

Auditoría, monitorización e protección de sistemas finais (endpoints): wazuh

ESCENARIO: wazuh

Wazuh → Máquina virtual Debian:

└ Host
RAM ≥ 6144MB CPU ≥ 2
Disco duro: Debian amd64
docker
Wazuh: dashboard, manager, indexer
Rede: Rede NAT si-wazuh → 10.10.10.0/24
IP/MS: 10.10.10.10/24

Axente GNU/Linux

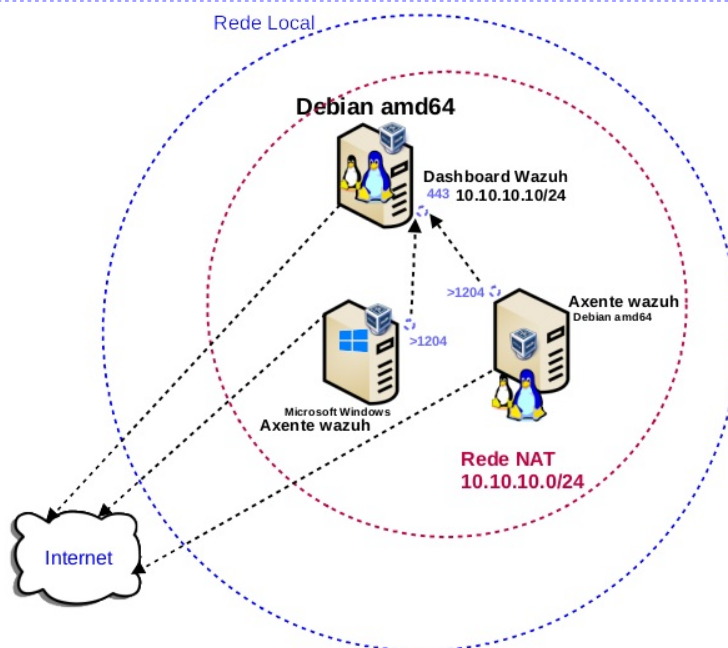
Máquina virtual Debian amd64:

└ Host
RAM ≥ 2048MB CPU ≥ 2
Disco duro: Debian amd64
Rede: Rede NAT si-wazuh → 10.10.10.0/24
IP/MS: dinámica

Axente Microsoft Windows

Máquina virtual Microsoft Windows:

└ Host
RAM ≥ 2048MB CPU ≥ 2
Disco duro: Windows amd64
Rede: Rede NAT si-wazuh → 10.10.10.0/24
IP/MS: dinámica



LIMITACIÓN DE RESPONSABILIDADE O autor do presente documento declina calquera responsabilidade asociada ao uso incorrecto e/ou malicioso que puidese realizarse coa información exposta no mesmo. Por tanto, non se fai responsable en ningún caso, nin pode ser considerado legalmente responsable en ningún caso, das consecuencias que poidan derivarse da información contida nel ou que esté enlazada dende ou hacia el, incluíndo os posibles erros e información incorrecta existentes, información difamatoria, así como das consecuencias que se poidan derivar sobre a súa aplicación en sistemas de información reais e/ou virtuais. Este documento foi xerado para uso didáctico e debe ser empregado en contornas privadas e virtuais controladas co permiso correspondente do administrador desas contornas.

NOTAS:

- [1] wazuh (XDR, SIEM)
- [2] docker
- [3] docker hub
- [4] Cheat Sheet Docker A3
- [5] wazuh - docker

Máquina virtual Debian amd64

1. Configuración da rede según o escenario

Na contorna gráfica abrir un terminal e executar:

user@debian:~\$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.

user@debian:~\$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
root@debian:~# /etc/init.d/avahi-daemon stop #Parar o demo avahi-daemon(control resolución de nomes) para poder configurar de forma manual a configuración de rede e non ter conflito con este demo.

root@debian:~# systemctl disable avahi-daemon #Impide que o servizo avahi-daemon sexa iniciado no arranque xerando os links K* nos runlevels (/etc/rcX.d)

root@debian:~# /etc/init.d/network-manager stop || pkill NetworkManager #Parar o demo network-manager(xestor de rede) ou o script NetworkManager (executado sen ser demo) para poder configurar doutro xeito (co comando ip(ifconfig) de forma manual ou mediante networking (ficheiros /etc/init.d/networking, /etc/init.d/networking.d) a configuración de rede e non ter conflito con este xestor.

root@debian:~# systemctl disable network-manager #Impide que o servizo network-manager sexa iniciado no arranque xerando os links K* nos runlevels (/etc/rcX.d)

root@debian:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, as tarxetas de redes: loopback(lo) e a correspondente á Rede NAT(enp0s3).

```
$ man interfaces #Ver ás páxinas de manual referente ao ficheiro de configuración de rede /etc/network/interfaces
$ cat /etc/network/interfaces #Amosar o contido do ficheiro configuración de rede /etc/network/interfaces
$ ls -l /etc/network/interfaces.d #Listar de forma extendida o contido do directorio /etc/network/interfaces/setup
$ cat /etc/network/interfaces.d/setup #Amosar o contido do ficheiro configuración de rede /etc/network/interfaces/setup
```

root@debian:~# cat > /etc/network/interfaces.d/setup <<EOF #Comezo do ficheiro a crear /etc/network/interfaces.d/setup
auto lo
iface lo inet loopback

auto enp0s3
iface enp0s3 inet static
address 10.10.10.10/24

EOF #Fin do ficheiro a crear /etc/network/interfaces.d/setup

root@debian:~# /etc/init.d/networking status #Comprobar o estado do demo networking, é dicir, comprobar se está activa a configuración de rede en /etc/network/interfaces (/etc/network/interfaces.d).

root@debian:~# /etc/init.d/networking start #Arrancar o demo networking, é dicir, activar a configuración de rede en /etc/network/interfaces (/etc/network/interfaces.d).

root@debian:~# /etc/init.d/networking status #Comprobar o estado do demo networking, é dicir, comprobar se está activa a configuración de rede en /etc/network/interfaces (/etc/network/interfaces.d).

root@debian:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, as tarxetas de redes: loopback(lo) e a correspondente á Rede NAT(enp0s3).

root@debian:~# ping -c4 10.10.10.10 #Comprobar mediante o comando ping a conectividade coa interface de rede local enp0s3

2. docker: Instalación e arranque

Executar no anterior terminal:

root@debian:~# apt update || apt-get update #Actualizar repositorios declarados no ficheiro /etc/apt/sources.list e nos ficheiros existentes no directorio /etc/apt/sources.list.d

Así, unha vez realizada a consulta dos ficheiros existentes nas rutas anteriores, descárganse uns ficheiros coas listas de paquetes posibles a instalar. Estes ficheiros son gardados en /var/lib/apt/lists

root@debian:~# apt -y install docker.io || apt-get -y install docker.io #Instalar o paquete de nome *docker.io*. Co parámetro -y automaticamente asumimos yes a calquera pregunta que ocorra na instalación do paquete.

root@debian:~# apt -y install docker-compose || apt-get -y install docker-compose #Instalar o paquete de nome *docker-compose*. Co parámetro -y automaticamente asumimos yes a calquera pregunta que ocorra na instalación do paquete.

root@debian:~# /etc/init.d/docker status || systemctl status docker #Comprobar o estado do demo docker

root@debian:~# /etc/init.d/docker start || systemctl start docker #Arrancar o demo docker

3. wazuh: Implantación en docker

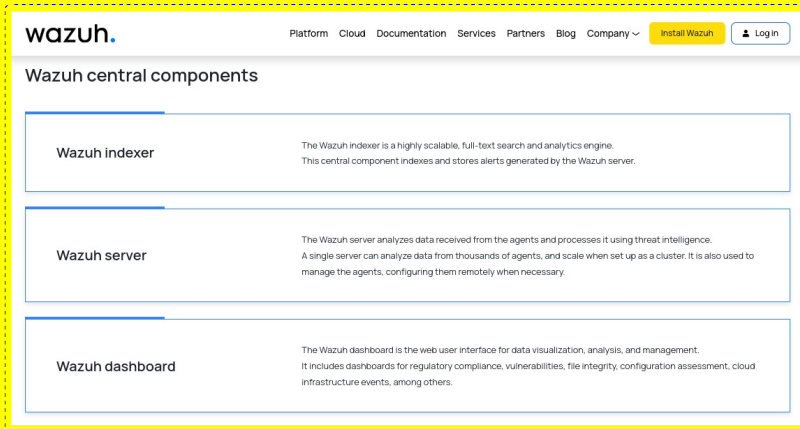
Na contorna gráfica abrir un terminal e executar:

```
root@debian:~# git clone https://github.com/wazuh/wazuh-docker.git -b v4.3.10 #Descargar wazuh dende github mediante git clone.
```

Wazuh permite auditar, monitorizar e protexer sistemas finais ou endpoints.

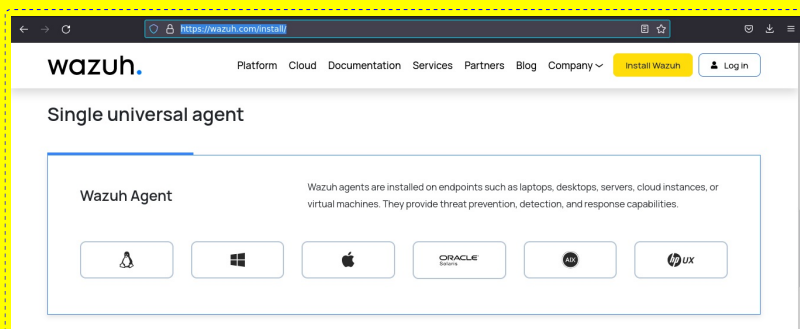
A solución Wazuh baséase no axente Wazuh, que se desprega nos puntos finais monitorizados, e en tres compoñentes centrais: o servidor Wazuh, o indexador de Wazuh e o panel de control de Wazuh:

- O indexador Wazuh é un motor de busca e análise de texto completo altamente escalable. Este compoñente central indexa e almacena as alertas xeradas polo servidor Wazuh.
- O servidor Wazuh analiza os datos recibidos dos axentes. Procesao mediante decodificadores e regras, utilizando intelixencia sobre ameazas para buscar indicadores de compromiso (IOC) coñecidos. Un único servidor pode analizar datos de centos ou miles de axentes e escalar horizontalmente cando se configura como un clúster. Este compoñente central tamén se utiliza para xestionar os axentes, configurándoos e actualizándoos de forma remota cando sexa necesario.
- O panel de control de Wazuh é a interface de usuario web para a visualización e análise de datos. Inclúe paneis listos para usar para eventos de seguridade, cumprimento normativo (por exemplo, PCI DSS, GDPR, CIS, HIPAA, NIST 800-53), aplicacións vulnerables detectadas, datos de seguimento da integridade dos ficheiros, resultados da avaliación da configuración, seguimento da infraestrutura na nube, eventos e outros. Tamén se usa para xestionar a configuración de Wazuh e supervisar o seu estado.



Os axentes Wazuh instálanse en puntos finais(endpoints) como ordenadores portátiles, escritorios, servidores, instancias de nube ou máquinas virtuais. Ofrecen capacidades de prevención, detección e resposta de ameazas. Así, podemos desplegar **axentes wazuh (Single Universal Agent)**

sobre sistemas operativos: GNU/Linux, Microsoft Windows, MacOS, Solaris, AIX, HP/UX.



```
root@debian:~# cd wazuh-docker/single-node #Acceder ao cartafol wazuh-docker/single-node.
```

```
root@debian:~# docker-compose -f generate-indexer-certs.yml run --rm generator #Xerar certificados para cada nodo para asegurar a comunicación entre os nodos.
```

```
root@debian:~# docker-compose up #Despregar os contenedores docker a través de docker-compose. Como non se indica coa opción -f un ficheiro, o ficheiro a interpretar será docker-compose.yml
```

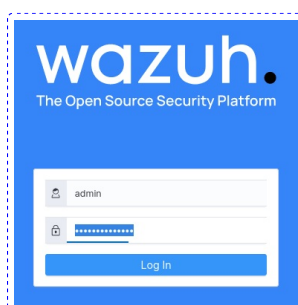
```
root@debian:~# cat docker-compose.yml #Ver o contido do ficheiro wazuh-docker/single-node/docker-compose.yml
```

```
root@debian:~# docker container ls #Listar os contenedores docker activos.
```

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|--------------|------------------------------|--------------------------|---------------|--------------|--|-------------------------------|
| aa59dabed15c | wazuh/wazuh-dashboard:4.3.10 | "/entrypoint.sh" | 6 minutes ago | Up 6 minutes | 443/tcp, 0.0.0.0:443->5601/tcp | single-node-wazuh-dashboard_1 |
| d2dfe78c1add | wazuh/wazuh-indexer:4.3.10 | "/entrypoint.sh open..." | 6 minutes ago | Up 6 minutes | 0.0.0.0:9200->9200/tcp | single-node-wazuh-indexer_1 |
| 4351a01c6578 | wazuh/wazuh-manager:4.3.10 | "/init" | 6 minutes ago | Up 6 minutes | 0.0.0.0:1514-1515->1514-1515/tcp, 0.0.0.0:514->514/udp, 0.0.0.0:55000->55000/tcp, 1516/tcp | single-node-wazuh-manager_1 |

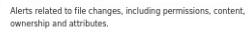
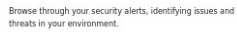
```
root@debian:~# firefox https://localhost:443 & #Lanzar o navegador firefox na URL https://localhost:443 no porto TCP 443, realizando a execución en segundo plano (&), é dicir, acceder ao dashboard(panel de control) de wazuh.
```

Credenciales de acceso predeterminadas: **admin/SecretPassword**

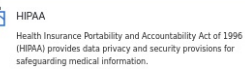
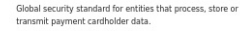
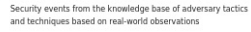


Never connected agents
0

AUDITING AND POLICY MONITORING



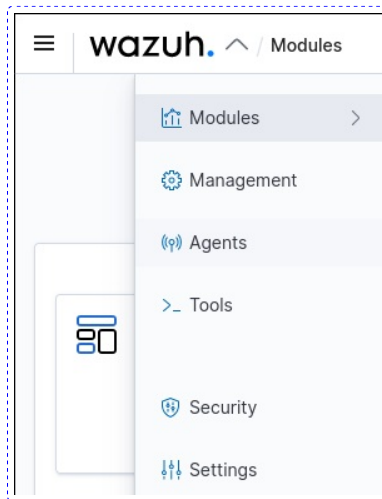
REGULATORY COMPLIANCE



4. Engadir axentes

A. Debian GNU/Linux

1. Acceder a Agents

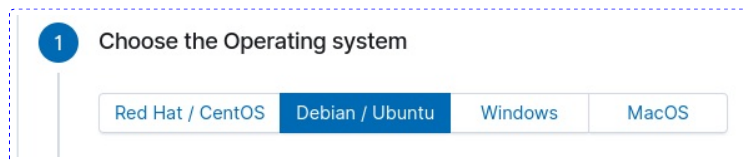


2. Acceder a Deploy new agent



3. Elixir:

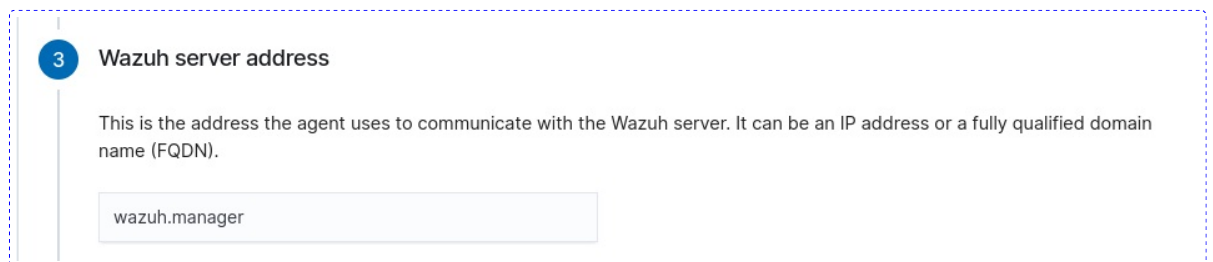
- Sistema operativo: Debian/Ubuntu



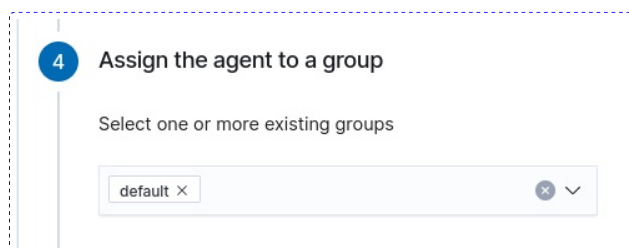
- Arquitectura: amd64



- Dirección IP/FQDN servidor wazuh: wazuh.manager



- Asignar un grupo ao axente: default



- No apartado 5 (Instalar e engadir o axente) aparece un comando coas opcións escollidas. Este comando debe ser executado co usuario root no axente para engadilo como tal en wazuh:

5

Install and enroll the agent

You can use this command to install and enroll the Wazuh agent in one or more hosts.

③

If the installer finds another Wazuh agent in the system, it will upgrade it preserving the configuration.

```
curl -so wazuh-agent-4.3.10.deb https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.3.10-1_amd64.deb && sudo WAZUH_MANAGER='wazuh.manager' WAZUH_AGENT_GROUP='default' dpkg -i ./wazuh-agent-4.3.10.deb
```

- No apartado 6 aparecen os comandos necesarios, según o sistema Systemd ou SysV init, para recargar o servizo pertencente ao axente (wazuh-agent) e así poidamos ver no dashboard a este novo axente agregado.

6

Start the agent

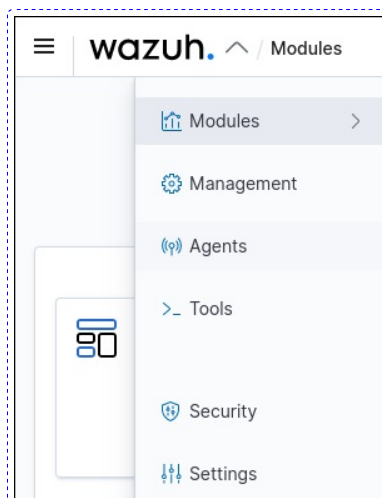
Systemd

SysV Init

```
sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
```

B. Microsoft Windows

1. Acceder a Agents



2. Acceder a Deploy new agent



3. Elixir:

- Sistema operativo: Microsoft Windows

1

Choose the Operating system

Red Hat / CentOS

Debian / Ubuntu

Windows

MacOS

- Dirección IP/FQDN servidor wazuh: 10.10.10.10

2

Wazuh server address

This is the address the agent uses to communicate with the Wazuh server. It can be an IP address or a fully qualified domain name (FQDN).

10.10.10.10

- Asignar un grupo ao axente: default

3

Assign the agent to a group

Select one or more existing groups

default

- No apartado 4 (Instalar e engadir o axente) aparece un comando coas opcións escollidas. Este comando debe ser executado co usuario administrador nunha consola de powershell no axente para engadilo como tal en wazuh:

4

Install and enroll the agent

You can use this command to install and enroll the Wazuh agent in one or more hosts.

① If the installer finds another Wazuh agent in the system, it will upgrade it preserving the configuration.

① Requirements

- You will need administrator privileges to perform this installation.
- PowerShell 3.0 or greater is required.

Keep in mind you need to run this command in a Windows PowerShell terminal.

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.3.10-1.msi -OutFile ${env:tmp}\wazuh-agent-4.3.10.msi; msixec.exe /i ${env:tmp}\wazuh-agent-4.3.10.msi /q WAZUH_MANAGER='wazuh.manager' WAZUH_REGISTRATION_SERVER='wazuh.manager' WAZUH_AGENT_GROUP='default'
```

- No apartado 6 aparecen os comandos necesarios, según o sistema Systemd ou SysV init, para recargar o servizo pertencente ao axente (wazuh-agent) e así poidamos ver no dashboard a este novo axente agregado.

5

Start the agent

NET START WazuhSvc

Máquinas virtuais axentes

5. Acceder ao axente Debian GNU/Linux: debianA

Executar nunha consola de comandos:

```
usuario@debianA:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
usuario@debianA:~$ ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, as tarxetas de redes: loopback(lo) e a
correspondente á Rede NAT(enp0s3).
usuario@debianA:~$ ping -c4 10.10.10.10 #Comprobar mediante o comando ping a conectividade co servidor wazuh
usuario@debianA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers,
visudo)
root@debianA:~# echo '10.10.10.10 wazuh.manager' >> /etc/hosts #Engadir no ficheiro /etc/hosts, é dicir, na táboa estática de
búsqueda para nomes de host (DNS) o nome wazuh.manager, para que atenda á IP 10.10.10.10
root@debianA:~# curl -so wazuh-agent-4.3.10.deb https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-
agent_4.3.10-1_amd64.deb && sudo WAZUH_MANAGER='wazuh.manager' WAZUH_AGENT_GROUP='default' dpkg -i
./wazuh-agent-4.3.10.deb #Executar o comando do apartado 3.A.5 que nos ofrece wazuh para poder agregar este sistema operativo como axente de
wazuh.
root@debianA:~# systemctl daemon-reload && systemctl enable wazuh-agent && systemctl start wazuh-agent Arrancar o
servizo do axente para que poida ser visto no dashboard de wazuh.
```

6. Acceder ao axente Microsoft Windows

Executar nun terminal:

```
> systeminfo #Amosar información de configuración detallada sobre o equipo e o seu sistema operativo.

> ipconfig /all #Amosar a configuración TCP/IP completa de todas as interfaces de rede.

> ping -c4 10.10.10.10 #Comprobar mediante o comando ping a conectividade co servidor wazuh
```

Executar en powershell con permisos de administrador:

```
> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.3.10-1.msi -OutFile
${env:tmp}\wazuh-agent-4.3.10.msi; msixexec.exe /i ${env:tmp}\wazuh-agent-4.3.10.msi /q
WAZUH_MANAGER='wazuh.manager' WAZUH_REGISTRATION_SERVER='wazuh.manager'
WAZUH_AGENT_GROUP='default' #Executar o comando do apartado 3.B.3 que nos ofrece wazuh para poder agregar este sistema operativo como
axente de wazuh.

> NET START WazuhSvc Arrancar o servizo do axente para que poida ser visto no dashboard de wazuh.
```

Dashboard: Máquina virtual Debian amd64

7. wazuh: Dashboard

Comprobar que os axentes engadidos aparecen no dashboard e que podemos acceder á súa monitorización.

Ricardo Feijoo Costa



This work is licensed under a **Creative Commons Attribution-ShareAlike 4.0 International License**