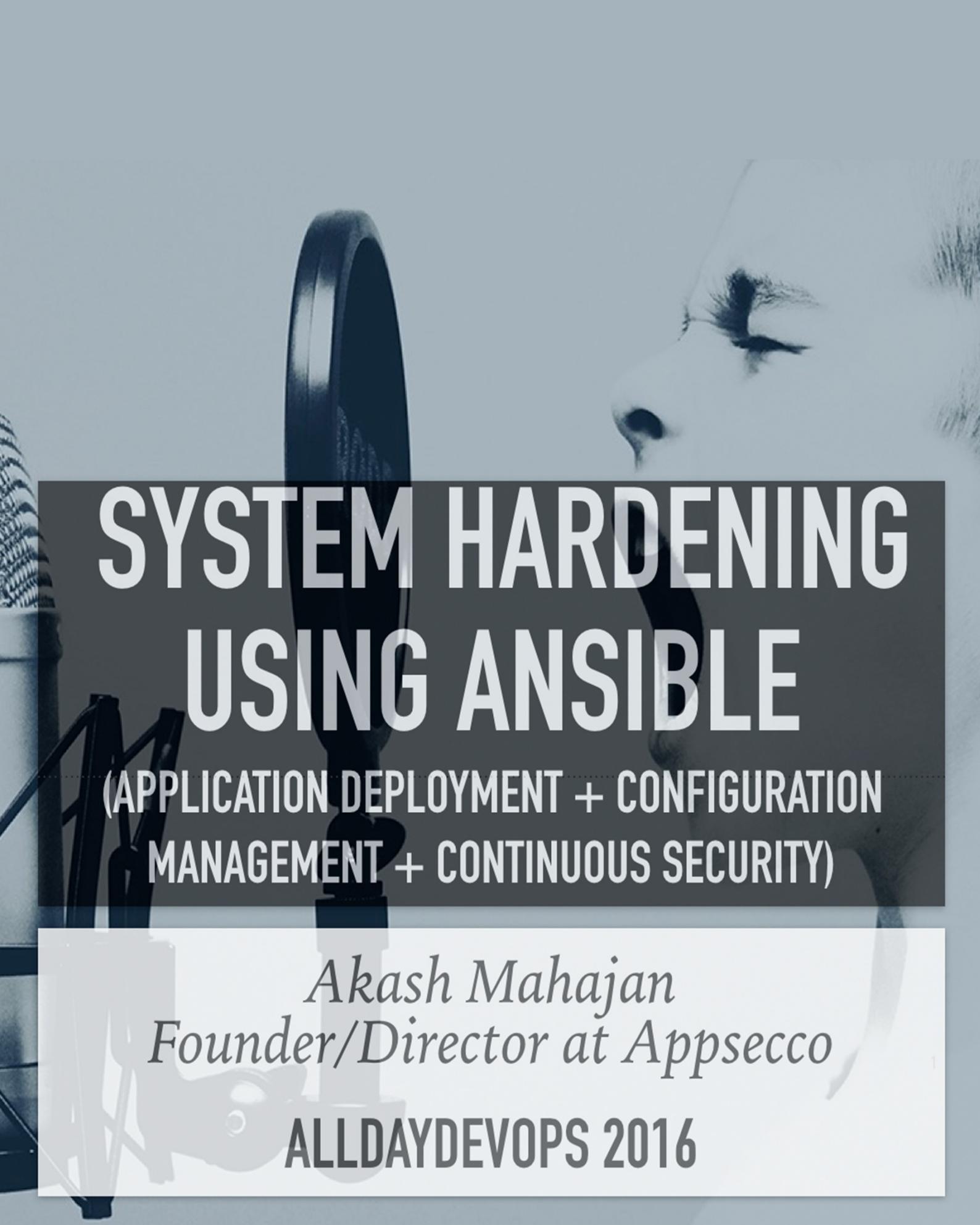


SYSTEM HARDENING USING ANSIBLE

(APPLICATION DEPLOYMENT + CONFIGURATION MANAGEMENT + CONTINUOUS SECURITY)

Akash Mahajan
Founder/Director at Appsecco

ALLDAYDEVOPS 2016

Table of Contents

Introduction	1.1
Running Ansible Playbooks with Jenkins	1.2

Security Hardening Using Ansible

Ansible offers a flexible approach to building a SecOps pipeline and by using it system hardening can become just another software project.

Using it we can do secure application deployment, configuration management and continuous monitoring.

Security can be codified & attack surfaces reduced by using Ansible.

This eBook contains a few recipes to get you started to enable continuous security monitoring with Ansible and tools like

- Jenkins (done)
- Ansible Tower(TBD)
- Rundeck(TBD)
- *Any Suggestions from your side*

Please sign-up for updates

We plan to add more recipies and tool integrations soon. If you would like to know when the new version is released please sign

https://appsecco.typeform.com/to/VeHwxS?v=addo-shua-ebook

Appsecco

Enable companies to design, specify, develop and purchase software that is secure

Work with companies to test existing software they have for security issues and give them the information they need to fix any problems we find

Ensure that companies can recover from security incidents they suffer and work with them to stop them from reoccurring

Please note

As with all [software] projects if you are not sure about what are doing seek expert advice as at no point this should be considered a substitute for professional advice.

Running Ansible Playbooks with Jenkins CI/CD

Adding the ansible plugin to Jenkins

This is a multi-step process. Thankfully we will do this only once.

Installing Ansible on the same host which has Jenkins

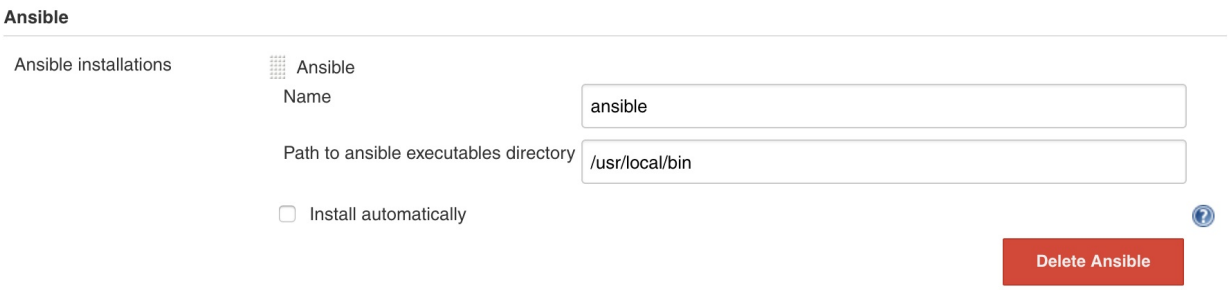
```
sudo apt-get install python-pip python-dev
sudo pip install ansible
sudo pip install markupsafe
```

Browse to Global Tool Configuration in Jenkins

https://example.com/jenkins/configureTools/

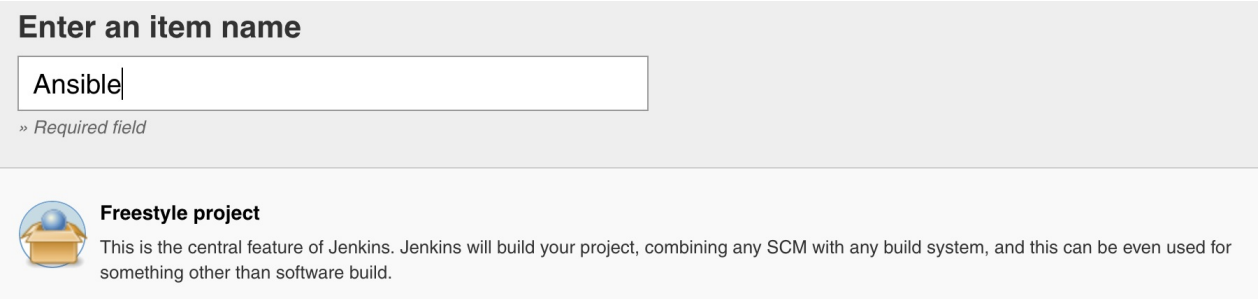
Substitute the domain with your domain or IP address

Scroll down on the web page



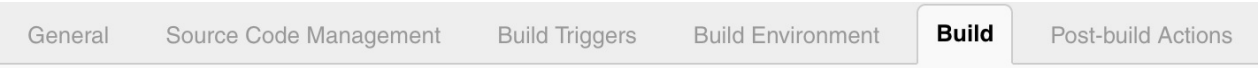
> The path may be different based on how you installed ansible. Please make sure you are using the correct path

Create a new job of type Freestyle project



Configure the job

Click on Build

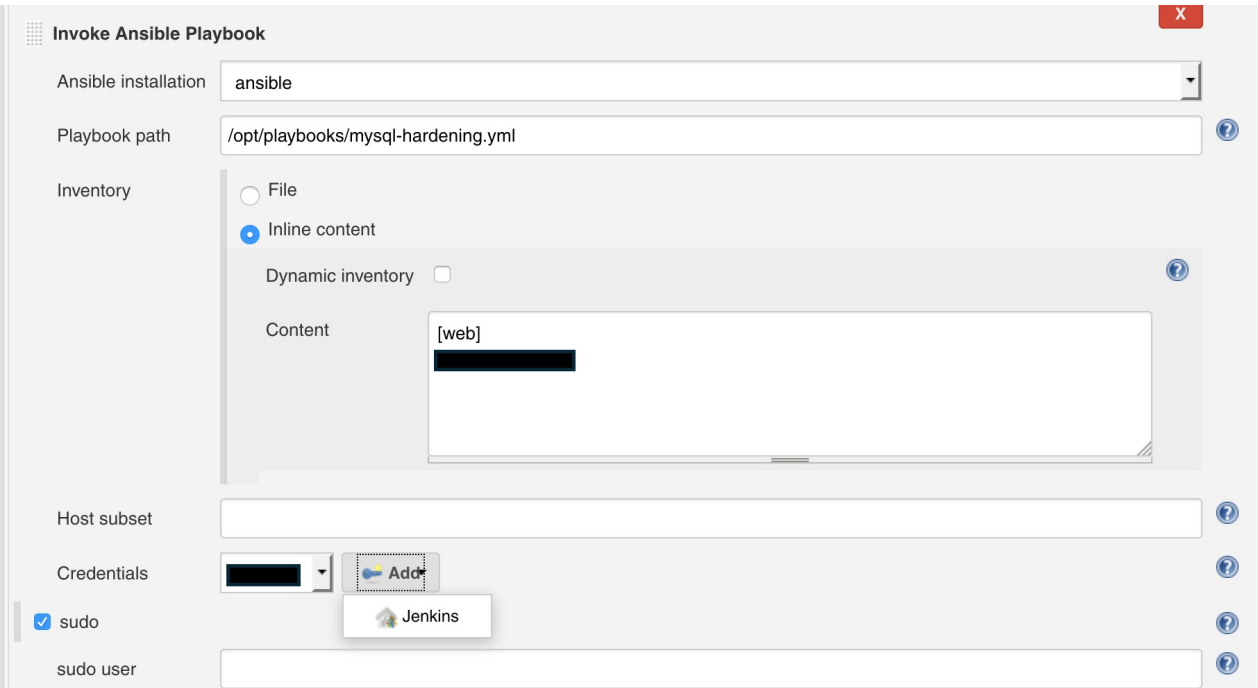


Fill in the details.

> Don't worry we will add all this to the system once Jenkins is configured

- Ansible Installation from dropdown
- Playbook Path /opt/playbooks/mysql-hardening.yml
- Inventory Content to show a text area
- Add credentials to jenkins
- Check `sudo` as we require it

Save this.



Now we are all set to execute the build (Ansible playbook) as required or even schedule it.

Configuration required on the host

We have already installed `ansible`.

Create a new directory in

```
$ sudo mkdir /opt/playbooks
```

Add the following content a new file `mysql-hardening.yml`

```
- hosts: all
  roles:
    - dev-sec.mysql-hardening
```

Configure Ansible

Depending on how you installed Ansible you may not have a folder in `/etc`

```
$ sudo mkdir /etc/ansible
```

Download the standard Ansible configuration file from github

```
$ cd /etc/ansible
$ wget https://raw.githubusercontent.com/ansible/ansible/devel/examples/ansible.cfg
```

Open the file in your favourite text editor and go to line number 57

```
$ sudo vi /etc/ansible/ansible.cfg
```

Edit this line from

```
#roles_path = /etc/ansible/roles
```

to this

```
roles_path = /opt/roles
```

There is no specific reason to create the `/opt/roles` directory. This is just for easy readability.

Ansible Galaxy

We can get the role for mysql hardening from Ansible Galaxy.

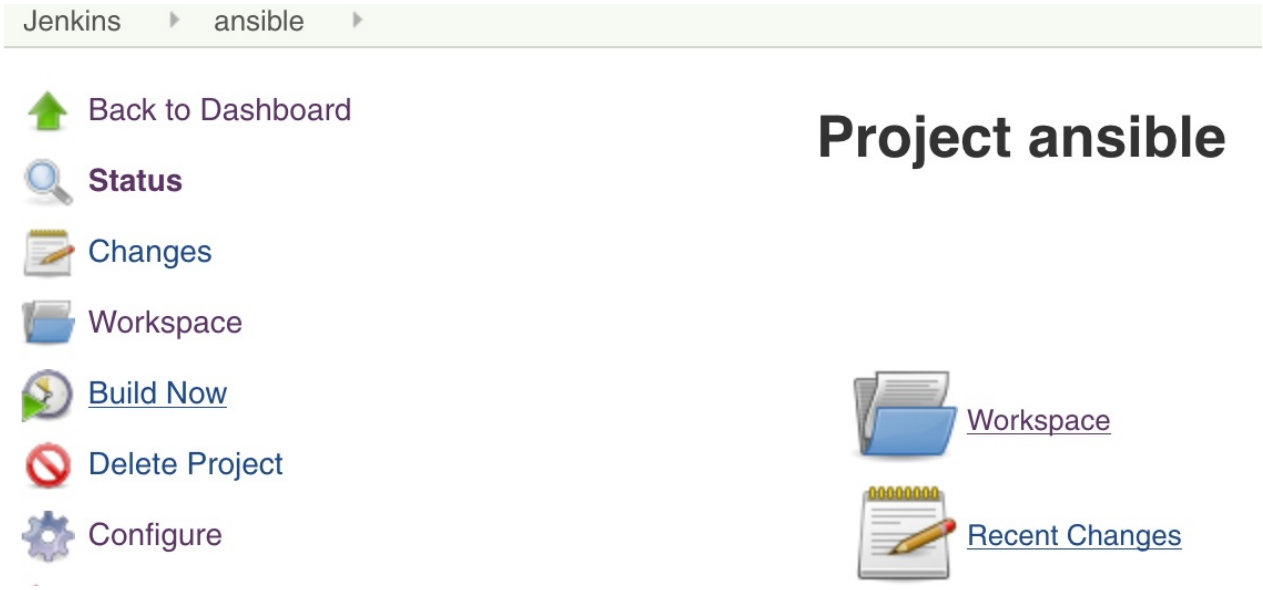
```
$ ansible-galaxy install dev-sec.mysql-hardening
```

Since we have configured the `roles_path` in the configuration file, the role will get downloaded to the following directory

```
$ ls -ltr /opt/roles
```

Now we have the playbook in the right place as configured in Jenkins and the role is installed as well.

We should be able to go ahead and build this in Jenkins



And we should be able to see the output of the playbook in console.


```
Started by user UserName LastName
Building in workspace /opt/bitnami/apps/jenkins/jenkins_home/workspace/ansible
[ansible] $ /usr/local/bin/ansible-playbook /opt/playbooks/mysql-hardening.yml -i /opt/bitnami/apache-tomcat
/temp/inventory8549216284887270689.ini -s -f 5 --private-key /opt/bitnami/apache-tomcat
/temp/ssh3135481797688612257.key -u [redacted]

PLAY [all] *********************************************************************

TASK [setup] *******************************************************************
ok: [redacted]

TASK [dev-sec.mysql-hardening : add the OS specific variables] *****************
ok: [redacted]

TASK [dev-sec.mysql-hardening : protect my.cnf] ********************************
ok: [redacted]
```

By Akash Mahajan, Director Appsecco