Wazuh SIEM

- Install Wazuh Server and Add Agents

Install Wazuh Server and Add Agents

Wazuh is an open source set of tools that we can put to work for us and our clients, helping us make sure that we have as many attack vectors buttoned down as possible, and all of our security systems up to date.

Recall that we previously installed Zabbix, and agents on our servers. This is a similar dynamic to what's needed for Wazuh. It has a server portion, and a client agent that goes on each machine. We can, of course, put our Ansible skills to use, so this will be a great way to expand our skills yet again.

Installing Wazuh Server

```
curl -sO https://packages.wazuh.com/4.7/wazuh-install.sh && sudo bash ./wazuh-install.sh -a
```

The above command will start the Wazuh installation for the server. When complete it will provide an initial user and login password for the system.

> **Make sure to make note of the initial username and password, then you can login at the address (IP / FQDN) and port provided.**

If you happen to miss the username and password that is created for you, never fear. We can still find it.

Finding the Initial User and Password if needed

We need to find the `wazuh-install-files.tar`, and inside of that find the `wazuh-passwords.txt` file. Here we can find our default password as initially setup by the system.

First, let's untar the `wazuh-install-files.tar` file so we can access the folder.

```
sudo tar -xf  wazuh-install-files.tar
```

Next, we'll move into the newly created directory.

```
cd wazuh-install-files
```

And now we'll `cat` out the passwords file.

`sudo cat wazuh-passwords.txt`

You should get something like this:

```
# Admin user for the web user interface and Wazuh indexer. Use this user to log in to Wazuh dashboard
  indexer_username: 'admin'
  indexer_password: 'a-really-secret-password-that-I-removed'

# Wazuh dashboard user for establishing the connection with Wazuh indexer
  indexer_username: 'kibanaserver'
  indexer_password: 'a-really-secret-password-that-I-removed'

# Regular Dashboard user, only has read permissions to all indices and all permissions on the .kibana index
  indexer_username: 'kibanaro'
  indexer_password: 'a-really-secret-password-that-I-removed'

# Filebeat user for CRUD operations on Wazuh indices
  indexer_username: 'logstash'
  indexer_password: 'a-really-secret-password-that-I-removed'

# User with READ access to all indices
  indexer_username: 'readall'
  indexer_password: 'a-really-secret-password-that-I-removed'

# User with permissions to perform snapshot and restore operations
  indexer_username: 'snapshotrestore'
  indexer_password: 'a-really-secret-password-that-I-removed'

# Password for wazuh API user
  api_username: 'wazuh'
  api_password: 'a-really-secret-password-that-I-removed'

# Password for wazuh-wui API user
  api_username: 'wazuh-wui'
  api_password: 'a-really-secret-password-that-I-removed'
```

Grab the 'admin' password, and go to your page to login.

After logging in for the first time, you should update the username and password for the system. Make sure to use a strong password.

Create an Admin User

In Wasuh, let's go to the menu icon in the upper left of the screen.

1. Find the option for 'Open Search Plugins / Security`.
2. Next, click on 'Internal Users'
3. On the Internal Users page, click on 'Create Internal User'
4. Enter a username, password, and confirm the password, then click the 'Create' buton to make the new user.

> NOTE: Wazuh maps a user to a role. We now need to map our user to an admin role.

1. Next, click on 'Roles'
2. Search for the 'all_access' role. Select the role by clicking on it's name in the list below the search.
3. Once in the details page, you'll duplicate the role by clicking the 'Duplicate Role' button in the upper right.
4. On the 'Duplicate Role' screen, give the role a new name. I called mine 'all_access_brian' for my user.
5. Now scroll to the bottom and click 'Create'.
6. On the new role details page, click the 'Mapped Users' tab, then click the 'Map Users' button.
7. Select your previously created user from the drop-down, and then click 'Map'.

Now we'll create a role mapping to map our user to the role we just created.

1. Click 'Role Mapping'
2. Give the role a name, I called mine 'brian_admin'
3. Next, select the role of 'administrator'.
4. Now select your previously created user, and click 'Create Role Mapping'.

Finally, we need to edit a file on the server where wazuh is running. It will be easier to become root for a minute, so on a Ubuntu server, do

```
sudo su
```

Enter your super user password, and keep in mind that you are now "root", so be careful what you do until you exit back out of root.

Now let's do

```
sudo nano /usr/share/wazuh-dashboard/data/wazuh/config/wazuh.yml
```

In this file, use CTRL + W to open search, and move to the uncommented entry for `run_as`. This was the second instance of that phrase in mine, so I had to use CTRL + W twice.

Once there, change it from `false` to `true`, then save with CTRL + O, press Enter to confirm, and exit nano with CTRL + X.

Now exit your su / root session by entering `exit` into the terminal, and make sure you are back at your normal user's prompt.

Finally, we need to restart the wazuh-dashboard service.

```
sudo systemctl restart wazuh-dashboard
```

Now, you can exit as admin, and login with your new user.

You can now start digging around in the system a bit, and looking at the User Interface. You'll notice that there is a lot going on, and making sure you understand where to update settings, find dashboards and information is important.

Install the Wazuh Agent on a Machine

In the upper left corner of the Wazuh Dashboard, click the hamburger menu (3 lines stacked) to expand the left side panel. Here you'll find all of the things you can do in the Wazuh Server. It's a bit unintuitive right now, in my opinion, but it's ever evolving, so keep an eye out for positive changes over time.

In the left menu scroll down to the 'Server Management' section, and click to expand it. Now click on the 'Endpoint Summary' option in that menu, and you'll be taken to where your summary of the agents you are running will be. Initially, this page is empty, and you'll have a giant 'Add an Agent' option in the middle. Click that, then we' need to fill out a short form. The contents of the form, will help the server generate an installation script for the endpoint you are installing the agent on.

1. First, choose the OS where you'll be running the agent. In my case, I really only run Linux machines, so I chose the DEB amd64 version.
2. Next, enter the server address. This is either the IP address or FQDN of the server. In my case I use a LAN IP, or a netbird VPN IP to identify the server. Keep in mind this is asking for the **server** address, and not the IP address of the client machine.
3. Now we'll give this agent a name to help us identify the machine we are adding to our Wazuh monitoring.
4. If you've created groups for your client machines, use the drop-down to select the group you want the machine to be a part of.
5. Copy the script provided, and paste it on the client machine, and let it run. If all goes well the Wazuh agent will be downloaded and auto-configured for this machine. (You may be prompted to enter your super-user password during installation).
6. Once the installation completes, run the final set of commands provided to start the service on the client.

You can now return to your dashboard, and refresh to see that the newly added agent has already begun sending data. If you don't see data coming in, double check your agent settings, and try again.

You now have Wazuh Up and Running. As I said before, you can also use Ansible to auto-provision the Wazuh client to multiple machines at once. They have some great directions on getting ansible installed and setup, as well as adding the necessary roles to ansible for wazuh to function properly.

Now What?

Getting Wazuh up and running is just step 1. It's a system that will take some time for you to configure properly to not only gather the data from your machines, but to recognize serious threats, deal with them, and / or notify you and your team of any issues that are found.

Security is about being proactive. Stay ahead of the curve. But when it's necessary to be reactive, put the tools to work for you so you can react quickly.

Support My Channel and Content

Support my Channel and ongoing efforts through Patreon:

https://www.patreon.com/awesomeopensource

Buy me a Beer / Coffee:
https://paypal.me/BrianMcGonagill