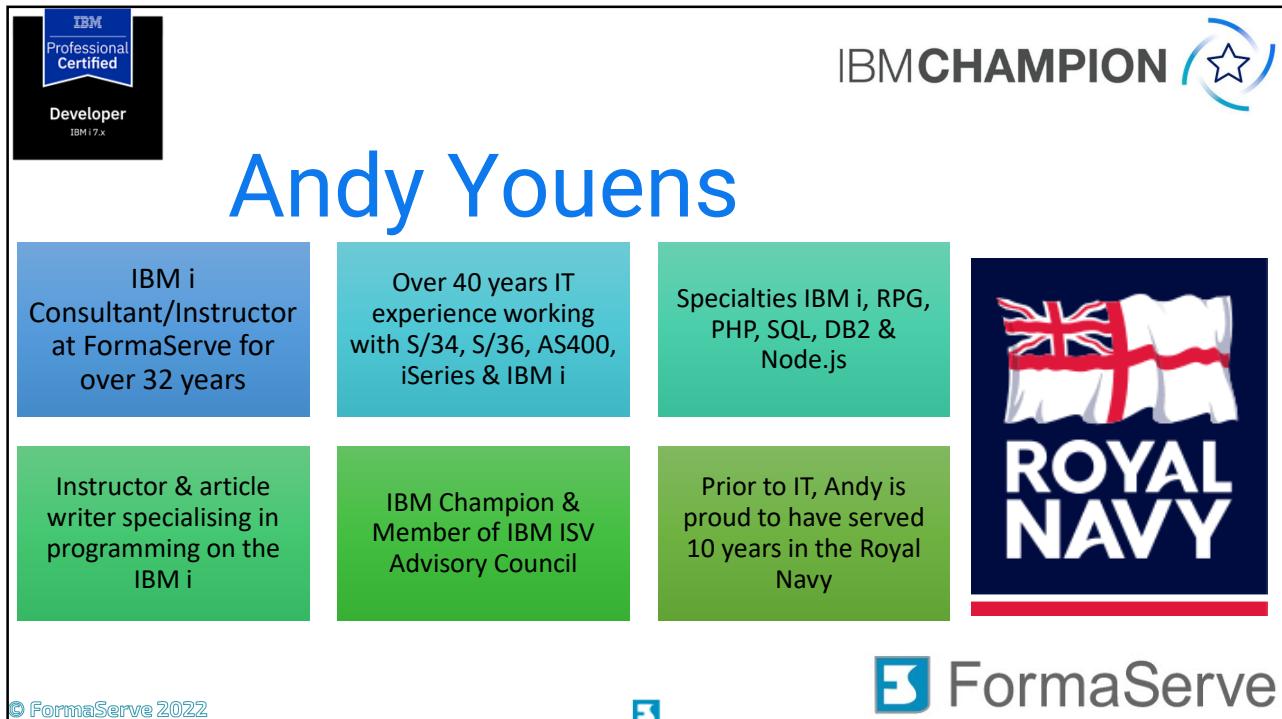




The slide features several circular graphic elements. In the top left is a large circle containing the "IBM i" logo. To its right is another circle containing the "FormaServe" logo. Further to the right is a circle containing the "IBM CHAMPION" text and a blue star icon. Below these circles, on the left, is a red circle containing a white "A" followed by the text "Ansible & IBM i Workshop". Below this text is the name "Andy Youens" in a large, stylized blue font.

1



Andy Youens

IBM Professional Certified Developer IBM i 7.x

IBM i Consultant/Instructor at FormaServe for over 32 years

Over 40 years IT experience working with S/34, S/36, AS400, iSeries & IBM i

Specialties IBM i, RPG, PHP, SQL, DB2 & Node.js

Instructor & article writer specialising in programming on the IBM i

IBM Champion & Member of IBM ISV Advisory Council

Prior to IT, Andy is proud to have served 10 years in the Royal Navy

ROYAL NAVY

© FormaServe 2022

FormaServe

2

Ansible Workshop Agenda



Part 1 – Andy!

- Introduction
- Installation
- Configuration
- Ad-Hoc Commands
- YAML
- Playbooks
- Encryption
- Top Tips
- Demo
 - WSL will be the control node
 - WSL to monitor IBM i & Windows Laptop

Part 2 – Everyone Else!

- Monitor IBM i's
- One IBM i will be the control node
- Run Ad-hoc commands
- Create/run playbooks
- Work individually or a group
- You decide!

GitHub Examples

- All examples from this workshop can be found in our GitHub repo
- https://github.com/FormaServe/f_Learning/tree/master/Ansible
- Presentation slides can be found in https://github.com/FormaServe/f_Learning/tree/master/pdf

f_Learning Ansible Help

Introduction 🎉

This document is intended as a crib-sheet when presenting Ansible on IBM i workshops at User Groups

Ping a group

Description	Command
Ping IBM i specifying inventory file	ansible ibmi -i ~/ansible/hosts -m ping
Limit ping to 1 IBM i only	ansible ibmi -i ~/ansible/hosts -m ping --limit Galatea
Show detailed log	ansible ibmi -m ping -v

IBM i AdHoc Commands

Description	Command
Create library	ansible ibmi -m ibm.power_ibmi_ibmi_cl_command -a "cmd='crtlib lib(AYOUENS)'"
Create library with joblog	ansible ibmi -m ibm.power_ibmi_ibmi_cl_command -a "cmd='crtlib lib(AYOUENS) text(ansible)' joblog=true"
Display IBM i facts	ansible ibmi -m ibm.power_ibmi_ibmi_facts
Upgrade git on a box	ansible ibmi -m yum -a "name=git state=latest"
Upgrade all open-source on a box	ansible ibmi -m yum -a "name=* state=latest"

Galaxy Stuff

Description	Command
IBM i Import	ansible-galaxy collection install ibm.power_ibmi

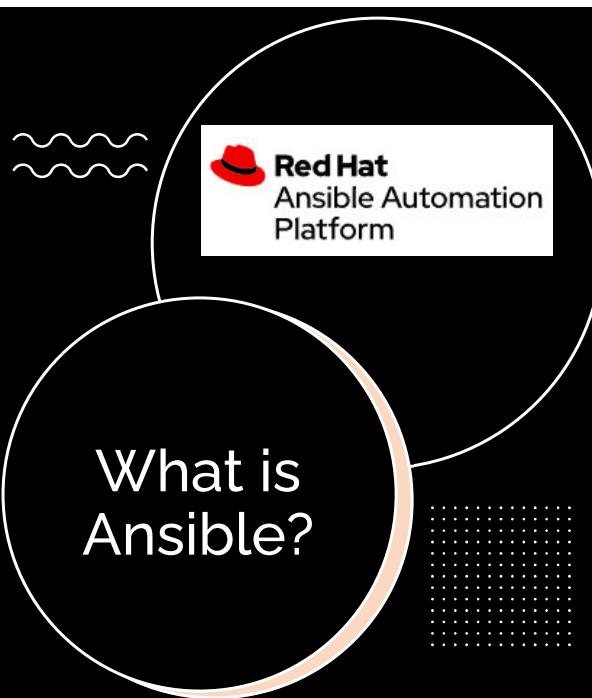
Ansible Introduction



© FormaServe 2022

[www.FormaServe.co.uk](#)

5



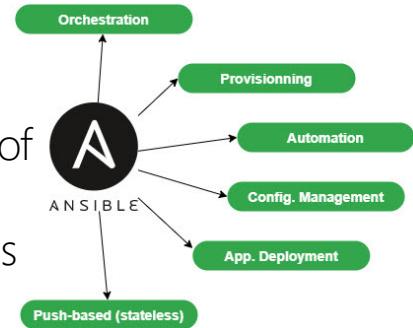
- First version 2012 from Ansible Works
- Acquired by Red Hat 2015
- A simple IT automation platform
- An open-source application from Red Hat*
- Written in Python & YAML based
- Highly flexible & configuration management of systems
- Configuration roll-back in case of error
- Report errors at <https://github.com/IBM/ansible-for-i/>

6

Ansible Advantages



- **Free** - It is open-source
- **Simple** - No problems in setting up & using
- **Powerful** - You can configure thousands of machines at a time
- **Flexible** - Supports multiple environments
- **Agentless** - Nothing to install on your hosts
- **Efficient** - It is light on your server



What can Ansible do for IBM i?



- Automate traditional IBM i administration tasks
- Improve application development process & efficiency to shorten software delivery cycle
- Manage multiple IBM i systems with interactive command lines
- Automate IBM i tasks with playbooks
- Allows operators with no IBM i skills to do basic management
- Ties the knot between open-systems & propriety systems

What can Ansible do for me?

The diagram illustrates the FormaServe IBM i Network architecture. A central blue horizontal bar represents the network backbone. Four client partitions, each with a blue icon featuring three horizontal lines and a dot, are connected to it. The partitions are labeled from left to right: Proteus, Galatea, Despina, and Sao. Above the network backbone, there is a small red circle containing a white letter 'A'.

FormaServe IBM i Network

© FormaServe 2022



www.FormaServe.co.uk

9

I Used Ansible to Setup this Workshop

- On every partition
 - Set the default shell to be BASH
 - For every user
 - Create user profile with home directory
 - Create a library for user profile
 - Create home directory on IFS
 - Create sub-directory Ansible off users home directory
 - Create sub-directory .ssh off users home directory
 - In the .ssh directory create an authorized_keys file
 - Change authority on users home directory for full access, including sub-directories
 - Create user shell profile & setup OSS path & BASH cursor

ANSIBLE

© FormaServe 2022



www.FormaServe.co.uk

10

What do you want to automate today?



Infrastructure
Build, provision, and manage applications and infrastructure across public or private cloud, containers, and virtual environments.

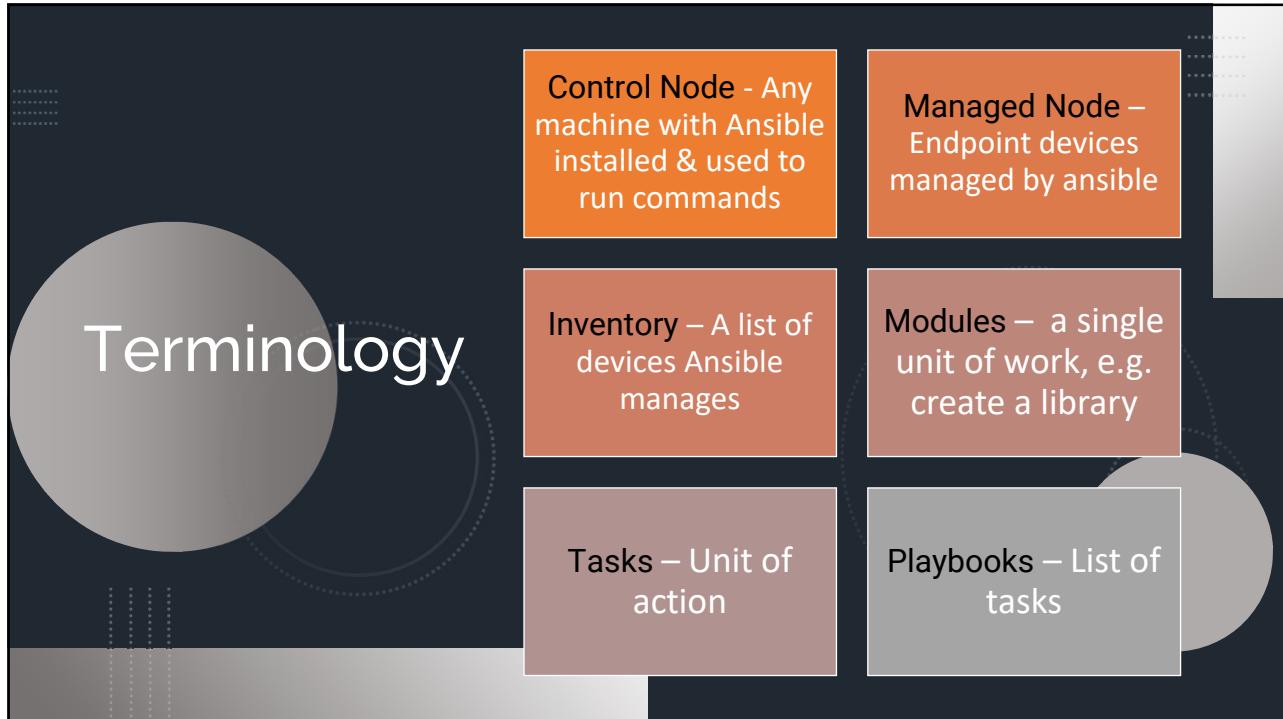
Applications
Automate your application deployments and make your installations, upgrades, and day-to-day management repeatable and reliable.

Networks
Manage entire network and IT processes across physical networks, software-defined networks, and cloud-based networks.

Containers
Manage and automate your Kubernetes clusters and scale containerized applications with support for Kubernetes Operators and frameworks.

Security
Orchestrate security systems using a curated collection of modules, roles, and playbooks to investigate and respond to threats.

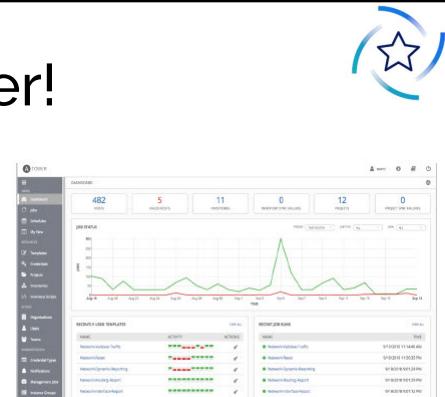
Cloud
Provision instances, networks, and infrastructure with support modules that ensure deployments work across public and private clouds.



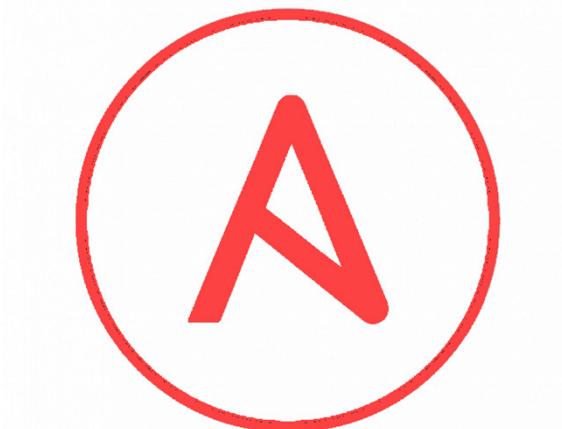
Not Covered – Ansible Tower!

- Commercial offering
- Enterprise version of Ansible, helps companies & teams scale quickly & effectively
- There is a cost associated with adopting it, installing it, & getting the software up & running in your environments

• **Everything else is Free!**



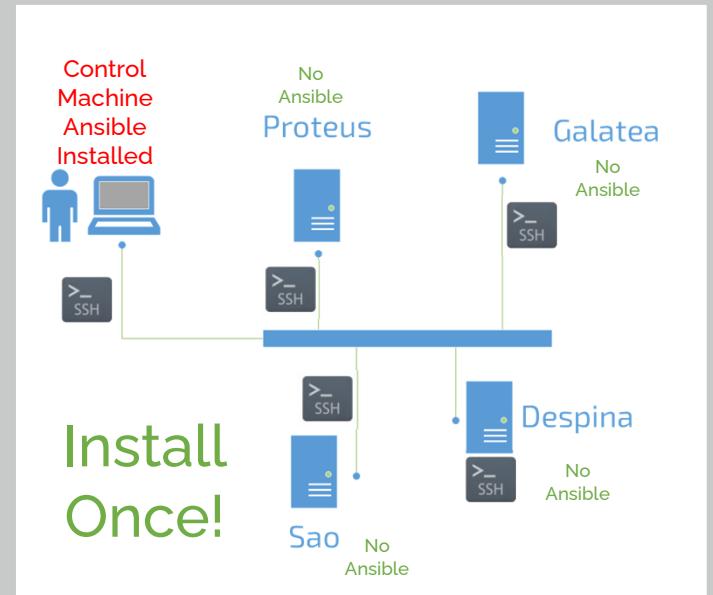
Install
Ansible



ANSIBLE

Ansible is Agentless

- One server with Ansible
- Ansible knows of 2 machine types
 - Control machine** – Manages other machines
 - Remote machine** – Handled by control machine

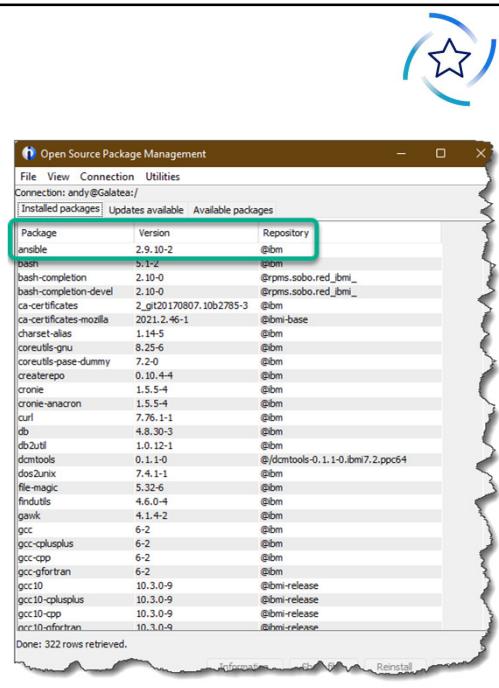


© FormaServe 2022

15

Installing Ansible

- IBM i for Control Node**
 - ACS Open Source Package Management
 - YUM install ansible
- WSL**
 - sudo apt install ansible
- Pre-req's on IBM i Nodes**
 - IBM i V7.3 & >
 - 5733SC1 - PASE
 - 5770DG1 - HTTP
 - Python3
 - Python3-iToolkit
 - Python3-ibm_db



© FormaServe 2022

16

Communications Methods



- **Linux/Unix**

- Uses SSH

- **IBM i**

- Uses SSH

- **Windows**

- Uses Powershell or SSH if configured

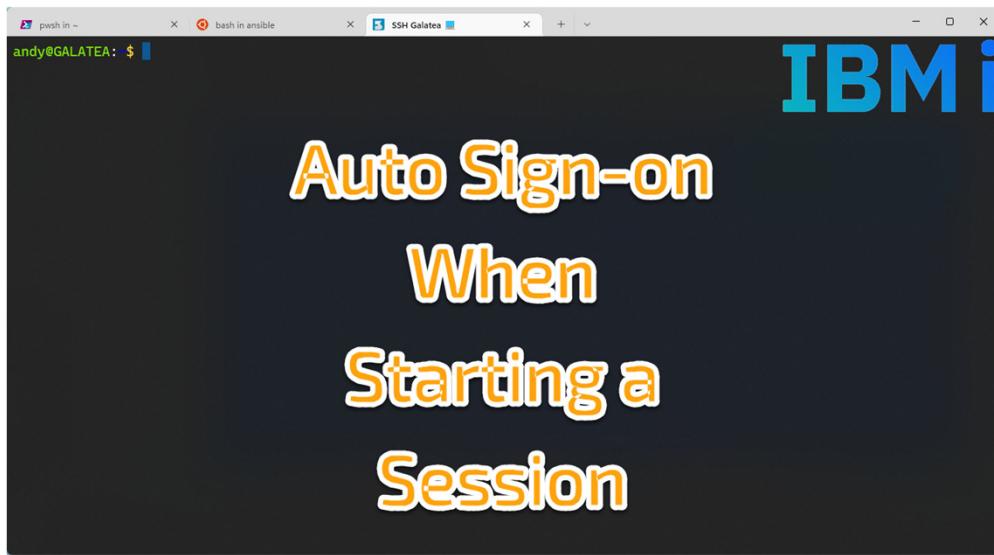


Communications to Hosts



- The **Control node** needs information to connect to nodes
 - The location of the **inventory** file, if not using default
 - The connection protocol to use - **Default SSH**
 - Whether non-standard ports are used (not 22)
 - What user it can login as
 - How Ansible should become root (sudo/QSECOFR)
 - Whether to prompt for a **SSH** password or key pair
- Lets make life easier for Ansible & setup SSH keys ...

SSH Keys



© FormaServe 2022



www.FormaServe.co.uk

19

How Does it Work?



- We have a pair of keys, a private key & a public key
- Each key pair is unique, the two keys work together
- If you have the private key, you can prove you have it without showing what it is
- It's like proving you know a password without having to show someone the password



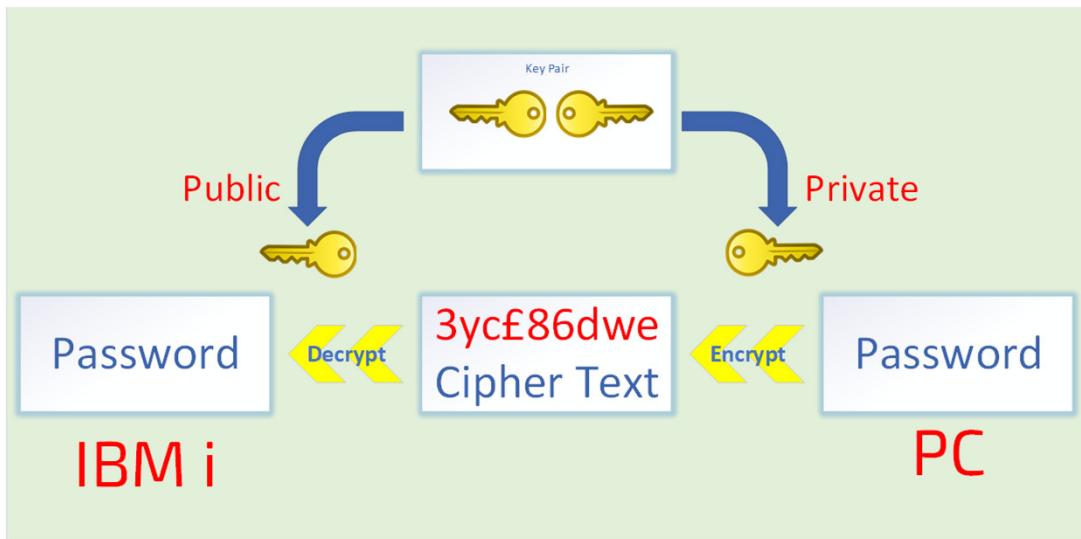
© FormaServe 2022



www.FormaServe.co.uk

20

Key Authentication



© FormaServe 2022



www.FormaServe.co.uk

21

Generating Keys - Use ssh-keygen



- ➊ We can use ...
 - ➌ Windows
 - ➌ Git for Windows
 - ➌ Putty



22



Send Public Key to IBM i

- Home .ssh Directory

Screenshot of a file browser showing the contents of the .ssh directory in the user's home folder on an IBM i system. The 'authorized_keys' file is highlighted with a green oval.

Name	Date modified	Type	Size
authorized_keys	06/01/2021 15:58	File	1 KB
championtweets.deploy.pem	21/05/2021 17:08	PEM File	4 KB
id_rsa	20/03/2021 13:34	File	3 KB
id_rsa.pub	20/03/2021 13:34	Microsoft Publish...	1 KB
id_rsa_old.pub	14/01/2020 16:36	Microsoft Publish...	1 KB
id_rsaagh.pub	21/05/2021 17:08	Microsoft Publish...	1 KB
known_hosts	20/03/2021 13:40	File	1 KB

© FormaServe 2022



www.FormaServe.co.uk

23

SSH Keys Not Working?



- Home Directory

- Are you the owner? - `ls -ld /home/Andy/`
- `chown andy /home/andy`
- Check permissions on home directory - `drwxr-sr-x`
- `chmod 755 /home/andy/`

- .ssh Directory

- Are you the owner of this directory? - `ls -ld /home/Andy/.ssh/`
- `chown andy /home/andy/.ssh`
- Check authority for `-rw - drwx-S--3`
- `chmod 700 /home/andy/.ssh/`

- Authorized_keys

- Check authority for `-rw`
- `chmod 600 /home/andy/.ssh/authorized_keys`

© FormaServe 2022



www.FormaServe.co.uk

24

Configure Ansible



© FormaServe 2022

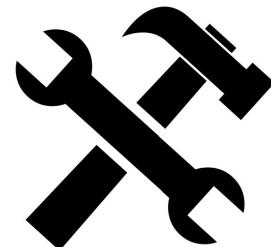


www.FormaServe.co.uk

25

Configuring Ansible

- ➊ Two main configuration files
 - ➌ Inventory/host
 - ➌ Ansible configuration
- ➋ Configure communications



© FormaServe 2022



www.FormaServe.co.uk

26



Inventory File

- ➊ Simplest form, a **list** of host names/IP addresses
- ➋ Can be named whatever you want!
- ➌ Can have any/no extension
- ➍ Common names
 - ➎ hosts/hosts.ini
 - ➎ inventory/**inventory.ini**
- ➎ Pick a name & stick with it!

```
10.0.0.36
10.0.0.39
10.0.0.37
10.0.0.38
81.142.39.107
10.0.2.3
10.0.0.2
norfolk
mercury
sun
jupiter
10.0.0.36
10.0.0.39
10.0.0.37
10.0.0.38
81.142.39.107
10.0.2.3
```

Inventory File - Groups

- ➊ Fit your hosts into groups for ease
- ➋ Three types of group
 - ➎ All - Every hosts
 - ➎ Named – **Ones we configure!**
 - ➎ Ungrouped - Contains every managed node that is not part of a group
- ➌ Localhost already present
- ➍ Do **NOT** give a group the same name as a host!

```
[adservers]
10.0.0.2

[windows]
norfolk
mercury

[ibmi]
10.0.0.36
10.0.0.39
10.0.0.37
10.0.0.38

[webservers]
81.142.39.107

[iot]
10.0.2.3
```

Inventory File - Groups



- ➊ A host can be a member of multiple groups
- ➋ This allows you to organise groups in different ways, for example ...
 - ➌ Web servers or database servers
 - ➌ Production & testing servers
 - ➌ Development machines
 - ➌ IBM i
 - ➌ By countries/regions

```
[adservers]
10.0.0.2
```

```
[windows]
norfolk
mercury
```

```
[ibmi]
10.0.0.36
10.0.0.39
10.0.0.37
10.0.0.38
```

```
[webservers]
81.142.39.107
```

```
[iot]
10.0.2.3
```

Inventory File - Groups



- ➊ Groups can contain specific host information

What about security??

```
[ibmi]
10.0.0.36
proteus ansible_host=10.0.0.39 ansible_python_interpreter="/QOpenSys/pkgs/bin/python2"
despina ansible_ssh_user="JRiddle" ansible_ssh_pass="James"
sao
pub400 ansible_ssh_port=2222 ansible_ssh_user="AYouens" ansible_ssh_pass="NoIdea"

[ibmi:vars]
# ansible_python_interpreter="/QOpenSys/pkgs/bin/python3"
ansible_python_interpreter="/QOpenSys/pkgs/bin/python3.6"
ansible_ssh_common_args="-o StrictHostKeyChecking=no"
```

Inventory File - Listing



- Use to list current inventory
- **ansible-inventory --list**
- Great debugging tool
- Useful if getting unpredicted results



```
andy@Norfolk:~/ansible$ ansible-inventory --list
{
    "_meta": {
        "hostvars": {
            "10.0.0.36": {
                "ansible_python_interpreter": "/QOpenSys/pkgs/bin/python3.6",
                "ansible_ssh_common_args": "-o StrictHostKeyChecking=no"
            },
            "despina": {
                "ansible_python_interpreter": "/QOpenSys/pkgs/bin/python3.6",
                "ansible_ssh_common_args": "-o StrictHostKeyChecking=no",
                "ansible_ssh_pass": "James",
                "ansible_ssh_user": "JRiddle"
            },
            "proteus": {
                "ansible_host": "10.0.0.39",
                "ansible_python_interpreter": "/QOpenSys/pkgs/bin/python2",
                "ansible_ssh_common_args": "-o StrictHostKeyChecking=no"
            },
            "pub400": {
                "ansible_python_interpreter": "/QOpenSys/pkgs/bin/python3.6",
                "ansible_ssh_common_args": "-o StrictHostKeyChecking=no",
                "ansible_ssh_pass": "NoIdea",
                "ansible_ssh_port": 2222,
                "ansible_ssh_user": "AYouens"
            },
            "sao": {
                "ansible_python_interpreter": "/QOpenSys/pkgs/bin/python3.6",
                "ansible_ssh_common_args": "-o StrictHostKeyChecking=no"
            }
        }
    }
}
```

© FormaServe 2022



www.FormaServe.co.uk

31

Ansible Configuration File



- During installation a sample config file at **/etc/ansible/ansible.cfg**
- It is recommended this file is copied to your default location
- A large file that contains all configuration values
- All values can be overwritten

```
# config file for ansible -- https://ansible.com/
# =====

# nearly all parameters can be overridden in ansible-playbook
# or with command line flags. ansible will read ANSIBLE_CONFIG,
# ansible.cfg in the current working directory, .ansible.cfg in
# the home directory or /etc/ansible/ansible.cfg, whichever it
# finds first

# some basic default values ...
[defaults]
# where is the inventory?
inventory      = ~/ansible/inventory

#library       = /usr/share/my_modules/
#module_utils  = /usr/share/my_module_utils/
#remote_tmp   = ~/ansible/tmp
#log_dir      = ~/ansible/
```

© FormaServe 2022



www.FormaServe.co.uk

32



Which Ansible Config File?

- Will use environment variable
 - **ANSIBLE_CONFIG (export ANSIBLE_CONFIG='/ansible')**
- Will search for a config file in the following order
 - **ansible.cfg** – Current working directory
 - **~/.ansible.cfg** – dot file in users home directory
 - **QOpenSys/etc/ansible/ansible.cfg** – default configuration, if no other config file has been found
- **ansible --version** or **ansible-config --version** will identify which configuration file is being used



Default File Locations

Name	Directory
Named Hosts or Inventory	(/QOpenSys)/etc/ansible/hosts
Configuration	(/QOpenSys)/etc/ansible/ansible.cfg

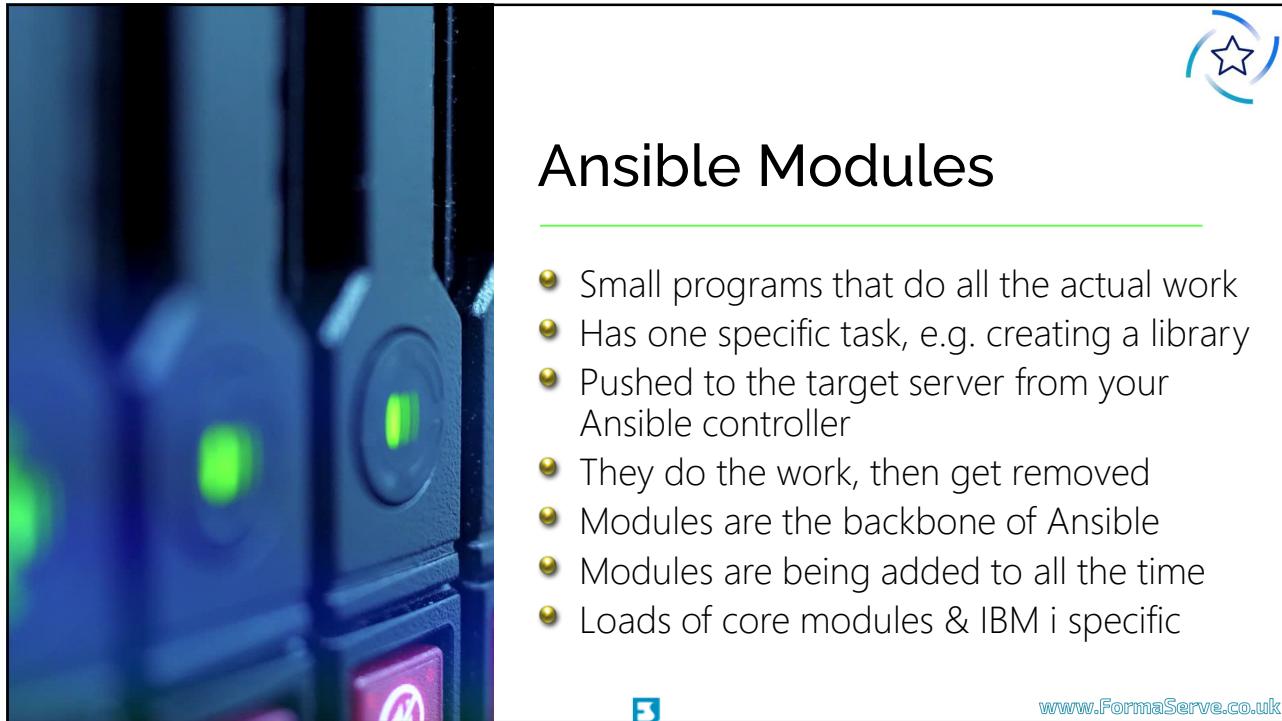
```
andy@Norfolk:~/ansible$ ansible --version
ansible 2.9.6
  config file = /home/andy/ansible/ansible.cfg
  configured module search path = ['/home/andy/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.8.10 (default, Mar 15 2022, 12:22:08) [GCC 9.4.0]
andy@Norfolk:~/ansible$
```



The slide features the Ansible logo, which consists of a red circle containing a white letter 'A'. Below the circle, the word 'ANSIBLE' is written in red capital letters. In the top right corner of the slide area, there is a small blue circular icon with a white star and a curved arrow.

Ansible Ad-hoc Commands

35



The slide features the Ansible logo in the top right corner. The main title 'Ansible Modules' is centered at the top of the slide. Below the title is a horizontal green line. To the left of the title is a blurred photograph of a server rack with glowing blue and green lights from the server units.

- Small programs that do all the actual work
- Has one specific task, e.g. creating a library
- Pushed to the target server from your Ansible controller
- They do the work, then get removed
- Modules are the backbone of Ansible
- Modules are being added to all the time
- Loads of core modules & IBM i specific

 www.FormaServe.co.uk

36

Core Modules

```
- name: Install Open Source packages
  yum:
    name:
      - nginx
      - postgresql
      - postgresql-server
    state: present
```

```
- name: Copy file to new environment
  copy:
    src: /ansible/environment.txt
    dest: /formaserve/test/config.ini
    owner: QSECOFR
```



```
- name: System timestamp - date
  debug:
    msg: "Current date: {{ cur_date }}"
```

```
- name: Use git to clone Repo & archive
  git:
    repo: https://github.com/formaserve/f_learning.git
    dest: /node_apps/training
    archive: /tmp/training-examples.zip
```

```
- name: Write some content in a file
  copy:
    dest: /home/formaserve/testing.txt
    content: |
      Good Morning.
      Good Afternoon.
```

Ansible Core Modules & More ...

- Use [ansible-doc](#) module-name for full details
- Great for examples
- List modules
 - [ansible-doc -l](#)
- Full details
 - [Ansible-doc ping](#)

- To be selective
 - [ansible-doc -l | grep ibm](#)

```
azurerm_aks_info
azurerm_aksversion_info
azurerm_appgateway
azurerm_applicationsecuritygroup
azurerm_applicationsecuritygroup_info
azurerm_appserviceplan
azurerm_appserviceplan_info
azurerm_automationaccount
azurerm_automationaccount_info
azurerm_autoscale
azurerm_autoscale_info
azurerm_availabilityset
azurerm_availabilityset_info
azurerm_azurediagnostic
azurerm_azurediagnostic_info
azurerm_batchaccount
azurerm_cdnendpoint
azurerm_cdnendpoint_info
azurerm_cdnpolicy
azurerm_cdnpolicy_info
azurerm_containerinstance
azurerm_containerinstance_info
azurerm_containerregistry
azurerm_containerregistry_info
azurerm_cosmosdbaccount
azurerm_cosmosdbaccount_info

Get Azure Kubernetes Service facts
Get available kubernetes versions supported by Azure ...
Manage Application Gateway instance
Manage Azure Application Security Group
Get Azure Application Security Group facts
Manage App Service Plan
Get Azure app service plan facts
Manage Azure Automation account
Get Azure automation account facts
Manage Azure autoscale setting
Get Azure Auto Scale Setting facts
Manage Azure Availability Set
Get Azure Availability Set facts
Manage Azure Firewall instance
Get AzureFirewall info
Manages a Batch Account on Azure
Manage a Azure CDN endpoint
Get Azure CDN endpoint facts
Manage a Azure CDN profile
Get Azure CDN profile facts
Manage an Azure Container Instance
Get Azure Container Instance facts
Manage an Azure Container Registry
Get Azure Container Registry facts
Manage Azure Database Account instance
Get Azure Cosmos DB Account facts
```

[ansible-doc -l | grep azure](#)



Ansible Command Arguments

Argument	Description	Example
-m	Specify module to run	-m ping
-i	Specify your inventory file	-i hosts.ini
-e	Specify extra variables (Overrides)	-e "key=123"
-v	Verbose mode (Joblog)	
-vvv	Show connection details	
-u	Specify remote user	-u QSECOFR
-h	Show help	
-b	Become a user	-b ANDY
-a	Module attributes	-m yum -a 'name=git state=latest'



Pinging IBMi Servers

- ➊ Ping - A great starting point! 
- ➋ Doesn't run TCP/IP Ping

```

andy ➔ □ ansible ➔ 83.101s ➔ ansible ibmi -m ping
galatea | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
proteus | SUCCESS => {
    "changed": false,
    "ping": "pong"
}

```

Two green checkmarks are placed below the command output, one next to each server name.

Create Library IBMi Servers



- Log is colour coded to reflect results

```
andy@Norfolk:~/ansible$ ansible ibmi -m command -a "system 'crtlib lib(AYOUENS) text(ansible)'"
10.0.0.36 | CHANGED | rc=0 >>
CPC2102: Library AYOUENS created
andy@Norfolk:~/ansible$ ansible ibmi -m command -a "system 'crtlib lib(AYOUENS) text(ansible)'"
10.0.0.36 | FAILED | rc=255 >>
CPF2111: Library AYOUENS already exists.non-zero return code
andy@Norfolk:~/ansible$
```

The log output shows two entries. The first entry is highlighted with a green checkmark and a red border, indicating a successful creation of the library. The second entry is highlighted with a red X and a red border, indicating a failure because the library already exists.

Colour Coding Commands



```
TASK [Create User Profile] *****
changed: [10.0.0.36] => (item=AJESSICAJ)
changed: [10.0.0.36] => (item=AEMILYY)
changed: [10.0.0.36] => (item=ASAMMYJ)
changed: [10.0.0.36] => (item=ACHLOEY)
changed: [10.0.0.36] => (item=AELLIEJ)

TASK [Create user library] *****
ok: [10.0.0.36] => (item=AJESSICAJ)
ok: [10.0.0.36] => (item=AEMILYY)
ok: [10.0.0.36] => (item=ASAMMYJ)
ok: [10.0.0.36] => (item=ACHLOEY)
ok: [10.0.0.36] => (item=AELLIEJ)
```

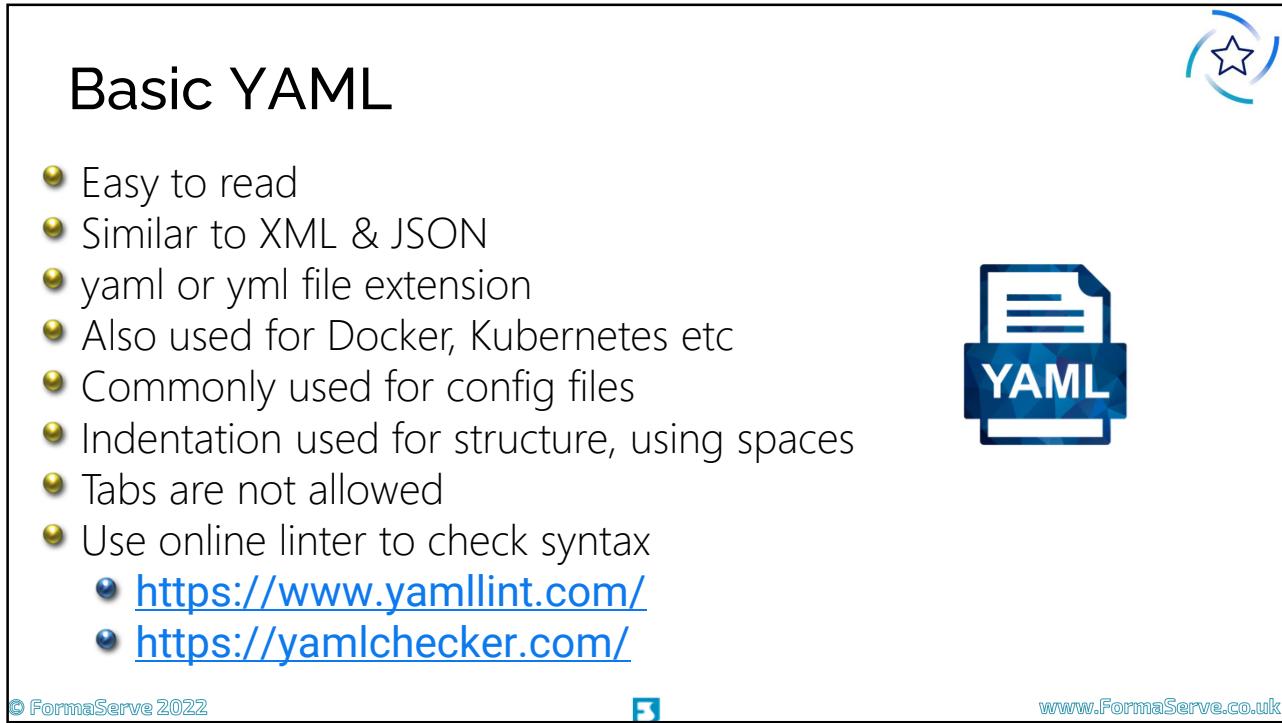
The log output shows five 'changed' entries for the 'Create User Profile' task, each highlighted with a green checkmark and a red border. It also shows five 'ok' entries for the 'Create user library' task, each highlighted with a green checkmark and a red border. There are no failed or error messages in this log.



The slide features the Ansible logo, which consists of a red circle containing a white letter 'A'. Below the circle, the word 'ANSIBLE' is written in red capital letters. In the top right corner of the slide area, there is a small blue circular icon with a white star and a curved arrow.

YAML in 5 Minutes!

43



The slide has a title 'Basic YAML' in large black font at the top left. In the top right corner is a small blue circular icon with a white star and a curved arrow.

Basic YAML

- Easy to read
- Similar to XML & JSON
- yaml or yml file extension
- Also used for Docker, Kubernetes etc
- Commonly used for config files
- Indentation used for structure, using spaces
- Tabs are not allowed
- Use online linter to check syntax
 - <https://www.yamllint.com/>
 - <https://yamlchecker.com/>

A blue icon depicting a document with horizontal lines and the word 'YAML' in white on a dark blue background.

Basic YAML



- Ansible uses **YAML**
- YAML** optionally starts with **---**
- Most commonly, uses simple **key-value** pair to represent data
- Represented in **key: value** pair



```
# key-pairs syntax
app: end-of-day
port: 8080
version: 1.9.9
OS: 'IBM i 7.3'
```

There should be a space between : & value

Basic YAML - Comments



```
# Produced for iUG Workshop
# Copyright © FormaServe Systems Ltd 2022
# April 2022

---

# YAML Examples

# key-pairs syntax
app: end-of-day
port: 8080
```



Basic YAML – Key Pairs

```
# key-pairs syntax
app: end-of-day
port: 8080
version: 1.9.9
OS: 'IBM i 7.3'
```



Basic YAML – Objects

```
# object syntax
customer:
  name: 'Andy Youens'
  address: 'Tottenham'
  postcode: N17
```

Basic YAML – Lists



```
# list syntax
users:
  - AYouens
  EYouens
  JJackman
  SJackman
  CYouens
  EJackman
```

Basic YAML – Boolean



```
# boolean values
# all true
customer1: true
supplier1: yes
ordered1: on
# all false
customer2: false
supplier2: no
ordered2: off
```



Multi & Single Lines in YAML

```
# multi line string
multiline: |
    This is a multi line string
    that spans multiple lines
    and is indented.

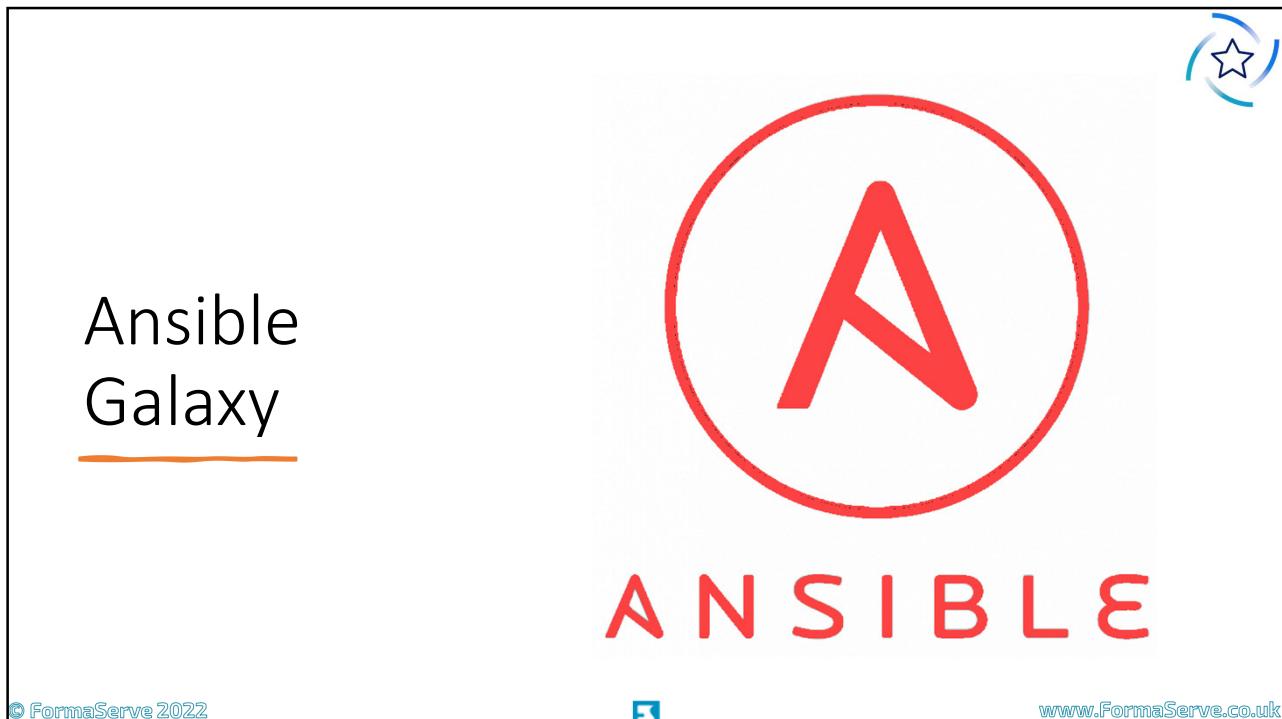
# single line string
singleline: >
    This is a single line string
    that spans multiple lines
    and is indented.
```



Working Example of YAML

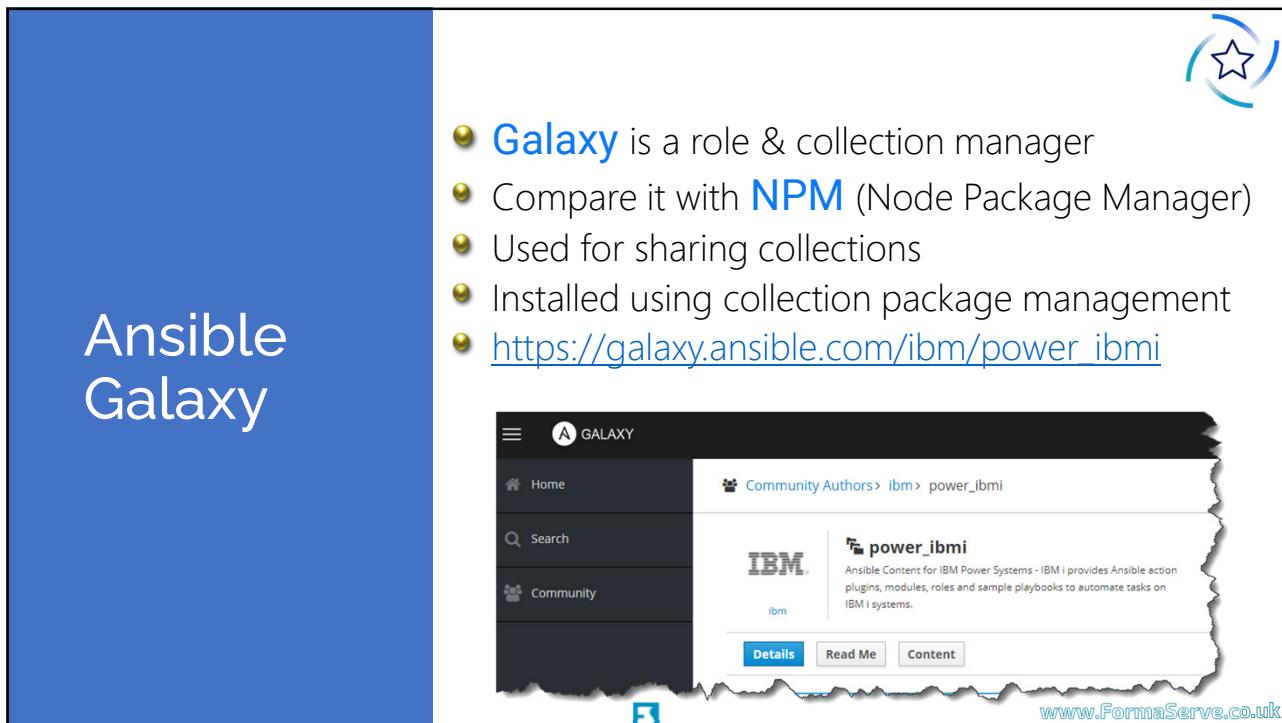
```
---
apiVersion: v1
kind: pod
metadata:
  name: pod-yaml2
  labels:
    app: yaml2
spec:
  containers:
  - name: pod-yaml2
    image: busybox
    ports:
    - containerPort: 8080
      protocol: TCP
      volumeMounts:
      - mountPath: /tmp/yaml2
        name: yaml2
```

- Key-value pairs
- metadata is an object
- labels is an object
- spec is an object
- containers is a list of objects
- ports is a list
- volumeMounts is a list of objects



The image shows the Ansible Galaxy landing page. It features a large red circle with a white letter 'A' inside. Below the circle, the word 'ANSIBLE' is written in red capital letters. In the top right corner, there is a blue circular logo with a white star and a curved arrow. On the left side, the text 'Ansible Galaxy' is displayed with a horizontal orange underline. At the bottom, there are copyright information ('© FormaServe 2022'), a Facebook icon, and a website URL ('www.FormaServe.co.uk').

53



The image shows a screenshot of the Ansible Galaxy interface. On the left, a blue sidebar contains the text 'Ansible Galaxy'. The main area displays a list of bullet points about Ansible Galaxy:

- Galaxy is a role & collection manager
- Compare it with **NPM** (Node Package Manager)
- Used for sharing collections
- Installed using collection package management
- https://galaxy.ansible.com/ibm/power_ibmi

Below the text, a screenshot of the Ansible Galaxy web interface shows the 'power_ibmi' collection details page. The page includes the IBM logo, a brief description of the collection, and buttons for 'Details', 'Read Me', and 'Content'. The URL 'www.FormaServe.co.uk' is visible at the bottom right of the screenshot.

54

Ansible Galaxy

- To install **Galaxy IBM i** collection run
● **ansible-galaxy collection install ibm.power_ibmi**
- To update use the same command as above with a **-f** flag to force an update
- Current version **1.7.1** as of **June 2022**



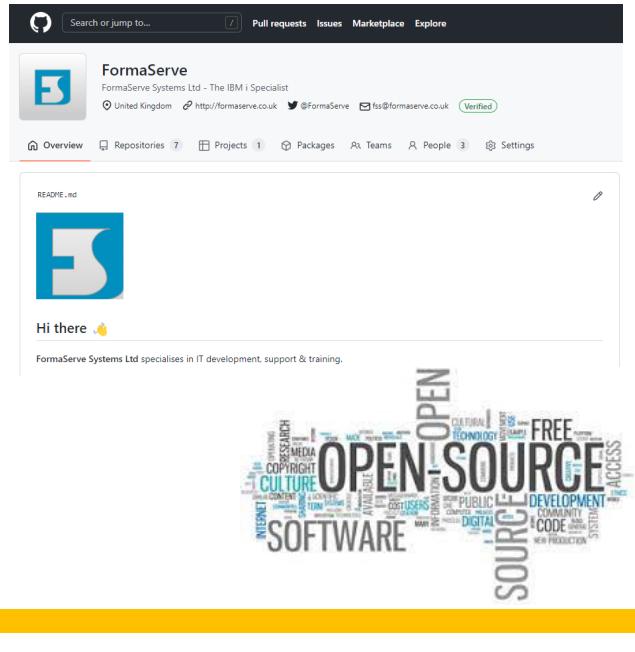


www.FormaServe.co.uk

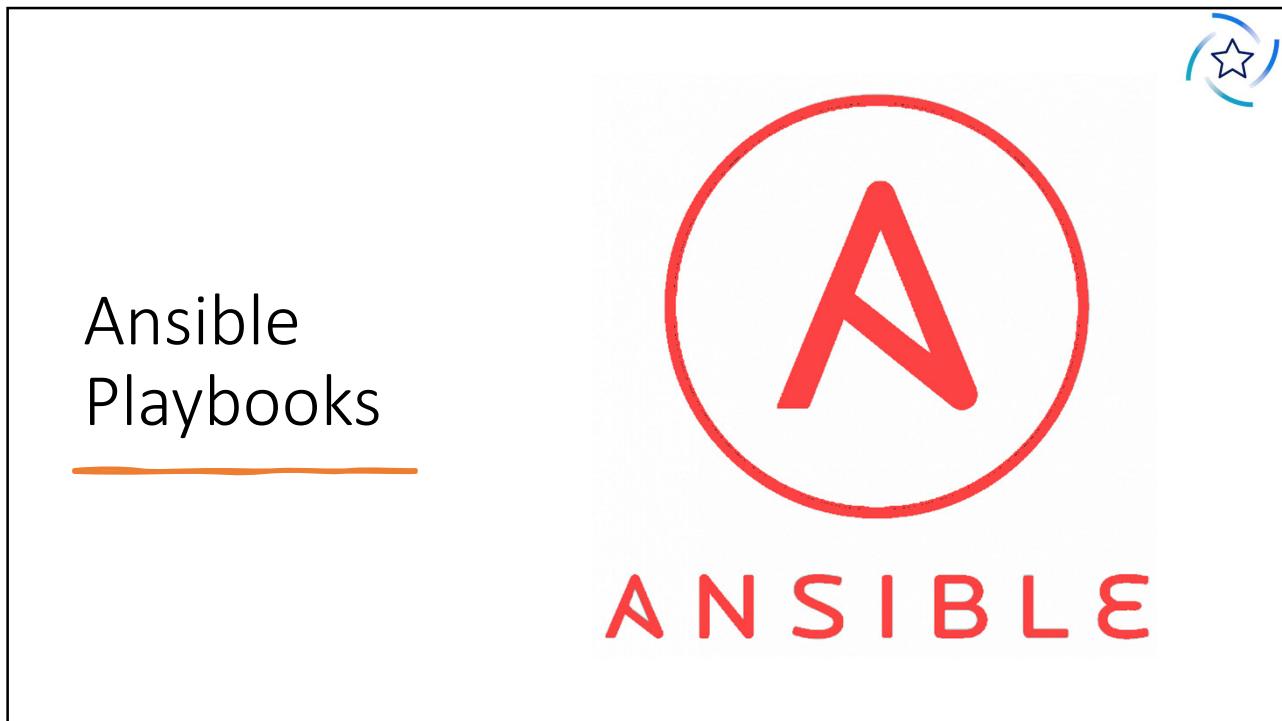
55

Ansible Galaxy

- If you want your Playbook to be shared with the world, why not upload it to Ansible Galaxy
- **Join in with Open-Source!**



56



57

A diagram illustrating the relationship between ad-hoc commands and playbooks. At the top, the text 'Ad-hoc Command' is shown above a sequence of four colored boxes: light blue (ansible), green (ibmi), yellow (-m yum), and red (-a "name=Nodejs18 state=latest"). A purple bracket groups these four items. Below this, the text 'Consider ad-hoc commands as CL commands & Playbook as a CL Program' is displayed. To the right, the word 'Playbook' is written next to a blue icon of a document. Further to the right is a code block showing YAML syntax for a playbook task:

```
---  
- name: playbook name  
hosts: ibmi  
tasks:  
- name: installing node  
  yum:  
    name: nodejs18  
    state: latest
```

© FormaServe 2022 www.FormaServe.co.uk

58



Writing Playbooks

- A playbook can have **multiple play tasks** designed to execute against **a set of hosts**
- Every playbook must contain
 - **A list of hosts to run/configure against**
 - **A list of tasks to be performed on those hosts**
- The rest are **optional**



Writing Playbooks

- The main way to automate tasks in **Ansible**
- A **YAML** based text file listing commands to run in a defined order
- Playbooks can change long, detailed admin tasks into a very easily routine with predicted & successful outcomes
- Has **.yml** extension
- Indentation with a **2 space** character - **Not TAB!**
 - Data elements at the same level must have the same indentation
 - Items that are children of another item must be indented more than their parents

A Simple Playbook



```
# Copyright © FormaServe Systems Ltd 2022
# April 2022
---
- name: Create a library on all IBMi hosts
  hosts: ibmi
  collections:
    - ibm.power_ibmi
  tasks:
    - name: Create library
      ibmi_cl_command:
        cmd: crtlib lib(AYOUENS)
      joblog: true
      register: crt_lib_result
```

1. Play name - Good doc's!
2. Which **hosts** to run on
3. Where the **collection** is stored
4. Name of the **task**
5. Which **module** to run
6. The **parameters** of the module (Create a library)
7. Display **joblog** if fails

Playbook Example



```
PATH: /QOpenSys/pkgs/bin
tasks:
- name: Verify git has been installed
  stat:
    path: /QOpenSys/pkgs/bin/git
  register: git_stat
- name: Install git if it is not there
  command: /QOpenSys/pkgs/bin/yum install git -y
  when: not git_stat.stat.exists
- name: Update git if installed
  command: /QOpenSys/pkgs/bin/yum update git -y
  when: git_stat.stat.exists
```

1. Use **Stat** module to check if git installed
Put result into **git_status**
2. If **git_status** returns not exists **install git**
3. If **git_status** returns found, **update git**



Playbook Example

```
---
  - name: Check system values on all IBMi hosts
    hosts: ibmi
    gather_facts: true
    collections:
      - ibm.power_ibmi
  tasks:
    - name: Get System Value information
      ibmi_sysval:
        sysvalue:
          - {'name': 'qmaxsgnacn', 'expect': '2'}
          - {'name': 'qccsid', 'expect': '285'}
      become_user: 'QSECOFR'
      become_user_password: '*****'

    - name: Compare the returned system values as list
      ibmi_sysval:
        sysvalue:
          - {'name': 'QATNPGM', 'expect': 'QEZMAIN QSYS'}
          - {'name': 'QATNPGM', 'expect': 'QSYS QEZMAIN'}
          - {'name': 'QATNPGM', 'expect': 'QEZMAIN QSYS', 'check': 'equal_as_list'}
          - {'name': 'QATNPGM', 'expect': 'QSYS QEZMAIN', 'check': 'equal_as_list'}
```

3 blocks!

© FormaServe 2022

F1

www.FormaServe.co.uk

63

Running Playbook

- Use the **ansible-playbook** command
- Pass the name of the playbook as an **argument/parameter**

```
andy@Norfolk:~/ansible$ ansible-playbook ibmi_library.yml
PLAY [Create a library on all IBMi hosts] ****
TASK [Gathering Facts] ****
ok: [10.0.0.36]
TASK [Create library] ****
ok: [10.0.0.36]
PLAY RECAP ****
10.0.0.36 : ok=2    changed=0   unreachable=0   failed=0   skipped=0   rescued=0   ignored=0
```

© FormaServe 2022

F1

www.FormaServe.co.uk

64



Variables in Playbooks

- **Store values** for reuse throughout your playbook
- Simplifies the creation & maintenance of a playbook
- Reduces the number of **errors**
- Provide a convenient way to manage **dynamic values**
- Examples
 - TCP servers to stop/start
 - Licensed programs to install
 - User profile details



Variables in Playbooks

- Variable names
 - Must start with a letter
 - Can only contain
 - Letters
 - Numbers
 - Underscore _
- Variable in playbooks are very similar to using variables in any programming language, CL, RPG, PHP etc.





Variables in Playbooks

- We can use variables in three forms

1. A **single line** variable declaration like we do in any programming language
 - Use **vars** to define inline variables within the playbook
2. A **separate file** with variables & values like a properties file
 - A key pair (name: 'Andy Youens')
 - Use **vars_files** to import files with variables
3. Use **include_vars** module for including variables from a set of files
 - Used at runtime for a task



Defining Variables

- In a **vars** block at the beginning of a playbook

```
---  
  - name: Create a user on all IBMi hosts  
    hosts: ibmi  
    vars:  
      user_profile: JRIDDLE  
      user_state: present  
    gather_facts: true  
    collections:  
      - ibm.power_ibmi
```



Defining Variables

- In a `vars_files` section

```
- name: Create a user on all IBMi hosts
  hosts: ibmi
  vars_files:
    - vars/user_profile.yml
  gather_facts: true
  collections:
    - jbm
```

! user_profile.yml × ! vars_file.yml ! v
 1 user_profile: JRIDDLE
 2 user_state: present
 3
 4



Defining Variables

- In a `include_vars`
- Is only activated within this step of the playbook
- Does not have to exist to run playbook, but expects it to be there when reaching that stage
- Could have been created in a previous step

```
- name: Check all ok!
  stat:
    path: /QOpenSys/pkgs/bin/git
    register: git_stat
  include_vars: vars/task2.yml
```

Defining Variables

- Example of all 3 methods we have discussed

Scoping

- Who Wins?



```

1   hosts: ibmi
2   name: Create new file on the IFS
   vars:
     | company: FormaServe
   vars_files:
     | - vars/user_profile.yml
   collections:
     | - ibm.power_ibmi

   tasks:
     - name: Create new file in test
       copy:
         dest: "vars/git.yml"
         content: |
           line git init
           line git add .
           line git commit -m Initial

     - name: Check all ok!
       stat:
         path: /QOpenSys/pkgs/bin/git
         register: git_stat
       include_vars: vars/git.yml

```

© FormaServe 2022

www.FormaServe.co.uk

71

Referencing Variables

- After declaring variables, you can use them in tasks
- Reference a variable by placing its name in double braces
 • `{{ variable_name }}`
- Ansible will substitute the variable, with its value at runtime

```

   - name: Create a user library on all IBMi hosts
     hosts: ibmi
     vars:
       user_profile: JRIDDLE
     gather_facts: true
     collections:
       - ibm.power_ibmi

     tasks:
       - name: Create library for {{ user_profile }}
         ibmi_cl_command:
           cmd: crtlib lib{{ user_profile }}
           joblog: true
         register: crt_lib_result

```

```

andy@Norfolk:~/ansible$ ansible-playbook ibmi_vars.yml
PLAY [Create a user library on all IBMi hosts] ****
TASK [Gathering Facts] ****
ok: [10.0.0.36]
TASK [Create library for JRIDDLE] **** ✓
ok: [10.0.0.36]

PLAY RECAP ****
10.0.0.36 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

© FormaServe 2022

72

Loops in Playbook - Example



```
tasks:
  - name: Create new user
    ibmi_user_and_group:
      operation: 'create'
      user: 'AEmilyY'
      become_user: 'QSECOFR'
      become_user_password: '████████'
      text: 'Created using Ansible Playbook'
      user_class: '*SECOFR'

  - name: Create another new user
    ibmi_user_and_group:
      operation: 'create'
      user: 'AJessicaJ'
      become_user: 'QSECOFR'
      become_user_password: '████████'
      text: 'Created using Ansible Playbook'
      user_class: '*SECOFR'
```



Is there a better way?

© FormaServe 2022



www.FormaServe.co.uk

73

Loops in Playbook - Example



```
- name: Create a new user on all IBMi hosts
hosts: ibmi
vars:
  user_profile:
    - AJRiddle
    - AEmilyY
    - AJessicaJ
collections:
  - ibm.power_ibmi

tasks:
  - name: Create new users
    ibmi_user_and_group:
      operation: 'create'
      user: "{{ item }}"
      password: ABC123
      user_class: '*SECOFR'
      become_user: 'QSECOFR'
      become_user_password: '████████'
      text: 'Created using Ansible Playbook'
      loop: "{{ user_profile }}"
      2
      3
```

1. A list of users in variable **user_profile**
2. Use a loop to step thro'
3. Use each item to create a new user

```
PLAY [Create a new user on all IBMi hosts] ****
TASK [Gathering Facts] ****
ok: [10.0.0.36]

TASK [Create new users] ****
changed: [10.0.0.36] => (item=AJRiddle)
changed: [10.0.0.36] => (item=AEmilyY)
changed: [10.0.0.36] => (item=AJessicaJ)

PLAY RECAP ****
10.0.0.36 : ok=2 changed=1 unreachable=0 failed=0
```

© FormaServe 2022



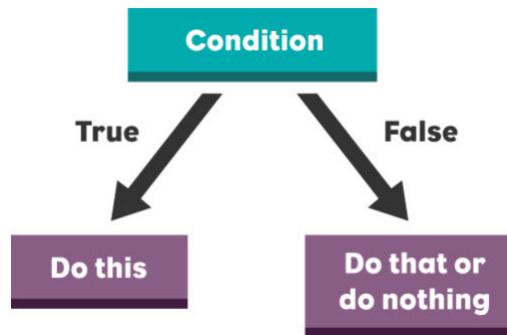
www.FormaServe.co.uk

74



Run Task Conditionally

- We can use **conditionals** to run, or not run, tasks when conditions dictate at run-time
- Both variables & facts can be examined by conditions
- A **When** statement is used to run a task conditionally



Run Task Conditionally - Operators

Operation	Symbol
Equal	<code>==</code>
Less than	<code><</code>
Greater than	<code>></code>
Less than or equal	<code><=</code>
Greater than or equal	<code>>=</code>
Not equal	<code>!</code>
Variable exists	<code>is defined</code>
Variable does not exist	<code>Is not defined</code>

Run Task Conditionally



Register

- Will capture the output of a command or task
- They only hold values while playbook is running
- The output is saved into a temporary variable that can be used
 - For debugging purposes
 - To check the status of something
- Great for checking the status of a previous command
- **Beware** - each command will return a different status!

Register Statement - Example



1. I'm using **STAT** to see if directory exists
2. Store the return from **STAT** into **git_stat**
3. If **STAT** return doesn't exists, then omit this task
4. If **STAT** return was exists, then omit this task

```

tasks:
- name: Verify git has been installed
  stat:
    path: /QOpenSys/pkgs/bin/git
  register: git_stat
  1

- name: Install git if it is not there
  command: /QOpenSys/pkgs/bin/yum install git -y
  when: not git_stat.stat.exists
  2

- name: Update git if installed
  command: /QOpenSys/pkgs/bin/yum update git -y
  when: git_stat.stat.exists
  3

  4

```





Error Handling

- Ansible **stops** executing tasks on a host when a task fails on that host
- Use **ignore_errors** to continue, ignoring the problem

```

---  
- name: Install basic packages  
hosts: ibmi  
tasks:  
  - name: install open-source  
    yum:  
      name: andy  
      state: present  
    ignore_errors: true  
  
  - name: install next  
    yum:  
      name: jane  
      state: present

```

TASK [Gathering Facts] ****
ok: [galatea]
TASK [install open-source] ****
fatal: [galatea]: FAILED! => {"ansible_facts": {"pkg_mgr": "unknown"}, "changed": ... ignoring
TASK [install next] ****
fatal: [galatea]: FAILED! => {"ansible_facts": {"pkg_mgr": "unknown"}, "changed": ... ignoring
PLAY RECAP ****
galatea : ok=2 changed=0 unreachable=0 failed=1

© FormaServe 2022



www.FormaServe.co.uk

79

Error Handling - Use -vvv Parameter



```

tasks:  
  - name: Create library  
    ibmi_cl_command:  
      cmd: crtlib lib(JYOUENS)  
      joblog: true

```

"job_log": [
 {
 "FROM_INSTRUCTION": "0D10",
 "FROM_LIBRARY": "QSYS",
 "FROM_MODULE": null,
 "FROM_PROGRAM": "QLICRLIB",
 "FROM_USER": "ANDY",
 "MESSAGE_FILE": "QCPFMMSG",
 "MESSAGE_ID": "CPF2111",
 "MESSAGE_LIBRARY": "QSYS",
 "MESSAGE_SECOND_LEVEL_TEXT": "&N Recovery . . . : Before creating
 "MESSAGE_TYPE": "ESCAPE",
 "ORDINAL_POSITION": 7,
 "SEVERITY": 40,
 "TO_INSTRUCTION": "5841",
 "TO_LIBRARY": "QSYS",
 "TO_MODULE": null,
 "TO_PROGRAM": "QCPFMMSG",
 "TYPE": "INFO",
 }
]

-vvv parameter

"MESSAGE_TEXT": "Library JYOUENS already exists.",
"MESSAGE_TIMESTAMP": "2022-05-03111:36:30.116500",
"MESSAGE_TYPE": "ESCAPE",
"ORDINAL_POSITION": 7,
"SEVERITY": 40,
"TO_INSTRUCTION": "5841",
"TO_LIBRARY": "QSYS",
"TO_MODULE": null,
"TO_PROGRAM": "QCPFMMSG",
"TYPE": "INFO",

© FormaServe 2022



www.FormaServe.co.uk

80

Sensitive Data



© FormaServe 2022



www.FormaServe.co.uk

81

Sensitive Data

- No problem, **Ansible Vault** to the rescue
- Ansible's security module
- Used for
 - **Passwords**
 - **API keys**
 - **Other secrets ...**
- Is used to manage these security details
- A crucial tool when it comes to data security & encrypting confidential information



© FormaServe 2022



www.FormaServe.co.uk

82

Sensitive Data

- ansible-vault commands
 - Create new files
 - **ansible-vault create filename**
 - View
 - **ansible-vault view filename**
 - Edit
 - **ansible-vault edit filename**
 - Encrypt existing file
 - **Ansible-vault encrypt filename**
 - Decrypt a file
 - **Ansible-vault decrypt filename**



Sensitive Data - Examples

- Change the password of an encrypted file
 - **ansible-vault rekey filename**
- Prompts for current password
- Then prompts for new password



```
andv ➤ ansible ➤ 811.573s ➤ ansible-vault rekey inventenc.ini
Vault password:
New Vault password:
Confirm New Vault password:
Rekey successful
```



Sensitive Data - Encrypt Example

```

andy@Norfolk:~/ansible$ cat secret 1
api_key: abc123

andy@Norfolk:~/ansible$ ansible-vault encrypt secret 2
New Vault password:
Confirm New Vault password:
Encryption successful
andy@Norfolk:~/ansible$ cat secret 3
$ANSIBLE_VAULT;1.1;AES256
63363836303236353830613535613935313037346337336565643630653533383963316235366462
3837393434663338316337346537326431666235653131620a353730343666613332396566646630
38333532333130663130326135336135383639653264303361626631633665653331313639383839
3065323636633635630a623832623761363139383736383562643063643962626630643635386530
38313739396135316438623763646661393466303736663131623838386537353661
andy@Norfolk:~/ansible$ 4

```

1. List file
2. Encrypt
3. Password
4. Success
5. List file



Sensitive Data - Examples

```

---
- name: Use Secret API Key
  hosts: ibmi

  tasks:
    - name: load variable from secret file
      include_vars:
        file: secret
    - name: Display encrypted variable
      debug:
        msg: "{{ api_key }}"

```

andy@Norfolk:~/ansible\$ ansible-playbook use_secret.yml

```

PLAY [Use Secret API Key] ****
TASK [Gathering Facts] ****
ok: [10.0.0.36]

TASK [load variable from secret file] ****
fatal: [10.0.0.36]: FAILED! => {"ansible_facts": {}, "ansible_included_var_files": [], "changed": false, "msg": "Attempting to decrypt but no vault secrets found"}

```



Sensitive Data - Examples

```
---
- name: Use Secret API Key
  hosts: ibmi

  tasks:
    - name: load variable from secret file
      include_vars:
        file: secret

    - name: Display encrypted variable
      debug:
        msg: "{{ api_key }}"
```

andy@Norfolk:~/ansible\$ ansible-playbook --ask-vault-pass use_secret.yml
Vault password:

```
PLAY [Use Secret API Key] ****
TASK [Gathering Facts] ****
ok: [10.0.0.36]

TASK [load variable from secret file] ****
ok: [10.0.0.36]

TASK [Display encrypted variable] ****
ok: [10.0.0.36] => {
  "msg": "abc123"
}

PLAY RECAP ****
10.0.0.36 : ok=3    changed=0   unreachable=0   failed=0   skipped=0
```

© FormaServe 2022



www.FormaServe.co.uk

87

Sensitive Data - Examples



- Don't show sensitive data in log

- Can make debugging difficult!

```
tasks:
  - name: load variable from secret file
    include_vars:
      file: secret

  - name: Display encrypted variable
    debug:
      msg: "{{ api_key }}"
      no_log: true
```

TASK [load variable from secret file] ****
ok: [10.0.0.36]

TASK [Display encrypted variable] ****
ok: [10.0.0.36]

PLAY RECAP ****
10.0.0.36 : ok=3 changed=0 unreachable=0 failed=0

© FormaServe 2022



www.FormaServe.co.uk

88



Sensitive Data - Decrypt File

- Use **ansible-vault decrypt** command
- If correct password input, file will be in the clear

```
andy@Norfolk:~/ansible$ ansible-vault decrypt secret  
Vault password:  
Decryption successful  
andy@Norfolk:~/ansible$ cat secret  
api_key: abc123
```



Sensitive Data - Configuration

- Does your **config file** contain sensitive data?
- If so, use **ansible-vault** to encrypt it!





Sensitive Data - Configuration

1. Encrypt inventory.ini
2. Enter password & encrypt
3. List password

```

andy ➤ ansible ➤ 85.861s ➤ ansible-vault encrypt inventory.ini --output inventenc.ini 1
New Vault password: 2
Confirm New Vault password: 2
Encryption successful
andy ➤ ansible ➤ 89.427s ➤ cat inventenc.ini 3
$ANSIBLE_VAULT;1.1;AES256
33353663306131666432383833396464303564336164336661633161333832633937393264643564
3239376532326366616636343061356437323430363032360a383034643464316439663666306634
65613866326435353936386363343164313431666139396530313137346339643835633938616139
3861383464303861370a373464663565376533343436356134313563303733363565613062663337
33383639366332663931336336616136383634313863303836376666376535343062326363623837

```

© FormaServe 2022



www.FormaServe.co.uk

91

Sensitive Data - Configuration

1. Run **ping** with **encrypted inventory file**
2. Will not run as no password input!
3. Use **--ask-vault-pass**
4. Command runs successfully

```

andy ➤ ansible ➤ 81.0ms ➤ ansible ibmi -m ping -i inventenc.ini 1
[WARNING]: * Failed to parse /home/andy/ansible/inventenc.ini with ini plugin: Attempting to decrypt but no vault secrets found
[WARNING]: Unable to parse /home/andy/ansible/inventenc.ini as an inventory source
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'
[WARNING]: Could not match supplied host pattern, ignoring: ibmi 2
andy ➤ ansible ➤ 81.081s ➤ ansible ibmi -m ping -i inventenc.ini --ask-vault-pass 3
Vault password: 4
galatea | SUCCESS => {
    "changed": false,
    "ping": "pong"
}

```

© FormaServe 2022



www.FormaServe.co.uk

92



93

A presentation slide titled "Ansible Top Tips". Below the title is a bullet point: "Add --step to step thro' & omit tasks". A screenshot of a terminal window shows the command "ansible-playbook ibmi_user1.yml --step". The output shows the playbook running tasks on an IBMi host. Arrows point from the text "Perform task: TASK: Gathering Facts (N)o/(y)es/(c)ontinue: y" and "Perform task: TASK: Create new user (N)o/(y)es/(c)ontinue: n" to the corresponding lines in the terminal output. The bottom of the slide features a decorative wavy pattern and includes copyright information: "© FormaServe 2022", "www.FormaServe.co.uk", and a small blue square icon.

94



Ansible Top Tips

- Use `ansible_facts` to retrieve ansible host details

```
- name: Uses shell module to retrieve domain info
  tasks:
    - name: What is our domain?
      debug:
        msg: "Domain name: {{ ansible_facts['domain'] }}"
```



```
TASK [What is our domain?] *****
ok: [localhost] => {
    "msg": "Domain name: local.formaserve.com"
}
```



Ansible Top Tips

- Validate a Playbook
 - Add the `--syntax-check` options
 - `Ansible-playbook --syntax-check ibmi_library.yml`

```
andy@Norfolk:~/ansible$ ansible-playbook --syntax-check ibmi_library.yml
ERROR! couldn't resolve module/action 'ibmi_cl_commands'. This often indicates a misspelling, missing collection, or incorrect module path.
```

```
The error appears to be in '/home/andy/ansible/ibmi_library.yml': line 10, column 7, but may
be elsewhere in the file depending on the exact syntax problem.
```

```
The offending line appears to be:
```

```
tasks:
  - name: Create library
    ^ here
andy@Norfolk:~/ansible$
```



Ansible Top Tips

- A YAML Variable **Gotcha!**
- What would the **app_path** variable be?

```
- hosts: db_servers
  vars:
    app_path: {{ ansible_config_file }}/22
```

- We need to use quotes!

```
- hosts: db_servers
  vars:
    app_path: "{{ base_path }}/22"
```

Ansible Top Tips



- **Limit Playbook Execution**
- You can limit the hosts you target on a particular run with the **--limit** flag
- **Ansible-playbook eod.yml --limit galatea**





Ansible Top Tips - Error Handling

💡 Block, rescue & always

1. Task runs okay
2. Task fails
3. Rescue block
4. Always - runs every time

```

PLAY [Show Error Handling] *****
TASK [List home directory] *****
changed: [localhost]
  1

TASK [Fail] *****
fatal: [localhost]: FAILED! => {"changed": true, "cmd": ["ls", "-l", "/tmp/does-not-exist"], "delta": "0:00:00.001820", "end": "2022-04-29 15:33:17.986334", "msg": "non-zero return code", "rc": 2, "start": "2022-04-29 15:33:17.984514", "stderr": "ls: cannot access '/tmp/does-not-exist': No such file or directory", "stderr_lines": ["ls: cannot access '/tmp/does-not-exist': No such file or directory"], "stdout": "", "stdout_lines": []}
  2

TASK [Rescue block that performs recovery] *****
ok: [localhost] => {
    "msg": "Ops, something went wrong ..."
}
  3

TASK [Run everytime] *****
ok: [localhost] => {
    "msg": "I will execute even in failure scenario"
}
  4

PLAY RECAP *****
localhost          : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=1    ignored=0
  5

```

The terminal window title is "push in ~ bash in ansible". The bottom status bar shows "andy ➜ ansible ➜ 86.36s ➜ bash 15:33:18". The URL "www.FormaServe.co.uk" is visible at the bottom right.

© FormaServe 2022



www.FormaServe.co.uk

99



Ansible Top Tips Use -vvv for Diagnostic

```

andy ➜ ansible ➜ 86.36s ➜ ansible-playbook ibmi_crt_dir.yml -vvv
ansible-playbook 2.9.6
  config file = /home/andy/.ansible/ansible.cfg
  configured module search path = ['/home/andy/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  executable location = /usr/bin/ansible-playbook
  python version = 3.8.10 (default, Mar 15 2022, 12:22:08) [GCC 9.4.0]
  Using /home/andy/.ansible/ansible.cfg as config file
  host_list declined parsing /home/andy/.ansible/inventory.ini as it did not pass its
  script declined parsing /home/andy/.ansible/inventory.ini as it did not pass its ver
  auto declined parsing /home/andy/.ansible/inventory.ini as it did not pass its verify
  yaml declined parsing /home/andy/.ansible/inventory.ini as it did not pass its verify
  Parsed /home/andy/.ansible/inventory.ini inventory source with ini plugin

PLAYBOOK: ibmi_crt_dir.yml *****
  1 plays in ibmi_crt_dir.yml

PLAY [Create Ansible environment for iUG Workshop] *****
META: ran handlers

TASK [Create home directory for user] *****
task path: /home/andy/.ansible/ibmi_crt_dir.yml:19
<10.0.0.36> ESTABLISH SSH CONNECTION FOR USER: None
<10.0.0.36> SSH: EXEC ssh -C -o ControlMaster=auto -o ControlPersist=60s -o KbdInte
ferredAuthentications=gssapi-with-mic,gssapi-keyex,hostbased,publickey -o Passwo
rdAuthentication=no -o StrictHostKeyChecking=no -o ControlPath=/home/andy/.ansible/cp/ef0647c7
  2
  3
  4
  5
  6
  7
  8
  9
  10
  11
  12
  13
  14
  15
  16
  17
  18
  19
  20
  21
  22
  23
  24
  25
  26
  27
  28
  29
  30
  31
  32
  33
  34
  35
  36
  37
  38
  39
  40
  41
  42
  43
  44
  45
  46
  47
  48
  49
  50
  51
  52
  53
  54
  55
  56
  57
  58
  59
  60
  61
  62
  63
  64
  65
  66
  67
  68
  69
  70
  71
  72
  73
  74
  75
  76
  77
  78
  79
  80
  81
  82
  83
  84
  85
  86
  87
  88
  89
  90
  91
  92
  93
  94
  95
  96
  97
  98
  99
  100
  101
  102
  103
  104
  105
  106
  107
  108
  109
  110
  111
  112
  113
  114
  115
  116
  117
  118
  119
  120
  121
  122
  123
  124
  125
  126
  127
  128
  129
  130
  131
  132
  133
  134
  135
  136
  137
  138
  139
  140
  141
  142
  143
  144
  145
  146
  147
  148
  149
  150
  151
  152
  153
  154
  155
  156
  157
  158
  159
  160
  161
  162
  163
  164
  165
  166
  167
  168
  169
  170
  171
  172
  173
  174
  175
  176
  177
  178
  179
  180
  181
  182
  183
  184
  185
  186
  187
  188
  189
  190
  191
  192
  193
  194
  195
  196
  197
  198
  199
  200
  201
  202
  203
  204
  205
  206
  207
  208
  209
  210
  211
  212
  213
  214
  215
  216
  217
  218
  219
  220
  221
  222
  223
  224
  225
  226
  227
  228
  229
  230
  231
  232
  233
  234
  235
  236
  237
  238
  239
  240
  241
  242
  243
  244
  245
  246
  247
  248
  249
  250
  251
  252
  253
  254
  255
  256
  257
  258
  259
  260
  261
  262
  263
  264
  265
  266
  267
  268
  269
  270
  271
  272
  273
  274
  275
  276
  277
  278
  279
  280
  281
  282
  283
  284
  285
  286
  287
  288
  289
  290
  291
  292
  293
  294
  295
  296
  297
  298
  299
  300
  301
  302
  303
  304
  305
  306
  307
  308
  309
  310
  311
  312
  313
  314
  315
  316
  317
  318
  319
  320
  321
  322
  323
  324
  325
  326
  327
  328
  329
  330
  331
  332
  333
  334
  335
  336
  337
  338
  339
  340
  341
  342
  343
  344
  345
  346
  347
  348
  349
  350
  351
  352
  353
  354
  355
  356
  357
  358
  359
  360
  361
  362
  363
  364
  365
  366
  367
  368
  369
  370
  371
  372
  373
  374
  375
  376
  377
  378
  379
  380
  381
  382
  383
  384
  385
  386
  387
  388
  389
  390
  391
  392
  393
  394
  395
  396
  397
  398
  399
  400
  401
  402
  403
  404
  405
  406
  407
  408
  409
  410
  411
  412
  413
  414
  415
  416
  417
  418
  419
  420
  421
  422
  423
  424
  425
  426
  427
  428
  429
  430
  431
  432
  433
  434
  435
  436
  437
  438
  439
  440
  441
  442
  443
  444
  445
  446
  447
  448
  449
  450
  451
  452
  453
  454
  455
  456
  457
  458
  459
  460
  461
  462
  463
  464
  465
  466
  467
  468
  469
  470
  471
  472
  473
  474
  475
  476
  477
  478
  479
  480
  481
  482
  483
  484
  485
  486
  487
  488
  489
  490
  491
  492
  493
  494
  495
  496
  497
  498
  499
  500
  501
  502
  503
  504
  505
  506
  507
  508
  509
  510
  511
  512
  513
  514
  515
  516
  517
  518
  519
  520
  521
  522
  523
  524
  525
  526
  527
  528
  529
  530
  531
  532
  533
  534
  535
  536
  537
  538
  539
  540
  541
  542
  543
  544
  545
  546
  547
  548
  549
  550
  551
  552
  553
  554
  555
  556
  557
  558
  559
  560
  561
  562
  563
  564
  565
  566
  567
  568
  569
  570
  571
  572
  573
  574
  575
  576
  577
  578
  579
  580
  581
  582
  583
  584
  585
  586
  587
  588
  589
  590
  591
  592
  593
  594
  595
  596
  597
  598
  599
  600
  601
  602
  603
  604
  605
  606
  607
  608
  609
  610
  611
  612
  613
  614
  615
  616
  617
  618
  619
  620
  621
  622
  623
  624
  625
  626
  627
  628
  629
  630
  631
  632
  633
  634
  635
  636
  637
  638
  639
  640
  641
  642
  643
  644
  645
  646
  647
  648
  649
  650
  651
  652
  653
  654
  655
  656
  657
  658
  659
  660
  661
  662
  663
  664
  665
  666
  667
  668
  669
  670
  671
  672
  673
  674
  675
  676
  677
  678
  679
  680
  681
  682
  683
  684
  685
  686
  687
  688
  689
  690
  691
  692
  693
  694
  695
  696
  697
  698
  699
  700
  701
  702
  703
  704
  705
  706
  707
  708
  709
  710
  711
  712
  713
  714
  715
  716
  717
  718
  719
  720
  721
  722
  723
  724
  725
  726
  727
  728
  729
  730
  731
  732
  733
  734
  735
  736
  737
  738
  739
  740
  741
  742
  743
  744
  745
  746
  747
  748
  749
  750
  751
  752
  753
  754
  755
  756
  757
  758
  759
  760
  761
  762
  763
  764
  765
  766
  767
  768
  769
  770
  771
  772
  773
  774
  775
  776
  777
  778
  779
  779
  780
  781
  782
  783
  784
  785
  786
  787
  788
  789
  790
  791
  792
  793
  794
  795
  796
  797
  798
  799
  800
  801
  802
  803
  804
  805
  806
  807
  808
  809
  8010
  8011
  8012
  8013
  8014
  8015
  8016
  8017
  8018
  8019
  8020
  8021
  8022
  8023
  8024
  8025
  8026
  8027
  8028
  8029
  8030
  8031
  8032
  8033
  8034
  8035
  8036
  8037
  8038
  8039
  8040
  8041
  8042
  8043
  8044
  8045
  8046
  8047
  8048
  8049
  80410
  80411
  80412
  80413
  80414
  80415
  80416
  80417
  80418
  80419
  80420
  80421
  80422
  80423
  80424
  80425
  80426
  80427
  80428
  80429
  80430
  80431
  80432
  80433
  80434
  80435
  80436
  80437
  80438
  80439
  80440
  80441
  80442
  80443
  80444
  80445
  80446
  80447
  80448
  80449
  80450
  80451
  80452
  80453
  80454
  80455
  80456
  80457
  80458
  80459
  80460
  80461
  80462
  80463
  80464
  80465
  80466
  80467
  80468
  80469
  80470
  80471
  80472
  80473
  80474
  80475
  80476
  80477
  80478
  80479
  80480
  80481
  80482
  80483
  80484
  80485
  80486
  80487
  80488
  80489
  80490
  80491
  80492
  80493
  80494
  80495
  80496
  80497
  80498
  80499
  804100
  804101
  804102
  804103
  804104
  804105
  804106
  804107
  804108
  804109
  804110
  804111
  804112
  804113
  804114
  804115
  804116
  804117
  804118
  804119
  804120
  804121
  804122
  804123
  804124
  804125
  804126
  804127
  804128
  804129
  804130
  804131
  804132
  804133
  804134
  804135
  804136
  804137
  804138
  804139
  804140
  804141
  804142
  804143
  804144
  804145
  804146
  804147
  804148
  804149
  804150
  804151
  804152
  804153
  804154
  804155
  804156
  804157
  804158
  804159
  804160
  804161
  804162
  804163
  804164
  804165
  804166
  804167
  804168
  804169
  804170
  804171
  804172
  804173
  804174
  804175
  804176
  804177
  804178
  804179
  804180
  804181
  804182
  804183
  804184
  804185
  804186
  804187
  804188
  804189
  804190
  804191
  804192
  804193
  804194
  804195
  804196
  804197
  804198
  804199
  804200
  804201
  804202
  804203
  804204
  804205
  804206
  804207
  804208
  804209
  804210
  804211
  804212
  804213
  804214
  804215
  804216
  804217
  804218
  804219
  804220
  804221
  804222
  804223
  804224
  804225
  804226
  804227
  804228
  804229
  804230
  804231
  804232
  804233
  804234
  804235
  804236
  804237
  804238
  804239
  804240
  804241
  804242
  804243
  804244
  804245
  804246
  804247
  804248
  804249
  804250
  804251
  804252
  804253
  804254
  804255
  804256
  804257
  804258
  804259
  804260
  804261
  804262
  804263
  804264
  804265
  804266
  804267
  804268
  804269
  804270
  804271
  804272
  804273
  804274
  804275
  804276
  804277
  804278
  804279
  804280
  804281
  804282
  804283
  804284
  804285
  804286
  804287
  804288
  804289
  804290
  804291
  804292
  804293
  804294
  804295
  804296
  804297
  804298
  804299
  804300
  804301
  804302
  804303
  804304
  804305
  804306
  804307
  804308
  804309
  804310
  804311
  804312
  804313
  804314
  804315
  804316
  804317
  804318
  804319
  804320
  804321
  804322
  804323
  804324
  804325
  804326
  804327
  804328
  804329
  804330
  804331
  804332
  804333
  804334
  804335
  804336
  804337
  804338
  804339
  804340
  804341
  804342
  804343
  804344
  804345
  804346
  804347
  804348
  804349
  804350
  804351
  804352
  804353
  804354
  804355
  804356
  804357
  804358
  804359
  804360
  804361
  804362
  804363
  804364
  804365
  804366
  804367
  804368
  804369
  804370
  804371
  804372
  804373
  804374
  804375
  804376
  804377
  804378
  804379
  804380
  804381
  804382
  804383
  804384
  804385
  804386
  804387
  804388
  804389
  804390
  804391
  804392
  804393
  804394
  804395
  804396
  804397
  804398
  804399
  804400
  804401
  804402
  804403
  804404
  804405
  804406
  804407
  804408
  804409
  804410
  804411
  804412
  804413
  804414
  804415
  804416
  804417
  804418
  804419
  804420
  804421
  804422
  804423
  804424
  804425
  804426
  804427
  804428
  804429
  804430
  804431
  804432
  804433
  804434
  804435
  804436
  804437
  804438
  804439
  804440
  804441
  804442
  804443
  804444
  804445
  804446
  804447
  804448
  804449
  804450
  804451
  804452
  804453
  804454
  804455
  804456
  804457
  804458
  804459
  804460
  804461
  804462
  804463
  804464
  804465
  804466
  804467
  804468
  804469
  804470
  804471
  804472
  804473
  804474
  804475
  804476
  804477
  804478
  804479
  804480
  804481
  804482
  804483
  804484
  804485
  804486
  804487
  804488
  804489
  804490
  804491
  804492
  804493
  804494
  804495
  804496
  804497
  804498
  804499
  804500
  804501
  804502
  804503
  804504
  804505
  804506
  804507
  804508
  804509
  804510
  804511
  804512
  804513
  804514
  804515
  804516
  804517
  804518
  804519
  804520
  804521
  804522
  804523
  804524
  804525
  804526
  804527
  804528
  804529
  804530
  804531
  804532
  804533
  804534
  804535
  804536
  804537
  804538
  
```



Ansible Top Tips - Use tags

```

tasks:
- name: Query Disk Status
  ibmi_sql_query:
    sql: "select unit_number, resource_name, protection_status from Q
  joblog: true
  register: disk_results
tags:
  - query
## to skip debug run
ansible-playbook ibmi_check_disks.yml --skip-tags debug

- name: Show disk output
  debug:
    msg: "Failed disk count: {{ disk_results.row_count }} Results: {{ disk_results }}"
tags:
  - debug
## to skip email run
ansible-playbook ibmi_check_disks.yml --skip-tags email

- name: Send email to support
  mail:
    host:
    port:
    username:
    password:
    to: Andy Youens <andy@formaserve.co.uk>
    from: FormaServe Admin<ffss@formaserve.co.uk>
    subject: "Disk Check on {{ ansible_hostname }}"
    subtype: html
    body: '<h3>Ansible disk checker.</h3>
<p>There are {{ disk_results.row_count }} disk(s) in error.
<br><br>
<p>Results are:</p>
<p>{{ disk_results.row }}</p>'
tags:
  - email
## to only run SQL
ansible-playbook ibmi_check_disks.yml --tags query

```

© FormaServe 2022



www.FormaServe.co.uk

101

Ansible Top Tips - CMDBs



- 💡 Find limitations on using static inventory files too limited?
- 💡 Take a look at [Configuration Management Databases](#)

[servicenow](#) Solutions Products Platform Customers Events About

Configuration Management Database (CMDB)

Gain visibility into your IT environment and make better decisions. Extend your CMDB with Service Grid, next-gen system of record.

[Get Data Sheet](#)

Benefits Features Resources How to Buy Related Apps

Benefits of CMDB

[ATLASSIAN](#) Products For teams Support Try now Buy now

ITSM Start your journey Incident Management More Resources

Watch keynotes, fan-favorite sessions, and in-depth product demos from our annual user conference, Team '22. [Learn more](#).

Ready for ITSM at high velocity? [Start your journey](#)

Browse topics

Service Request Management

IT Asset Management

Guide to configuration management databases (CMDBs)

According to ITIL, a configuration management database (CMDB) "is used to store configuration records throughout their lifecycle and...maintain the relationships between [them]."

© FormaServe 2022



www.FormaServe.co.uk

102

Ansible Resources

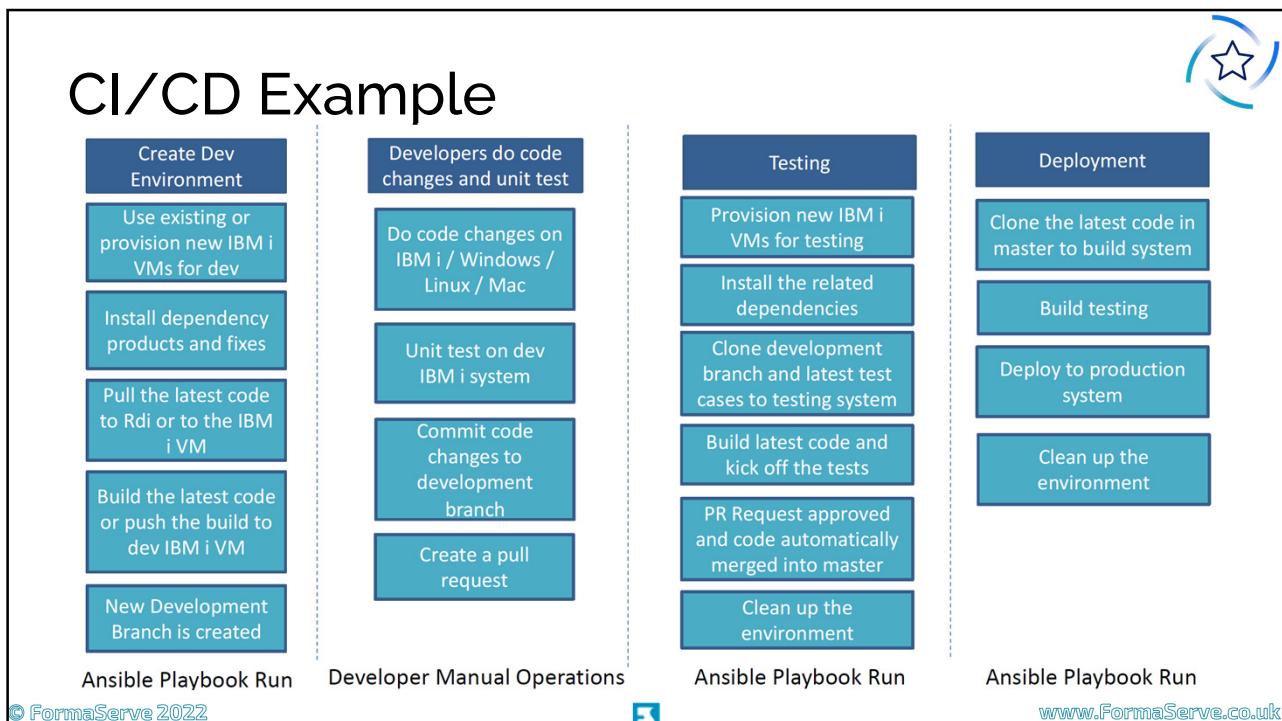


103

Resources



- Workshop examples https://github.com/FormaServe/f_Learning
- Ansible for IBM i Galaxy https://galaxy.ansible.com/ibm/power_ibmi
- Ansible for IBM i GitHub repo <https://github.com/IBM/ansible-for-i/>
- Ansible for IBM i Documentation <https://ibm.github.io/ansible-for-i/index.html>
- IBM i collections in Automation Hub
https://cloud.redhat.com/ansible/automationhub/ibm/power_ibmi
- Ansible for IBM i Articles
 - https://ibm.github.io/cloud-iblog/archivers/2020_0602_automate_your_ibm_i_tasks_with_ansible
 - <https://developer.ibm.com/tutorials/ansible-automation-for-power/>
 - <https://powerwire.eu/an-ansible-ibm-i-disk-checker>
- Blog <https://ibm.github.io/cloud-i-blog/>
- Ansible documents <https://docs.ansible.com/>
- Ansible IBM i modules <https://ibm.github.io/ansible-for-i/modules.html>



105

Anything Else?  **FormaServe** 

 A small white 3D humanoid figure stands next to a large red question mark, looking thoughtful with a hand on its chin.

 +44 (0)330 010958
 Andy@formaserve.co.uk
 www.formaserve.co.uk
 @FormaServe & @AndyYouens

 **Thank You!**

© FormaServe 2022 www.FormaServe.co.uk

106



FormaServe

Still Training!



Even in these strange times

We are still giving virtual or on-site training

It's your choice 😊

Get in touch!

Email Andy@formaserve.co.uk

<https://www.formaserve.co.uk/training.php>

<https://learning.formaserve.co.uk>

107

Legal Bumph!

- © Copyright FormaServe Systems 1990 - 2022
- All rights reserved
- All trademarks, trade names, service marks & logos referenced herein belong to their respective companies
- No unauthorised use, copying or distribution permitted

FormaServe Systems

- This presentation is for informational purposes only
- FormaServe assumes no responsibility for the accuracy or completeness of the information within

