.105', 'DNS name' as empty, 'Connect to' as 'IP', 'Port' as '10050', and a 'Default' radio button selected. There are 'Add' and 'Remove' buttons.

Figura 2.7: Nuevo host CentOS

- En la pestaña *templates*, añadimos:

The screenshot shows the 'Hosts' configuration page with the 'Templates' tab selected. The breadcrumb shows 'All hosts / CentOS Agent'. The 'Enabled' status is shown. Below the tabs, there's a table of 'Linked templates' with columns 'Name' and 'Action'. It lists 'Template App HTTP Service' and 'Template App SSH Service', both with 'Unlink' actions. Below this is a 'Link new templates' section with a search input 'type here to search' and a 'Select' button. At the bottom are buttons for 'Update', 'Clone', 'Full clone', 'Delete', and 'Cancel'.

Figura 2.8: Templates centOS

- Hacer lo mismo para Ubuntu Server. En esta caso el proceso es igual ya que los procesos a monitorizar son los mismos, **ssh** y **http**. Lo únicamente que tendremos que cambiar son:
 - El nombre del host: Ubuntu Server and Agent
 - IP: 127.0.0.1 , ya que utiliza la misma IP que el servidor

Si todo ha ido correctamente, tendremos:

<input type="checkbox"/>	Name	Applications	Items	Triggers	Graphs	Discovery	Web	Interface	Templates	Status	Availability	Agent encryption	Info
<input type="checkbox"/>	Ubuntu Server and Agent	Applications 2	Items 2	Triggers 2	Graphs	Discovery	Web	127.0.0.1:10050	Template App HTTPS Service, Template App SSH Service	Enabled	ZBX SNMP JMX IPMI	NONE	
<input type="checkbox"/>	CentOS Agent	Applications 2	Items 2	Triggers 2	Graphs	Discovery	Web	192.168.56.105:10050	Template App HTTP Service, Template App SSH Service	Enabled	ZBX SNMP JMX IPMI	NONE	
<input type="checkbox"/>	Zabbix server	Applications 11	Items 87	Triggers 52	Graphs 16	Discovery 2	Web	127.0.0.1:10050	Template App Zabbix Server, Template OS Linux (Template App Zabbix Agent)	Enabled	ZBX SNMP JMX IPMI	NONE	

Displaying 3 of 3 found

Figura 2.9: Templates frontend

Nos dirigimos a la pestaña *Monitoring/DashBoard* y nos encontramos el siguiente error:

Problems								
Time	Recovery time	Status	Info	Host	Problem • Severity	Duration	Ack	Actions
11:39:12		PROBLEM		Ubuntu Server and Agent	SSH service is down on Ubuntu Server and Agent	3m 27s	No	
11:39:11		PROBLEM		Ubuntu Server and Agent	HTTPS service is down on Ubuntu Server and Agent	3m 28s	No	
11:39:10		PROBLEM		CentOS Agent	SSH service is down on CentOS Agent	3m 29s	No	
11:11:00	11:33:52	RESOLVED		Zabbix server	Zabbix agent on Zabbix server is unreachable for 5 minutes	22m 52s	No	

Figura 2.10: Error en los servicios

Si comprobamos la captura anterior, vemos que hemos introducido el servicio *HTTPS* en lugar de *HTTP* por lo que para solucionarlo, cambiamos dicho servicio en la pestaña de templates.

Por otro lado, los servicios están inactivos por lo que los habilitamos:

```
# systemctl restart sshd
```

Por último, como usamos el puerto 22022 para ssh, cambiamos dicho puerto en el item del template

All templates / Template App SSH Service Applications 1 Items 1 Triggers 1 Graphs Screens Discovery rules Web scenarios

Item Preprocessing

Name SSH service is running

Type Simple check

Key net.tcp.service[ssh „22022] Select

2.6. Revisión de errores

En caso de aparecer un mensaje de conexión inválida, podemos ver los logs del proceso en `/var/log/zabbix`, tanto en Ubuntu Server como en CentOS

```
13369:20180423:194200.640 agent #1 started [collector]
13370:20180423:194200.641 agent #2 started [listener #1]
13371:20180423:194200.642 agent #3 started [listener #2]
13372:20180423:194200.644 agent #4 started [listener #3]
13373:20180423:194200.645 active check configuration update from [127.0.0.1:10051] started to fail (cannot connect to [127.0.0.1]:10051): [111] Connection refused)
1115:20180513:021521.692 Starting Zabbix Agent [Zabbix server]. Zabbix 3.4.9 (revision 80550).
1115:20180513:021521.698 **** Enabled features ****
1115:20180513:021521.698 IPv6 support: YES
1115:20180513:021521.698 TLS support: YES
1115:20180513:021521.698 *****
1115:20180513:021521.698 using configuration file: /etc/zabbix/zabbix_agentd.conf
1115:20180513:021521.700 agent #0 started [main process]
1130:20180513:021521.701 agent #5 started [active checks #1]
1130:20180513:021521.703 active check configuration update from [127.0.0.1:10051] started to fail (cannot connect to [127.0.0.1]:10051): [111] Connection refused)
1129:20180513:021521.703 agent #4 started [listener #3]
1128:20180513:021521.705 agent #3 started [listener #2]
1127:20180513:021521.706 agent #2 started [listener #1]
1126:20180513:021521.708 agent #1 started [collector]
[root@localhost zabbix]#
```

Figura 2.11: `/var/log/zabbix/zabbix_agentd.log`

El problema anterior se ha solucionado introduciendo la ip correcta en los archivos de configuración de zabbix.

Para probar que el proceso ha finalizado correctamente, podemos hacerlo con la orden:
`# zabbix_get -s 'dirIP' -k net.tcp.service['Service name']`
que previamente hemos instalado tanto en CentOS como en Ubuntu Server

Ubuntu Server:

- `sudo apt-cache search zabbix`
`sudo apt-get install zabbix-get`

CentOS

- `yum search zabbix-get`
`yum install zabbix-get.x86_64`

```
alberto@ubuntu:~$ zabbix_get -s 192.168.56.105 -k net.tcp.service[http]
1
```

Figura 2.12: Ubuntu server: Zabbix_get

```
[root@localhost zabbix]# zabbix_get -s 192.168.56.110 -k net.tcp.service[http]
1
```

Figura 2.13: CentOS: Zabbix_get

Por otro lado, también hemos probado que zabbix funciona correctamente parando y reiniciando el servicio de la forma:

```
[root@localhost zabbix]# systemctl stop sshd
[root@localhost zabbix]# systemctl restart sshd
[root@localhost zabbix]# systemctl stop sshd
[root@localhost zabbix]# systemctl stop httpd
[root@localhost zabbix]# systemctl restart sshd
[root@localhost zabbix]# systemctl restart httpd
[root@localhost zabbix]#
```

Figura 2.14: Prueba de comandos

Problems ...								
Time ▼	Recovery time	Status	Info	Host	Problem • Severity	Duration	Ack	Actions
12:30:51	12:31:51	RESOLVED		CentOS Agent	SSH service is down on CentOS Agent	1m	No	
12:30:50	12:31:50	RESOLVED		CentOS Agent	HTTP service is down on CentOS Agent	1m	No	
12:26:51	12:27:51	RESOLVED		CentOS Agent	SSH service is down on CentOS Agent	1m	No	
12:00								
11:57:53	12:12:53	RESOLVED		Ubuntu Server	SSH service is down on Ubuntu Server	15m	No	
11:57:51	12:12:51	RESOLVED		CentOS Agent	SSH service is down on CentOS Agent	15m	No	

Figura 2.15: Zabbix - Correcto funcionamiento

3. CentOS

En CentOS únicamente tendremos que instalar el agente por lo que no tenemos que instalar una base de datos

3.1. Instalación

Nos descargamos el siguiente repositorio:

```
# rpm -ivh http://repo.zabbix.com/zabbix/3.4/rhel/7/x86_64/zabbix-release-3.4-2.el7.noarch.rpm
```

Instalamos el agente de Zabbix:

```
# yum install zabbix-agent
```

Si probamos a iniciar el agente nos dará error, por lo que haremos lo siguiente:

```
# cat /var/log/audit/audit.log | grep zabbix_agentd | grep denied | audit2allow -M zabbix_agent_setrlimit
```

```
# semodule -i zabbix_agent_setrlimit.pp
```

```
# systemctl restart zabbix-agent
```

```
[root@localhost ~]# cat /var/log/audit/audit.log | grep zabbix_agentd | grep denied | audit2allow -M zabbix_agent_setrlimit
***** IMPORTANT *****
To make this policy package active, execute:

semodule -i zabbix_agent_setrlimit.pp

[root@localhost ~]# semodule -i zabbix_agent_setrlimit.pp
[root@localhost ~]# systemctl restart zabbix-agent
```

Ahora ya podemos iniciar el agente sin problemas:

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

```
# systemctl status zabbix-agent
```

```
[root@localhost ~]# systemctl status zabbix-agent
■ zabbix-agent.service - Zabbix Agent
   Loaded: loaded (/usr/lib/systemd/system/zabbix-agent.service; disabled; vendor preset: disabled)
   Active: active (running) since lun 2018-05-07 21:40:01 CEST; 49min ago
     Process: 2474 ExecStop=/bin/kill -SIGTERM $MAINPID (code=exited, status=0/SUCCESS)
     Process: 2476 ExecStart=/usr/sbin/zabbix_agentd -c $CONFFILE (code=exited, status=0/SUCCESS)
    Main PID: 2478 (zabbix_agentd)
      CGroup: /system.slice/zabbix-agent.service
              └─2478 /usr/sbin/zabbix_agentd -c /etc/zabbix/zabbix_agentd.conf
                 └─2479 /usr/sbin/zabbix_agentd: collector [idle 1 sec]
                    └─2480 /usr/sbin/zabbix_agentd: listener #1 [waiting for connection]
                       └─2481 /usr/sbin/zabbix_agentd: listener #2 [waiting for connection]
                          └─2482 /usr/sbin/zabbix_agentd: listener #3 [waiting for connection]
                             └─2483 /usr/sbin/zabbix_agentd: active checks #1 [idle 1 sec]

may 07 21:40:01 localhost.localdomain systemd[1]: Starting Zabbix Agent...
may 07 21:40:01 localhost.localdomain systemd[1]: PID file /run/zabbix/zabbix...
may 07 21:40:01 localhost.localdomain systemd[1]: Started Zabbix Agent.
```

Referencias

- [1] Zabbix Documentation,
<https://www.zabbix.com/documentation/3.4/manual/installation>
- [2] Ubuntu Server,
https://www.zabbix.com/documentation/3.4/manual/installation/install_from_packages/debian_ubuntu
- [3] CentOS,
https://www.zabbix.com/documentation/3.4/manual/installation/install_from_packages/rhel_centos