Setup QGIS and Lizmap Server Guide

Table of Contents

Run the playbook

Follow these steps to run the playbook successful.

> ℹ️ *User- and Hostname*
>
> We will use `test-host` as an example host and `manager` as the user, but it is recommended to use a custom username instead.
>
> Use the `getent group sudo` command to check if the user you want to use has administrative rights.

Requirements

- Git

```
git clone https://somepi.ddns.net/gitea/gilex-dev/GisServer.git
cd GisServer
```

- Python venv

```
python3 -m venv ./.venv --system-site-packages
source ./.venv/bin/activate
pip3 install -r requirements.txt
ansible-galaxy install -r requirements.yaml
```

Adding your host

1. Choose the inventory name used by ansible (this does not have to be the actual hostname) `export NEW_ANSIBLE_INV_NAME="testing-server-1"`

2. Set hostname used by ansible `export NEW_ANSIBLE_HOST="test-host"`

3. Set username used by ansible `export NEW_ANSIBLE_USER="manager"` .

```
ssh-keygen -a 100 -t ed25519 -C "a comment" -f ~/.ssh/id_
$NEW_ANSIBLE_USER@NEW_ANSIBLE_HOST
ssh-keygen -a 100 -t ed25519 -C "another comment" -f
~/.ssh/id_publisher@$NEW_ANSIBLE_HOST
ssh-copy-id -i ~/.ssh/id_$NEW_ANSIBLE_USER@NEW_ANSIBLE_HOST $NEW_ANSIBLE_USER
@$NEW_ANSIBLE_HOST
ssh-add ~/.ssh/id_$NEW_ANSIBLE_USER@NEW_ANSIBLE_HOST # optional, to remember key-
phrase
```

4. Add the host to the inventory `nano inventory.yaml` (keep other entries)

```
testing: # or production or any other sensible group name
    hosts:
        # $NEW_ANSIBLE_INV_NAME: # or same as $NEW_ANSIBLE_HOST
            ansible_host: "{{ inventory_hostname }}" # or any address where the
host is reachable
            ansible_user: # NEW_ANSIBLE_USER
            ansible_ssh_private_key_file: #
~/.ssh/id_$NEW_ANSIBLE_USER@$NEW_ANSIBLE_HOST
```

5. Create a directory for the host `mkdir host_vars/$NEW_ANSIBLE_HOST`

6. Copy the basic config file

```
cp host_vars/variables.yaml.template host_vars/$NEW_ANSIBLE_HOST/variables.yaml
```

7. Create the vault to store passwords needed by ansible

```
ansible-vault encrypt host_vars/secrets.yaml.template --output
host_vars/$NEW_ANSIBLE_HOST/secrets.yaml`
```

8. Edit the vault

```
ansible-vault edit host_vars/$NEW_ANSIBLE_HOST/secrets.yaml
# Generate a new password for every entry (preferably A-z0-9 to avoid escape
mishaps)
```

9. Add the host config to your `~/.ssh/config` file to allow a simple ssh connection

```
Host # e.g. $NEW_ANSIBLE_HOST
    HostName # $NEW_ANSIBLE_HOST
    User # $NEW_ANSIBLE_USER
    IdentityFile # ~/.ssh/id_$NEW_ANSIBLE_USER@NEW_ANSIBLE_HOST
    IdentitiesOnly yes
```

```
    VisualHostKey yes
```

10. (optional) Clear env variables `unset NEW_ANSIBLE_HOST`, `unset NEW_ANSIBLE_USER` and `unset NEW_ANSIBLE_INV_NAME`

11. Repeat for any other hosts you want to be managed by this playbook

Run ansible (finally)

If you don't want to make any changes in `host_vars/$NEW_ANSIBLE_HOST/variables.yaml` you can now run the playbook with the following command:

```
ansible-playbook playbooks/main.yaml --ask-vault-pass
# or store the vault password in a file (e.g. `vault_password.txt`). !!!Only do this
while testing!!!
# ansible-playbook playbooks/main.yaml --vault-password-file vault_password.txt
```

See the [ansible documentation](https://docs.ansible.com/ansible/latest/inventory_guide/intro_patterns.html) for specifying which hosts to target.

Don't panic if the following tasks take long to complete the first time you run the playbook on a host:

- `Ensure valid apt cache and required tools are present`

- `Ensure qgis-server is present`