Welcome to Wazuh

<https://documentation.wazuh.com/current/index.html>

YouTube Link:

<https://www.youtube.com/watch?v=WqQUIz5XRpQ>

Installation Guide:

<https://documentation.wazuh.com/current/installation-guide/open-distro/all-in-one-deployment/unattended-installation.html>

Wazuh is a free and open source platform for threat detection, security monitoring, incident response and regulatory compliance. It can be used to monitor endpoints, cloud services and containers, and to aggregate and analyze data from external sources. Wazuh provides the following capabilities:

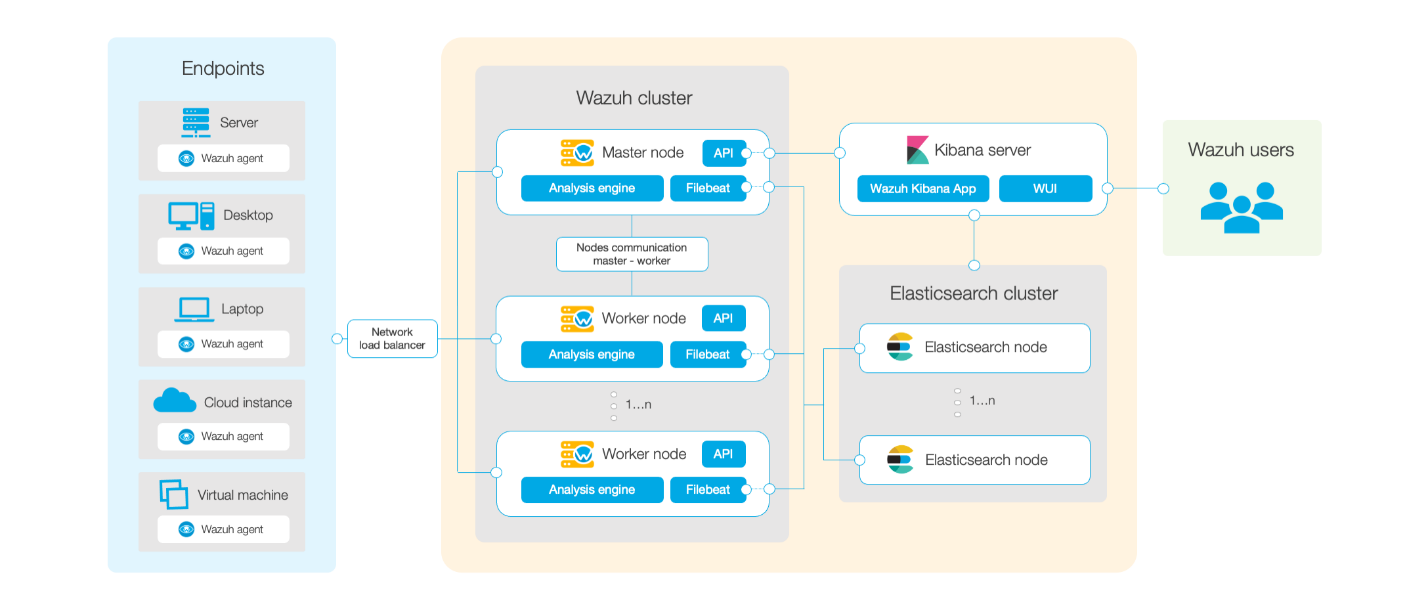
# Architecture[¶](https://documentation.wazuh.com/current/getting-started/architecture.html#architecture)

The Wazuh architecture is based on [agents](https://documentation.wazuh.com/current/getting-started/components/wazuh_agent.html#wazuh-agent), running on the monitored endpoints, that forward security data to a central [server](https://documentation.wazuh.com/current/getting-started/components/wazuh_server.html#wazuh-server). Moreover, agentless devices (such as firewalls, switches, routers, access points, etc.) are supported and can actively submit log data via Syslog, SSH, or using their own API. The central server decodes and analyzes the incoming information, and passes the results along to an Elasticsearch cluster for indexing and storage.

An Elasticsearch cluster is a collection of one or more nodes that communicate with each other to perform read and write operations on indexes. Small Wazuh deployments, which do not require processing large amounts of data, can easily be handled by a single-node cluster. Multi-node clusters are recommended when there is a large number of monitored endpoints, when a large volume of data is anticipated, or when high availability is required.

For production environments it is recommended to deploy the Wazuh server and Elasticsearch to different hosts. In this scenario, Filebeat is used to securely forward Wazuh alerts and/or archived events to the Elasticsearch cluster (single-node or multi-node) using TLS encryption.

The diagram below represents a Wazuh deployment architecture. It shows the solution components and how the [Wazuh servers](https://documentation.wazuh.com/current/getting-started/components/wazuh_server.html" \l "wazuh-server) and [Elasticsearch](https://documentation.wazuh.com/current/getting-started/components/elastic_stack.html#elastic-stack) can be configured as a cluster, providing load balancing and high-availability.

[](https://documentation.wazuh.com/current/_images/deployment.png)

## **Wazuh agent - Wazuh server communication**[**¶**](https://documentation.wazuh.com/current/getting-started/architecture.html#wazuh-agent-wazuh-server-communication)

[Wazuh agent](https://documentation.wazuh.com/current/getting-started/components/wazuh_agent.html#wazuh-agent) continuously sends events to the [Wazuh server](https://documentation.wazuh.com/current/getting-started/components/wazuh_server.html" \l "wazuh-server) for analysis and threat detection. In order to start shipping them, the agent establishes a connection with the server service for agents connection, which listens on port 1514 (this is configurable). The Wazuh server then decodes and rule-checks the received events, utilizing the analysis engine. Events that trip a rule are augmented with alert data such as rule id and rule name. Events can be spooled to one or both of the following files, depending on whether or not a rule is tripped:

* The file /var/ossec/logs/archives/archives.json contains all events whether they tripped a rule or not.
* The file /var/ossec/logs/alerts/alerts.json contains only events that tripped a rule with high enough priority (the threshold is configurable).

The Wazuh messages protocol uses AES encryption by default, with 128 bits per block and 256-bit keys (Blowfish encryption is also optional).

Note

Read the [Benefits of using AES in Wazuh communications](https://wazuh.com/blog/benefits-of-using-aes-in-our-communications) document for more information.

## **Wazuh server - Elastic Stack communication**[**¶**](https://documentation.wazuh.com/current/getting-started/architecture.html#wazuh-server-elastic-stack-communication)

Wazuh server uses Filebeat to send alert and event data to the Elasticsearch server, using TLS encryption. Filebeat reads the Wazuh server output data and sends it to Elasticsearch (by default listening on port 9200/TCP). Once the data is indexed by Elasticsearch, Kibana is used to mine and visualize the information.

The Wazuh web user interface runs inside Kibana, as a plugin. It queries the Wazuh RESTful API (by default listening on port 55000/TCP on the Wazuh server) in order to display configuration and status-related information of the [Wazuh server](https://documentation.wazuh.com/current/getting-started/components/wazuh_server.html" \l "wazuh-server) and [agents](https://documentation.wazuh.com/current/getting-started/components/wazuh_agent.html#wazuh-agent). It can also modify, through API calls, agents or server configuration settings when desired. This communication is encrypted with TLS and authenticated with username and password.

## **Required ports**[**¶**](https://documentation.wazuh.com/current/getting-started/architecture.html#required-ports)

For the communication of Wazuh components several services are used. Below is the list of default ports used by these services. Users can modify these port numbers when necessary.

| **Component** | **Software** | **Port** | **Protocol** | **Purpose** |
| --- | --- | --- | --- | --- |
| Wazuh server | Wazuh manager | 1514 | TCP (default) | Agents connection service |
| 1514 | UDP | Agents connection service |
| 1515 | TCP | Agents registration service |
| 1516 | TCP | Wazuh cluster daemon |
| 514 | UDP (default) | Wazuh syslog collector (disabled by default) |
| 514 | TCP | Wazuh syslog collector (disabled by default) |
| Wazuh API | 55000 | TCP | Wazuh RESTful API |
| Elastic Stack | Elasticsearch | 9200 | TCP | Elasticsearch RESTful API |
| 9300-9400 | TCP | Elasticsearch cluster communication |
| Kibana | 5601 | TCP | Kibana web interface |

## **Archival data storage**[**¶**](https://documentation.wazuh.com/current/getting-started/architecture.html#archival-data-storage)

Both alerts and non-alert events are stored in files on the Wazuh server, in addition to being sent to Elasticsearch. These files can be written in JSON format (.json) and/or in plain text format (.log - no decoded fields but more compact). These files are daily compressed and signed using MD5, SHA1, and SHA256 checksums. The directory and filename structure is as follows:

root@wazuh-manager:/var/ossec/logs/archives/2020/Jan# ls -l

total 176

-rw-r----- 1 ossec ossec 234350 Jan 2 00:00 ossec-archive-01.json.gz

-rw-r----- 1 ossec ossec 350 Jan 2 00:00 ossec-archive-01.json.sum

-rw-r----- 1 ossec ossec 176221 Jan 2 00:00 ossec-archive-01.log.gz

-rw-r----- 1 ossec ossec 346 Jan 2 00:00 ossec-archive-01.log.sum

-rw-r----- 1 ossec ossec 224320 Jan 2 00:00 ossec-archive-02.json.gz

-rw-r----- 1 ossec ossec 350 Jan 2 00:00 ossec-archive-02.json.sum

-rw-r----- 1 ossec ossec 151642 Jan 2 00:00 ossec-archive-02.log.gz

-rw-r----- 1 ossec ossec 346 Jan 2 00:00 ossec-archive-02.log.sum

-rw-r----- 1 ossec ossec 315251 Jan 2 00:00 ossec-archive-03.json.gz

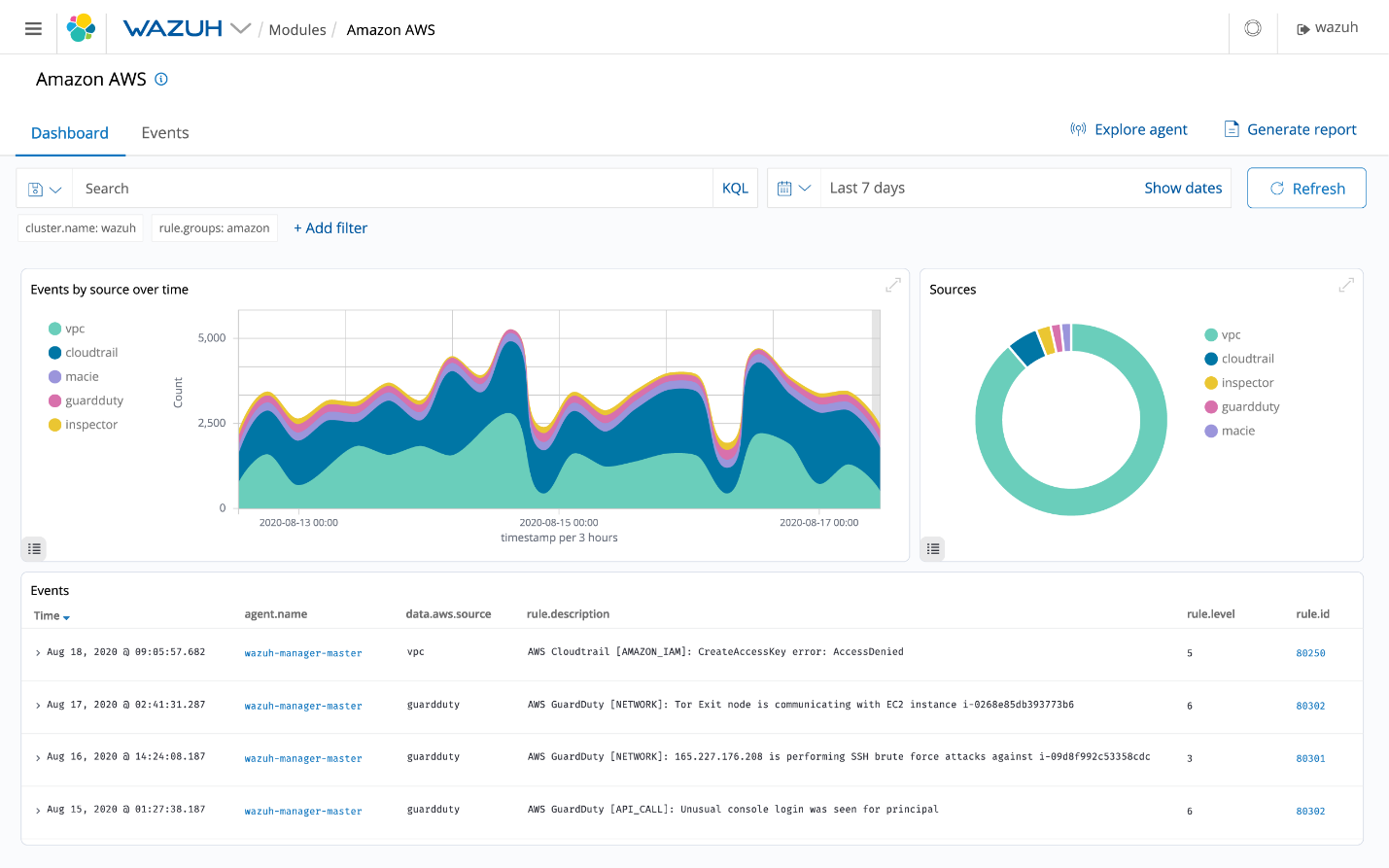
-rw-r----- 1 ossec ossec 350 Jan 2 00:00 ossec-archive-03.json.sum

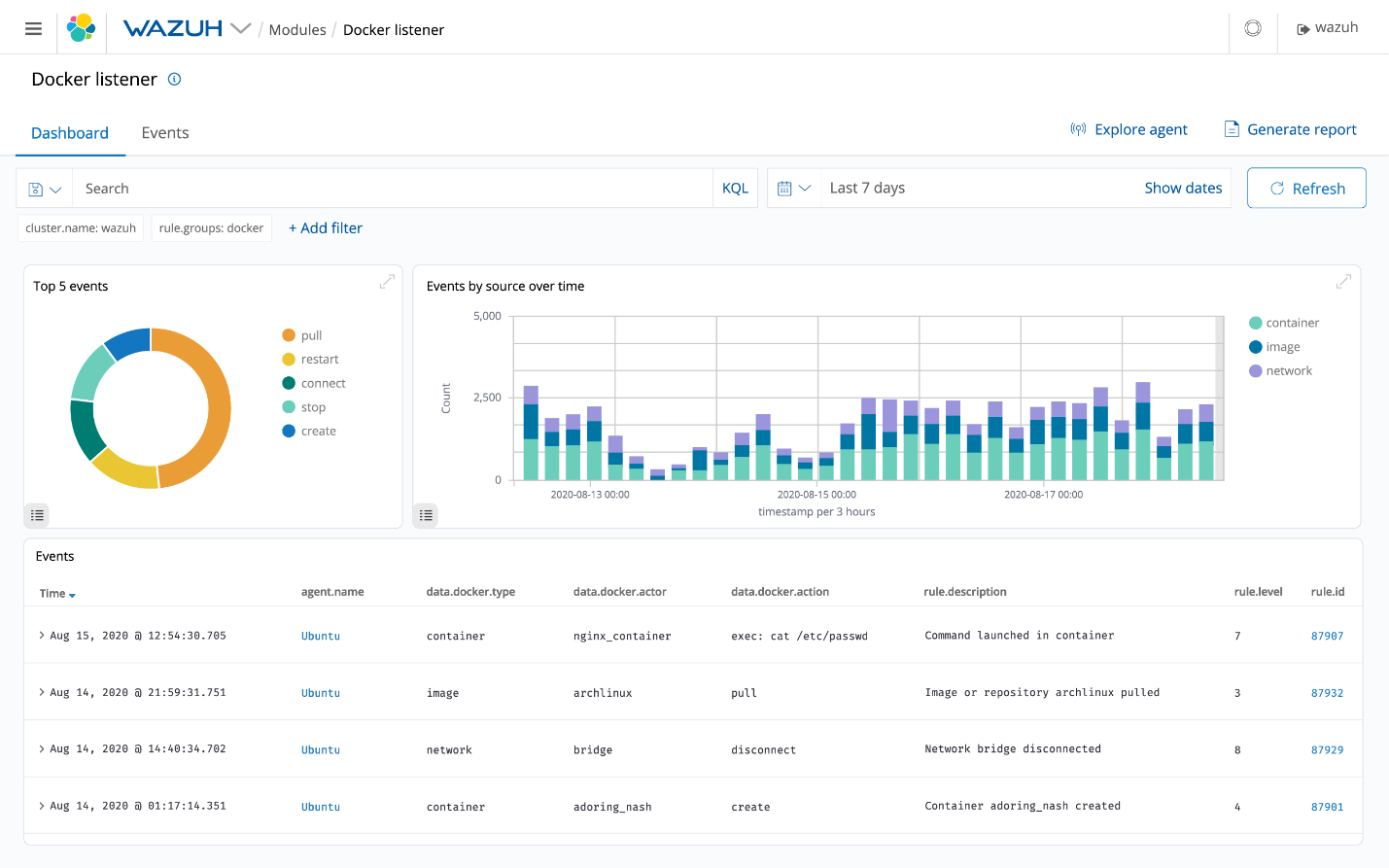
-rw-r----- 1 ossec ossec 156296 Jan 2 00:00 ossec-archive-03.log.gz

-rw-r----- 1 ossec ossec 346 Jan 2 00:00 ossec-archive-03.log.sum

Rotation and backups of archive files are recommended according to the storage capacity of the [Wazuh server](https://documentation.wazuh.com/current/getting-started/components/wazuh_server.html" \l "wazuh-server). By using cron jobs, you could easily arrange to keep only a certain time window of archive files locally on the server (e.g., last year or last three months).

On the other hand, you may choose to dispense with storing archive files at all and simply rely on Elasticsearch for archive storage, especially if you are running periodic Elasticsearch snapshot backups and/or a multi-node Elasticsearch cluster with shard replicas for high availability. You could even use a cron job to move snapshotted indexes to a final data storage server, and sign them using MD5, SHA1, and SHA256 hashing algorithms.

[](https://documentation.wazuh.com/current/_images/09-Wazuh-Cloud-Security.png)

[](https://documentation.wazuh.com/current/_images/10-Wazuh-Containers-Security.png)

# Use cases

The Wazuh platform is used to protect and monitor systems in different ways. Because of all of its capabilities, it is often used for threat prevention, detection, and response. Besides, the Wazuh platform is used to meet regulatory compliance requirements (such PCI DSS or HIPAA) and configuration standards (CIS hardening guides).

Wazuh is a popular security solution among IaaS users (eg. Amazon AWS, Azure, or Google cloud), used to monitor virtual machines and cloud instances. This is done at a system level utilizing the [Wazuh security agent](https://documentation.wazuh.com/current/getting-started/components/wazuh_agent.html" \l "wazuh-agent), and at an infrastructure level pulling data directly from the cloud provider API.

Security Analytics

Wazuh is used to collect, aggregate, index and analyze security data, helping organizations detect intrusions, threats and behavioral anomalies.

As cyber threats are becoming more sophisticated, real-time monitoring and security analysis are needed for fast threat detection and remediation. That is why our light-weight agent provides the necessary monitoring and response capabilities, while our server component provides the security intelligence and performs data analysis.

Intrusion Detection

Wazuh agents scan the monitored systems looking for malware, rootkits and suspicious anomalies. They can detect hidden files, cloaked processes or unregistered network listeners, as well as inconsistencies in system call responses.

In addition to agent capabilities, the server component uses a signature-based approach to intrusion detection, using its regular expression engine to analyze collected log data and look for indicators of compromise.

Log Data Analysis

Wazuh agents read operating system and application logs, and securely forward them to a central manager for rule-based analysis and storage.

Wazuh rules help the user to notice application or system errors, misconfigurations, attempted and/or successful malicious activities, policy violations, and other security and operational issues.

File Integrity Monitoring

Wazuh monitors the file system, identifying changes in content, permissions, ownership and attributes of files that need attention. It also natively identifies users and applications used to create or modify files.

File integrity monitoring capabilities can be used in combination with threat intelligence to identify threats or compromised hosts. In addition, several regulatory compliance standards, such as PCI DSS, require it.

Vulnerability Detection

Wazuh agents pull software inventory data and send this information to the server, where it is correlated with continuously updated CVE (Common Vulnerabilities and Exposure) databases, in order to identify well-known vulnerable software.

Automated vulnerability assessment helps the user to identify the weak spots of their critical assets and take action before being exploited by attackers.

Configuration Assessment

Wazuh monitors system and application configuration settings to ensure they are compliant with your security policies, standards and/or hardening guides. Agents perform periodic scans to detect applications that are known to be vulnerable, unpatched, or insecurely configured.

Additionally, configuration checks can be customized, tailoring them to properly align with your organization. Alerts include recommendations for better configuration, references and mapping with regulatory compliance.

Incident Response

Wazuh provides out-of-the-box active responses to perform various countermeasures to address active threats, such as blocking access to a system from the threat source when certain criteria are met.

In addition, Wazuh can be used to remotely run commands or system queries, identifying indicators of compromise (IOCs) and helping perform other live forensics or incident response tasks.

Regulatory Compliance

Wazuh provides some of the necessary security controls to become compliant with industry standards and regulations. These features, combined with its scalability and multi-platform support help organizations meet technical compliance requirements.

Wazuh is widely used by payment processing companies and financial institutions to meet PCI DSS (Payment Card Industry Data Security Standard) requirements. Its web user interface provides reports and dashboards that can help with this and other regulations such as GDPR, NIST 800-53, GPG13, TSC SOC2, and HIPAA.

Cloud Security Monitoring

Wazuh helps monitor cloud infrastructure at an API level, using integration modules that are able to pull security data from well known cloud providers like Amazon AWS, Azure, or Google Cloud. In addition, Wazuh provides rules to assess the configuration of your cloud environment, easily spotting weaknesses.

Furthermore, Wazuh light-weight and multi-platform agents are commonly used to monitor cloud environments at the instance level.

Containers Security

Wazuh provides security visibility into hosts and Docker containers, monitoring their behavior and detecting threats, vulnerabilities, and anomalies. The Wazuh agent has native integration with the Docker engine that allows users to monitor images, volumes, network configurations, and running containers.

Wazuh continuously collects and analyzes detailed runtime information. For example, alerting for containers running in privileged mode, vulnerable applications, a shell running in a container, changes to persistent volumes or images, and other possible threats.

# Installation guide[¶](https://documentation.wazuh.com/current/installation-guide/index.html#id1)

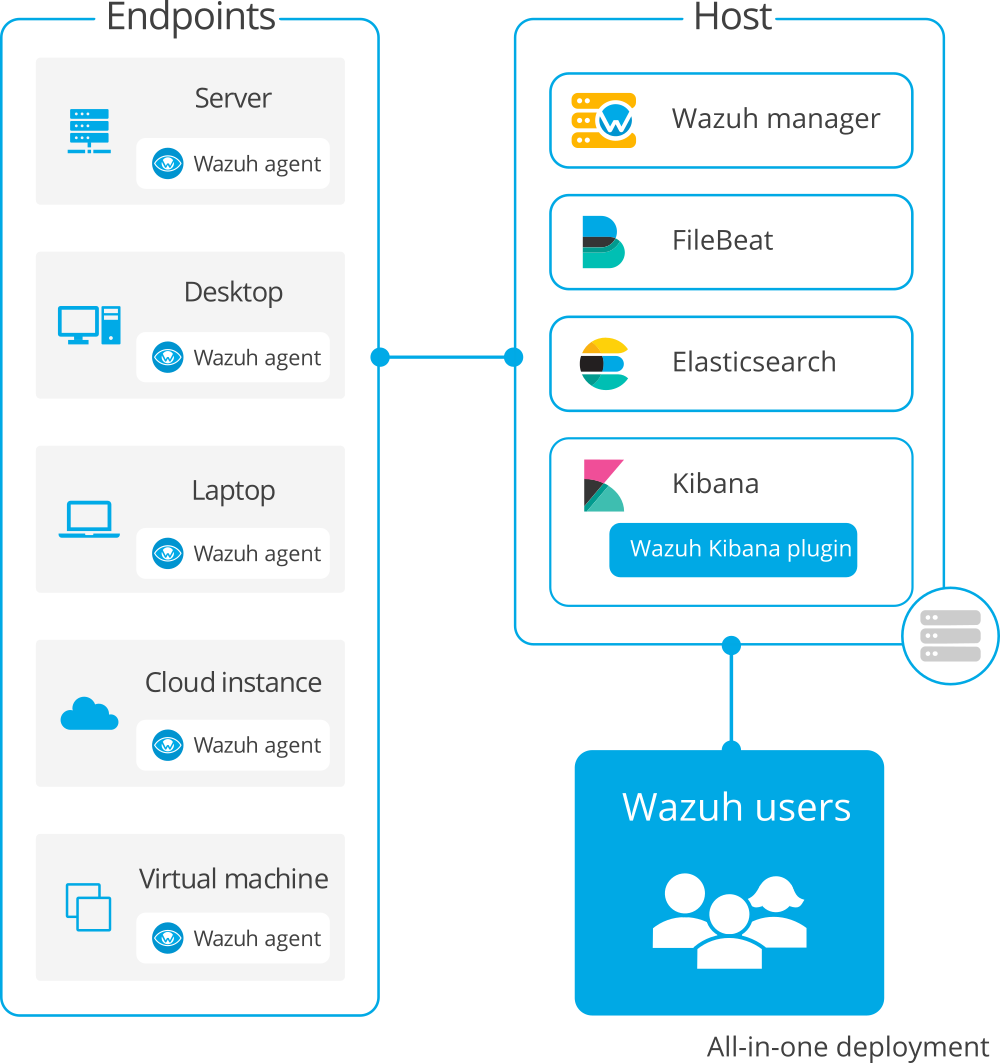
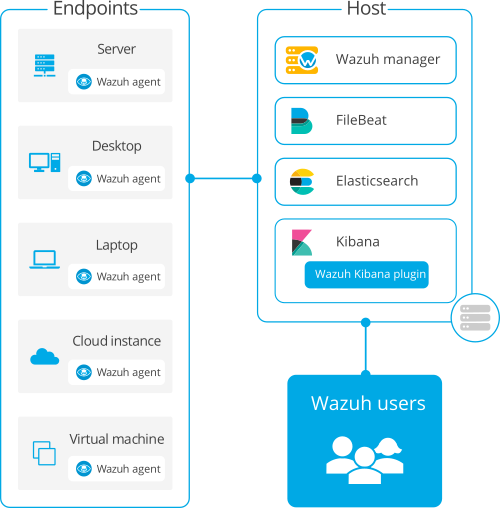
Install Wazuh and its multiple components by following the installation workflow that best suits your needs. To learn more about each component and its capabilities, check the [Components](https://documentation.wazuh.com/current/getting-started/components/index.html#components) section.

## **Wazuh server installation**[**¶**](https://documentation.wazuh.com/current/installation-guide/index.html#wazuh-server-installation)

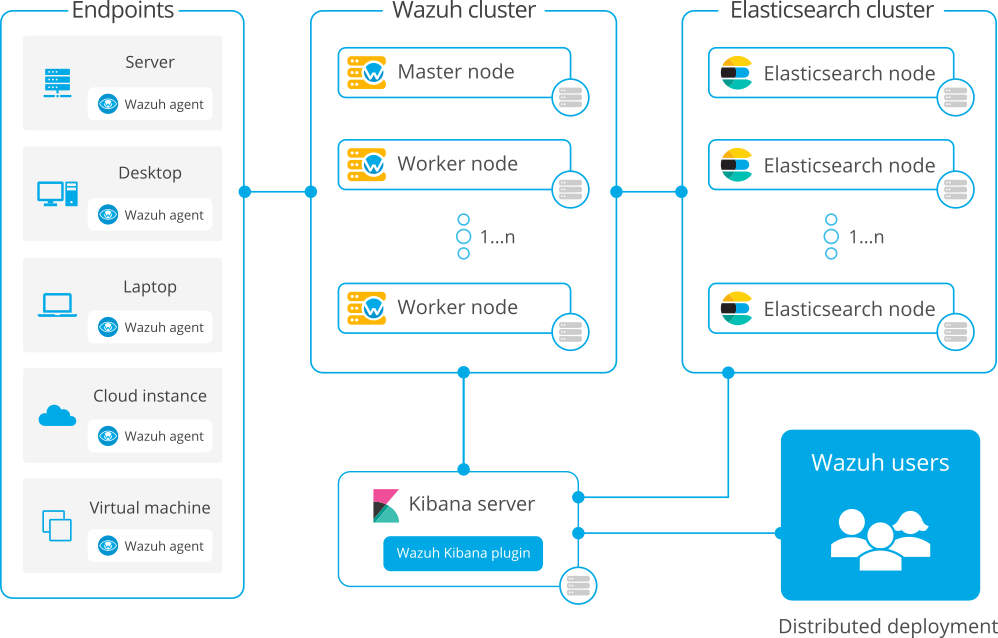
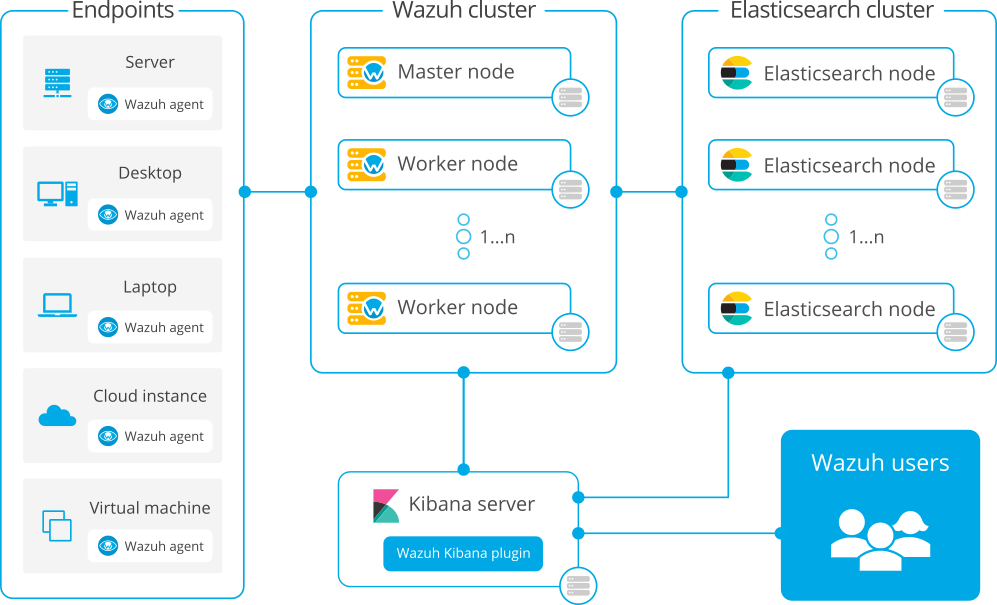
Wazuh allows you to adapt the deployment process according to your enterprise needs. You can choose between two different alternatives for deploying a Wazuh installation:

* [All-in-one](https://documentation.wazuh.com/current/installation-guide/open-distro/all-in-one-deployment/index.html#all-in-one-index): All the Wazuh components are installed on the same host. This type of deployment is appropriate for testing and small working environments. If you want to test Wazuh, you can download our ready-to-use [OVA](https://documentation.wazuh.com/current/virtual-machine/virtual-machine.html#virtual-machine).
* [Distributed](https://documentation.wazuh.com/current/installation-guide/open-distro/distributed-deployment/index.html#distributed-index): Each component is installed on a separate host as a single-node or multi-node cluster. This type of deployment allows high availability and scalability of the product and is convenient for large working environments.

### **All-in-one deployment**

[](https://documentation.wazuh.com/current/_images/all_in_one1.png)

### **Distributed deployment**

[](https://documentation.wazuh.com/current/_images/distributed1.png)

Wazuh can also be installed with commercial options like Elastic Stack basic license or Splunk. To learn more about these options and other installation alternatives, see the [more installation alternatives](https://documentation.wazuh.com/current/installation-guide/more-installation-alternatives/index.html#more-installation-alternatives) section.

Note

Wazuh also offers the [Wazuh Cloud](https://wazuh.com/cloud/), where all components are hosted on our PCI-DSS certified SaaS solution and maintained by our team. With the Wazuh cloud, no dedicated hardware is required and everything is ready to use. This service offers a highly flexible infrastructure to match your enterprise needs.

# Requirements[¶](https://documentation.wazuh.com/current/installation-guide/requirements.html#requirements)

Check the supported operating systems and the recommended hardware requirements for the different types of deployments of the Wazuh installation. Make sure that your system environment meets all requirements and that you have root user privileges.

## **Supported operating systems**[**¶**](https://documentation.wazuh.com/current/installation-guide/requirements.html#supported-operating-systems)

The Wazuh server and Elastic Stack components can be installed in the following Linux operating systems:

* Amazon Linux 1 and 2
* CentOS 6 or later
* Debian 7 or later
* Fedora 31 or later
* Oracle Linux 6 or later
* Red Hat Enterprise Linux 6 or later
* Ubuntu 12 or later

## **All-in-one deployment**[**¶**](https://documentation.wazuh.com/current/installation-guide/requirements.html#all-in-one-deployment)

In an all-in-one deployment, both the Wazuh server and Elastic Stack are installed on the same host. This type of deployment is suitable for testing and small production environments. A typical use case for this kind of environment supports around 100 agents.

The minimum requirements for this type of deployment are 4 GB of RAM and 2 CPU cores, and the recommended are 16 GB of RAM and 8 CPU cores. A 64-bit operating system is required.

Disk space requirements depend on the alerts per second (APS) generated. The expected APS vary significantly depending on the amount and type of monitored endpoints. Estimated storage per agent needed for 90 days of alerts depending on the type of monitored endpoint:

| **Monitored endpoints** | **APS** | **Storage (GB/90 days)** |
| --- | --- | --- |
| Servers | 0.25 | 3.8 |
| Workstations | 0.1 | 1.5 |
| Network devices | 0.5 | 7.6 |

For example, for an environment with 80 workstations, 10 servers, and 10 network devices, the storage needed for 90 days of alerts is 236 GB approximately.

## **Distributed deployment**[**¶**](https://documentation.wazuh.com/current/installation-guide/requirements.html#distributed-deployment)

In a distributed deployment, both the Wazuh server and Elastic Stack are installed on separate hosts. This configuration is recommended for production environments as it provides the high availability and scalability of the services.

The Wazuh server and Elastic Stack can each be installed as a single-node or as a multi-node cluster. Kibana can either be installed on the same node as Elasticsearch or on a dedicated host. Hardware recommendations for each node:

|  | **Minimum** | | **Recommended** | |
| --- | --- | --- | --- | --- |
| **Component** | **RAM (GB)** | **CPU (cores)** | **RAM (GB)** | **CPU (cores)** |
| Wazuh server | 2 | 2 | 8 | 4 |
| Elastic Stack | 4 | 2 | 16 | 8 |

A 64-bit operating system is necessary.

Regarding disk space requirements, the amount of data depends on the alerts per second (APS) generated. Estimated disk space per agent needed to store 90 days of alerts on a Wazuh server and on an Elasticsearch server, depending on the type of monitored endpoints:

| **Monitored endpoints** | **APS** | **Storage in Wazuh Manager**  **(GB/90 days)** | **Storage in Elasticsearch**  **(GB/90 days)** |
| --- | --- | --- | --- |
| Servers | 0.25 | 0.1 | 3.7 |
| Workstations | 0.1 | 0.04 | 1.5 |
| Network devices | 0.5 | 0.2 | 7.4 |

For example, for an environment with 80 workstations, 10 servers, and 10 network devices, the storage needed for 90 days of alerts is 230 GB on the Elasticsearch server and 6 GB on the Wazuh server approximately.

## **Scaling**[**¶**](https://documentation.wazuh.com/current/installation-guide/requirements.html#scaling)

To determine if a Wazuh server requires more resources, the following files can be monitored:

* /var/ossec/var/run/ossec-analysisd.state: the variable events\_dropped indicates whether events are being dropped due to lack of resources.
* /var/ossec/var/run/ossec-remoted.state: the variable discarded\_count indicates if messages from the agents were discarded.

These two variables should be zero if the environment is working properly. If it is not the case, additional nodes can be added to the cluster.

To monitor if the Elastic Stack environment is working properly, tools such as the performance analyzer are available.

In case that scaling is needed, a distributed deployment of [Wazuh with Elastic Stack](https://documentation.wazuh.com/current/installation-guide/open-distro/distributed-deployment/index.html" \l "distributed-index) is recommended.

# Linux[¶](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#linux)

This document will guide you to install or deploy the Wazuh agent.

Note

To execute the commands described below, root privileges are required.

## **Adding the Wazuh repository**[**¶**](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#adding-the-wazuh-repository)

Yum

APT

ZYpp

1. Import the GPG key:
2. # rpm --import https://packages.wazuh.com/key/GPG-KEY-WAZUH
3. Add the repository:
4. # cat > /etc/yum.repos.d/wazuh.repo << EOF
5. [wazuh]
6. gpgcheck=1
7. gpgkey=https://packages.wazuh.com/key/GPG-KEY-WAZUH
8. enabled=1
9. name=EL-$releasever - Wazuh
10. baseurl=https://packages.wazuh.com/4.x/yum/
11. protect=1
12. EOF

## **Installing Wazuh agent**[**¶**](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#installing-wazuh-agent)

1. Install the Wazuh agent on your terminal. You can choose between installation or deployment:
   1. Installation:

Yum

APT

ZYpp

# yum install wazuh-agent

Once the agent is installed, the next step is to register it and configure it to communicate with the manager. For more information on this process, visit the [user manual](https://documentation.wazuh.com/current/user-manual/registering/index.html#register-agents) section.

* 1. Deployment:

The registration and configuration of the agent can be automated using variables. It is necessary to define, at least, the variable WAZUH\_MANAGER. The agent will use this value to register and this will be the assigned manager for forwarding events.

Yum

APT

ZYpp

# WAZUH\_MANAGER**=**"10.0.0.2" yum install wazuh-agent

See the following document for additional deployment options: [deployment variables](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/deployment_variables/linux/deployment_variables_apt.html#deployment-variables-apt).

1. Enable the service

Systemd

SysV Init

# systemctl daemon-reload

# systemctl enable wazuh-agent

# systemctl start wazuh-agent

**(Optional)** Disable Wazuh updates:

The version of the Wazuh manager is recommended to be greater than or equal to that of the Wazuh agents. Therefore, we recommend disabling the Wazuh repository to prevent accidental upgrades. To do so, use the following command:

Yum

APT

ZYpp

# sed -i "s/^enabled=1/enabled=0/" /etc/yum.repos.d/wazuh.repo

Visit our [packages list](https://documentation.wazuh.com/current/installation-guide/packages-list.html#packages) section to download the Wazuh agent package directly or to check the compatible versions.

## **Uninstall**[**¶**](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#uninstall)

To uninstall the agent:

Yum

APT

ZYpp

# yum remove wazuh-agent

Some files are marked as configuration files. Due to this designation, the package manager does not remove these files from the filesystem. The complete file deletion action is the responsibility of the user and can be done by deleting the folder /var/ossec.

# Linux[¶](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#linux)

This document will guide you to install or deploy the Wazuh agent.

Note

To execute the commands described below, root privileges are required.

## **Adding the Wazuh repository**[**¶**](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#adding-the-wazuh-repository)

Yum

APT

ZYpp

1. Install the GPG key:
2. # curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | apt-key add -
3. Add the repository:
4. # echo "deb https://packages.wazuh.com/4.x/apt/ stable main" | tee -a /etc/apt/sources.list.d/wazuh.list
5. Update the package information:
6. # apt-get update

## **Installing Wazuh agent**[**¶**](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#installing-wazuh-agent)

1. Install the Wazuh agent on your terminal. You can choose between installation or deployment:
   1. Installation:

Yum

APT

ZYpp

# apt-get install wazuh-agent

Once the agent is installed, the next step is to register it and configure it to communicate with the manager. For more information on this process, visit the [user manual](https://documentation.wazuh.com/current/user-manual/registering/index.html#register-agents) section.

* 1. Deployment:

The registration and configuration of the agent can be automated using variables. It is necessary to define, at least, the variable WAZUH\_MANAGER. The agent will use this value to register and this will be the assigned manager for forwarding events.

Yum

APT

ZYpp

# WAZUH\_MANAGER**=**"10.0.0.2" apt-get install wazuh-agent

See the following document for additional deployment options: [deployment variables](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/deployment_variables/linux/deployment_variables_apt.html#deployment-variables-apt).

1. Enable the service

Systemd

SysV Init

# systemctl daemon-reload

# systemctl enable wazuh-agent

# systemctl start wazuh-agent

**(Optional)** Disable Wazuh updates:

The version of the Wazuh manager is recommended to be greater than or equal to that of the Wazuh agents. Therefore, we recommend disabling the Wazuh repository to prevent accidental upgrades. To do so, use the following command:

Yum

APT

ZYpp

# sed -i "s/^deb/#deb/" /etc/apt/sources.list.d/wazuh.list

# apt-get update

Alternatively, you can set the package state to hold, which will stop updates (although you can still upgrade it manually using apt-get install).

# echo "wazuh-agent hold" | dpkg --set-selections

Visit our [packages list](https://documentation.wazuh.com/current/installation-guide/packages-list.html#packages) section to download the Wazuh agent package directly or to check the compatible versions.

## **Uninstall**[**¶**](https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh_agent_package_linux.html#uninstall)

To uninstall the agent:

Yum

APT

ZYpp

# apt-get remove wazuh-agent

There are certain files marked as configuration files. Due to this designation, the package manager does not remove those files from the filesystem. A complete file removal can be done using the following command:

# apt-get remove --purge wazuh-agent

# Docker[¶](https://documentation.wazuh.com/current/docker/index.html#docker)

[Docker](https://www.docker.com/) is an open-source project that automates the deployment of different applications inside software containers. Docker containers wrap up a piece of software in a complete filesystem that contains everything it needs to run like: code, system tools, libraries, etc. This process guarantees that the system will always run the same, regardless the environment it is running.

We have created our own fork based on [“deviantony” dockerfiles](https://github.com/deviantony/docker-elk) and [“xetus-oss” dockerfiles](https://github.com/xetus-oss/docker-ossec-server). Thank you, Anthony Lapenna, for your contribution to the community. If you want to contribute to the Wazuh fork, please go to our [Docker repository](https://github.com/wazuh/wazuh-docker).

The images we created are in the [Docker hub](https://hub.docker.com/). You can install Wazuh with a single-host architecture using a set of Docker images that contains [Wazuh Manager](https://github.com/wazuh/wazuh), [Filebeat](https://www.elastic.co/products/beats/filebeat), [Elasticsearch](https://registry.hub.docker.com/_/elasticsearch/), [Kibana](https://registry.hub.docker.com/_/kibana/) and optionally [Nginx](https://hub.docker.com/_/nginx/). [Open Distro for Elasticsearch](https://opendistro.github.io/for-elasticsearch/) is fully supported as well.

This section will show you the process of installing and configuring the Wazuh deployment on Docker:

* [Docker installation](https://documentation.wazuh.com/current/docker/docker-installation.html)
* [Wazuh Docker deployment](https://documentation.wazuh.com/current/docker/wazuh-container.html)
* [Wazuh Docker utilities](https://documentation.wazuh.com/current/docker/container-usage.html)
* [Upgrade Guide (3.x to 4.0)](https://documentation.wazuh.com/current/docker/upgrade-guide.html)