How to install Ansible Semaphore

Installation

To install Semaphore on Ubuntu Linux controller using the recommended methods, please follow the instructions below

Method

Snap Installation

sudo snap install semaphore
sudo snap stop semaphore\\
{replace ben with your own credentials
sudo semaphore user add --admin --login ben --name=ben --email=ben1996@gmail.com --password=12345
sudo snap start semaphore
sudo snap services semaphore

Package Manager

wget https://github.com/ansible-semaphore/semaphore/releases/download/v2.8.75/semaphore_2.8.75_linux_amd64.deb
sudo dpkg -i semaphore_2.8.75_linux_amd64.deb
semaphore setup
semaphore service --config=./config.json

Docker

Create a docker-compose.yml file with the following inside

```
services:
  # uncomment this section and comment out the mysql section to use postgres instead of mysql
  #postgres:
        #restart: unless-stopped
        #image: postgres:14
        #hostname: postgres
        #volumes:
        #  - semaphore-postgres:/var/lib/postgresql/data
        #environment:
        #  POSTGRES_USER: semaphore
        #  POSTGRES_PASSWORD: semaphore
        #  POSTGRES_DB: semaphore
  # if you wish to use postgres, comment the mysql service section below
  mysql:
        restart: unless-stopped
        image: mysql:8.0
        hostname: mysql
        volumes:
        - semaphore-mysql:/var/lib/mysql
        environment:
        MYSQL_RANDOM_ROOT_PASSWORD: 'yes'
        MYSQL_DATABASE: semaphore
        MYSQL_USER: semaphore
        MYSQL_PASSWORD: semaphore
  semaphore:
        restart: unless-stopped
        ports:
        - 3000:3000
        image: semaphoreui/semaphore:latest
        environment:
        SEMAPHORE_DB_USER: semaphore
        SEMAPHORE_DB_PASS: semaphore
        SEMAPHORE_DB_HOST: mysql # for postgres, change to: postgres
        SEMAPHORE_DB_PORT: 3306 # change to 5432 for postgres
        SEMAPHORE_DB_DIALECT: mysql # for postgres, change to: postgres
        SEMAPHORE_DB: semaphore
        SEMAPHORE_PLAYBOOK_PATH: /tmp/semaphore/
        SEMAPHORE_ADMIN_PASSWORD: changeme
        SEMAPHORE_ADMIN_NAME: admin
```

SEMAPHORE_ADMIN_EMAIL: admin@localhost
SEMAPHORE_ADMIN: admin
SEMAPHORE_ACCESS_KEY_ENCRYPTION: gs72mPntFATGJs9qK0pQ0rKtfidlexiMjYCH9gWKhTU=
SEMAPHORE_LDAP_ACTIVATED: 'no' # if you wish to use ldap, set to: 'yes'
SEMAPHORE_LDAP_HOST: dc01.local.example.com
SEMAPHORE_LDAP_PORT: '636'
SEMAPHORE_LDAP_NEEDTLS: 'yes'
SEMAPHORE_LDAP_DN_BIND: 'uid=bind_user,cn=users,cn=accounts,dc=local,dc=shiftsystems,dc=net'
SEMAPHORE_LDAP_PASSWORD: 'ldap_bind_account_password'
SEMAPHORE_LDAP_DN_SEARCH: 'dc=local,dc=example,dc=com'
SEMAPHORE_LDAP_SEARCH_FILTER: "(\u0026(uid=%s)(memberOf=cn=ipausers,cn=groups,cn=accounts,dc=local,dc=example,dc=com))"
depends_on:
- mysql # for postgres, change to: postgres
volumes:
semaphore-mysql: # to use postgres, switch to: semaphore-postgres

Then run
Docker-compose up

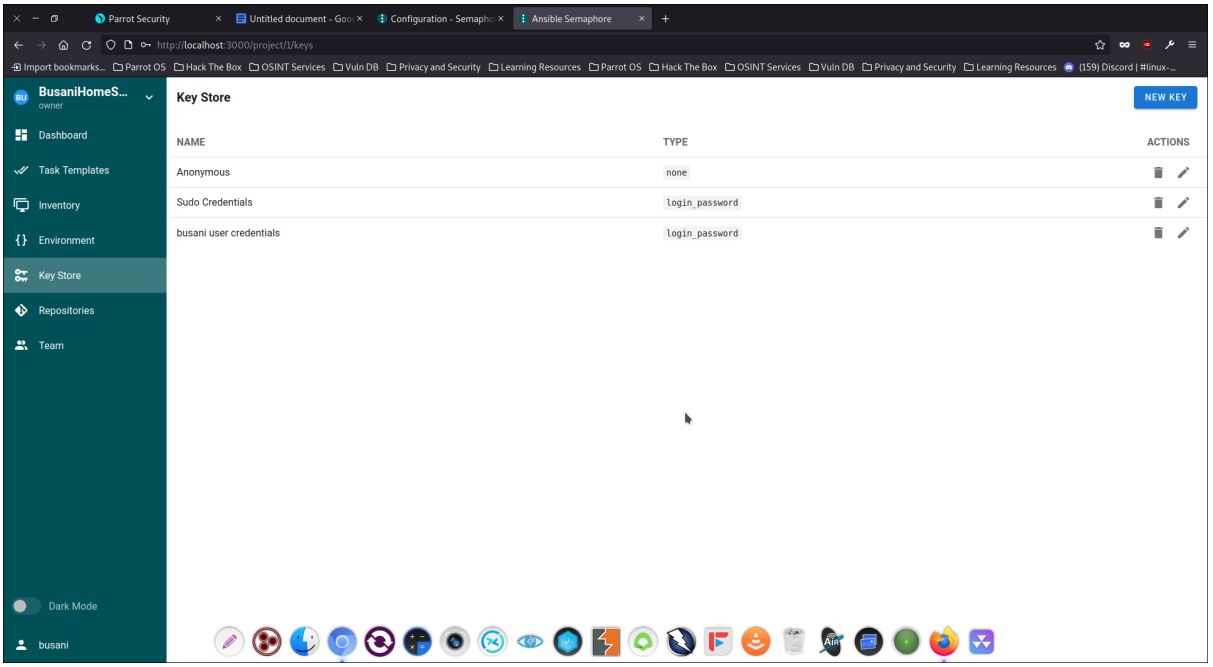
Semaphore Usage

- Navigate to your browser to https://localhost:3000

- Enter your Credentials

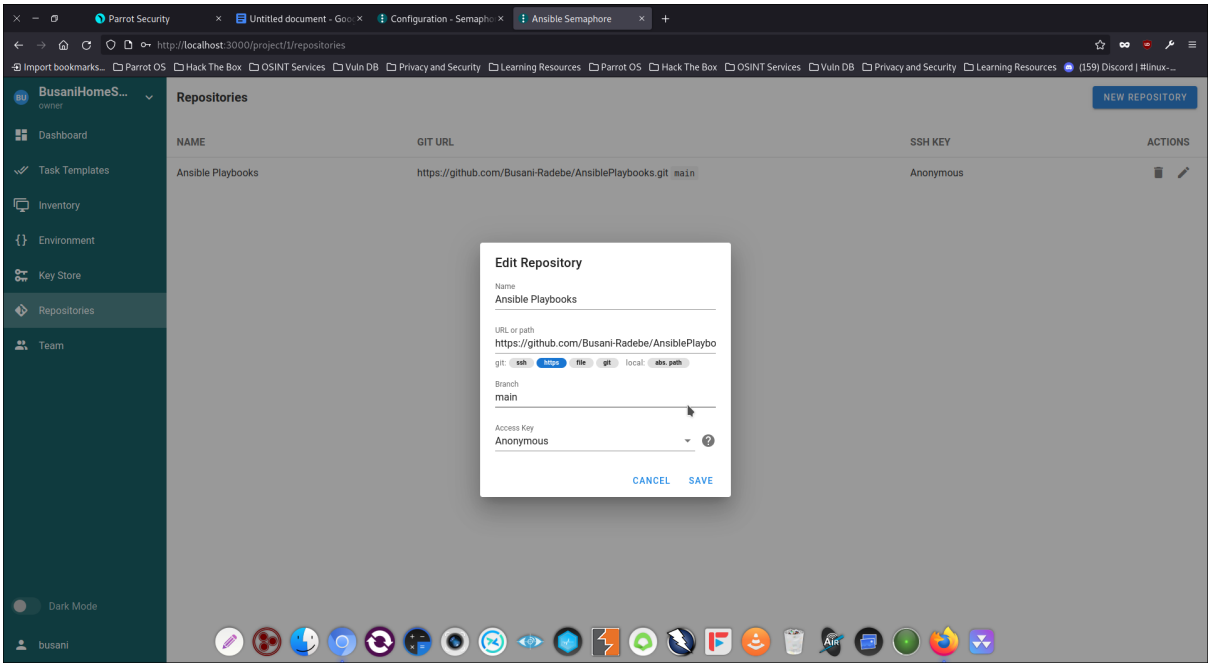
- Give your Project a Name

- Create Anonymous,Sudo Credentials ,normal user Credentials for your machines ,Credentials for your public Repository
  When prompted for type choose “none”

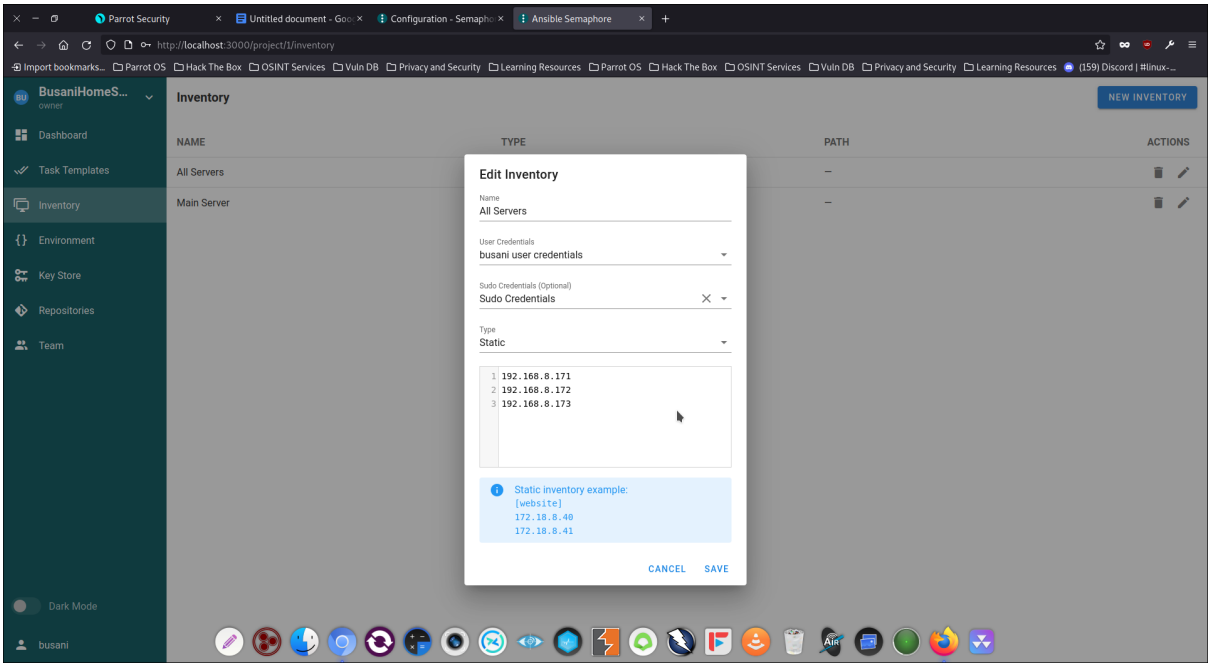


Link Github Repository

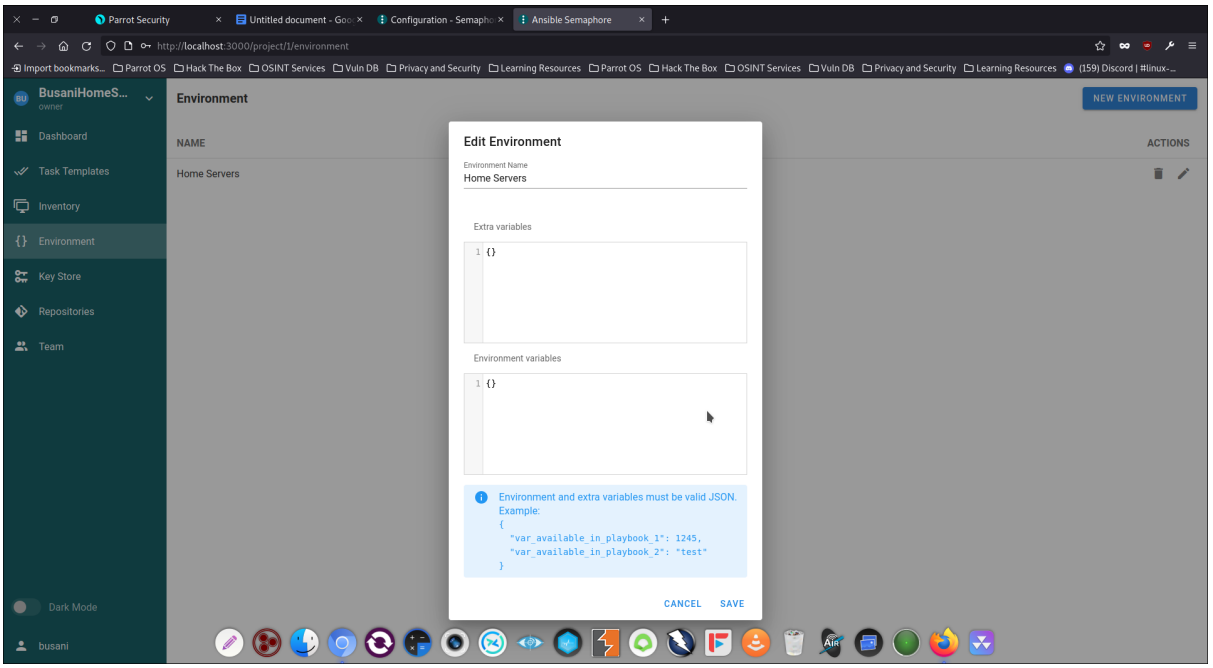
Add your Github Repository to the Repositories tab on the left hand where you will store Your playbooks



Add your machines ip addresses to your Inventory List



Add an empty Environment



Create Task Template & Playbook

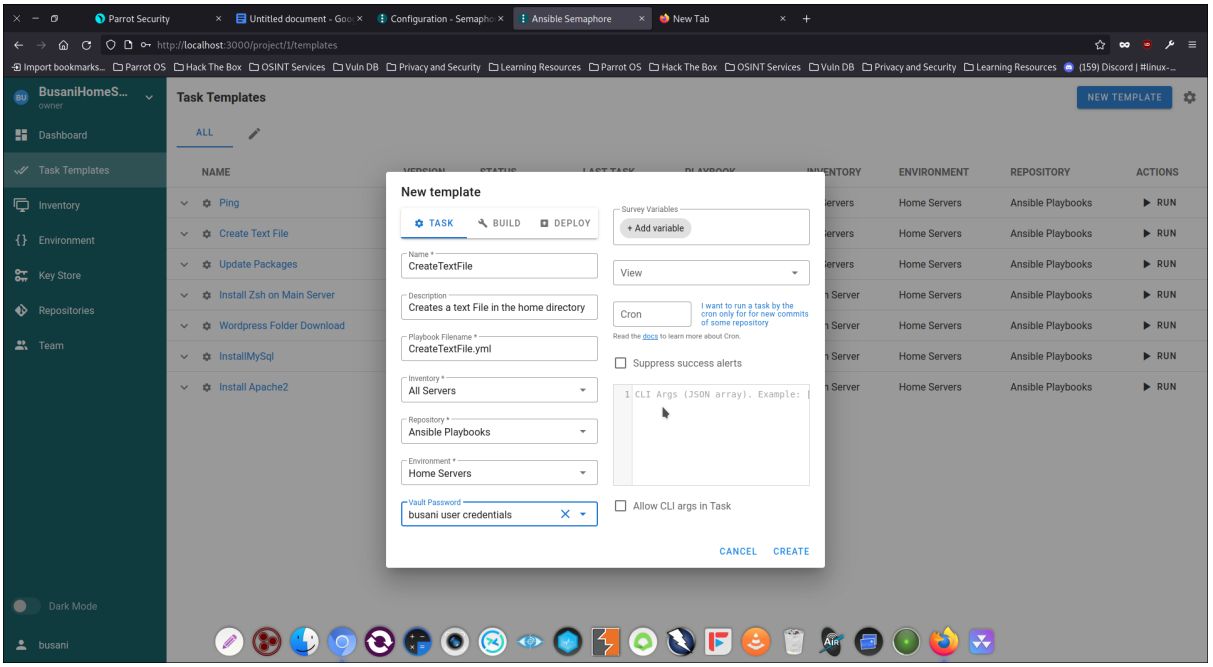
Now You are Ready to run Your First Automation

Create a CreateTextFile.yaml file in your github repository and paste the following

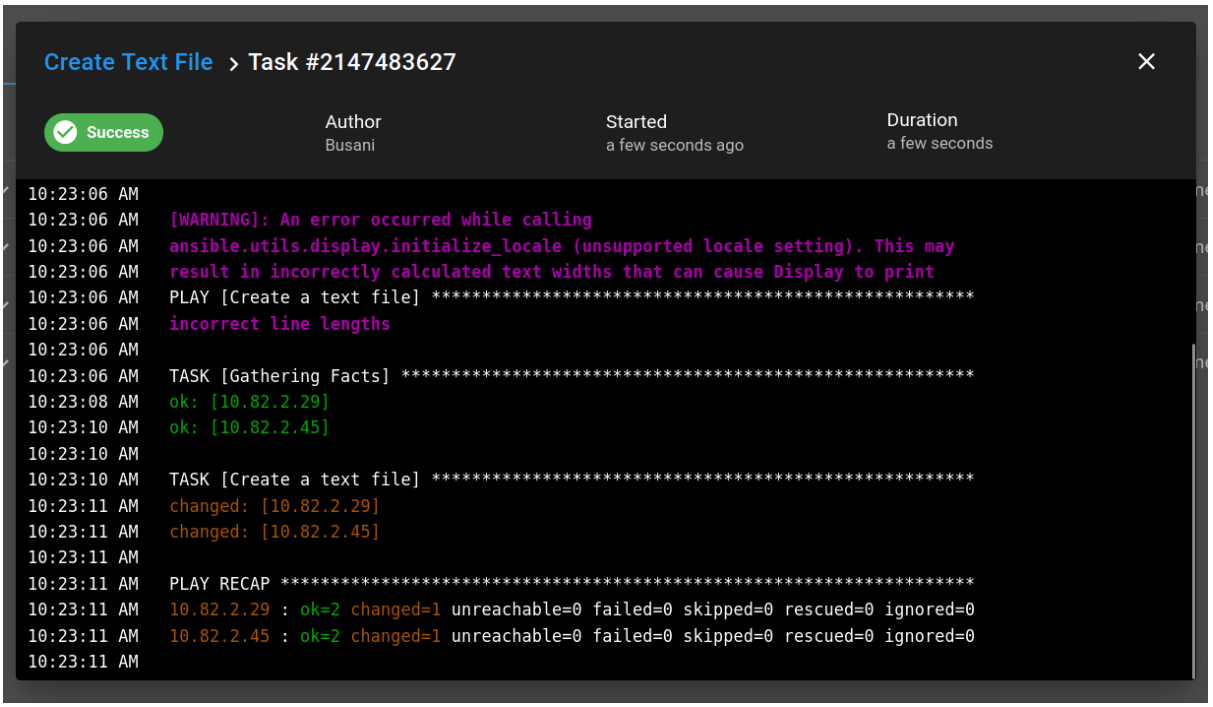
```
---
- name: Create a text file
  hosts: all
  become: yes  # Use 'yes' if you need elevated privileges (sudo)

  tasks:
        - name: Create a text file
        file:
        path: /home/busani/file.txt  # Specify the desired path and filename
        state: touch  # This ensures the file is created if it doesn't exist
        owner: busani  # Replace with the desired owner
        group: busani  # Replace with the desired group
        mode: "0644"  # Replace with the desired file permissions
```

{replace busani with your own username on your machine}

Create A new Task Template as Follows and then Click Create and then you should be able to Run the task template



Results should be as Follows



You should be able to see the file.txt on each of your Virtual Machine