





#CiscoLiveAPJC



# Infrastructure as Code for ACI with Ansible

Richard Watson – Technical Solutions Architect BRKDCN-1810



#CiscoLiveAPJC

# Cisco Webex App

### Questions?

Use Cisco Webex App to chat with the speaker after the session

### How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click "Join the Discussion"
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until Thursday 22 December, 2022.



https://ciscolive.ciscoevents.com/ciscolivebot/#BRKDCN-1810

### ...a bit about me





- Why automate?
- Automating ACI
- Introduction to Ansible
- Where to start?
  - Simple automation use case
  - Infrastructure-as-Code
  - CI/CD Pipelines
- Where to next?

### Common infrastructure automation outcomes





# Why ACI is the best for automation?

"Automated Cloud Infrastructure"

- Intent-based, Declarative, Abstracted, Outcome focused
- Model & API Driven
  - Everything in ACI is a Managed Object
  - Management Information Tree (MIT) tree defines relationships
    - Discovered Components
    - System Configuration
    - Status, statistics & faults
  - Admins interact with Logical Model (Intent) through API (via GUI)
  - Logical Model converted to Concrete Model (Actual) on demand







# How to automate ACI?

Your choice in how to automate your fabric

https://<ADDRESS>/api/<QUERY TYPE>/<IDENTIFIER>.<FORMAT>[?<QUERY PARAMS>]



cisco

# What is Ansible?



# ANSIBLE

- Open-source Configuration Management Tool
- Commercial support from RedHat
- Declarative (when possible) and idempotent
- Can manage a wide range of systems:
  - VMs, network, cloud etc.
- Agentless
- YAML-based
- Python server-side library dependencies
- Stateless
  - No history kept of previous results



# Installing Ansible



- Ansible is written in Python and installed as a Python package (PIP)
- Best practice to use Python virtual environment on control server
  - Allows specifying explicit Ansible version and required Python libraries
  - Separate environments may have separate versions



### Ansible Components



# Installing Collections

https://galaxy.ansible.com

- Modules used to be packaged with Ansible core files
- Unscalable, hard to maintain
- Now Vendors contribute "collections" of modules under a namespace (i.e. cisco)
- Installed from CLI

### (venv) # ansible-galaxy collection install cisco.aci



# Cisco ACI Ansible Collection Modules (Tasks)

https://docs.ansible.com/ansible/latest/collections/cisco/aci/index.html#plugin-index

- 142 tasks (v2.2.0)
  - Fabric & Tenant services
  - Shows ACI model class (where applicable)

$ \rightarrow$ C ( $$ docs.ansible.com/ansible	e/latest/collections/cisco/aci/index.html#plugin-index
A Documentation	
Collections in the Cisco Namespace Cisco.Aci Description Plugin Index	Issue Tracker Repository (Sources) Plugin Index
Cisco.Asa Cisco.Dnac Cisco.Intersight	These are the plugins in the cisco.aci collection: Modules
Cisco.los Cisco.losxr Cisco.lse	<ul> <li>aci_aaa_user module - Manage AAA users (aaa:User)</li> <li>aci_aaa_user_certificate module - Manage AAA user certificates (aaa:UserCert)</li> <li>aci_access_port_block_to_access_port module - Manage port blocks of Fabric interface poli</li> </ul>
Cisco.Niso Cisco.Niso Cisco.Niso	<ul> <li>aci_access_port_to_interface_policy_leaf_profile module - Manage Fabric interface policy le         aci_access_sub_port_block_to_access_port module - Manage sub port blocks of Fabric inter         aci_aep module - Manage attachable Access Entity Profile (AEP) objects (infra:AttEntityP, in         aci_aep_to_domain module - Bind AEPs to Physical or Virtual Domains (infra:AttEntityP)         aci_aep_to_domain module - Bind AEPs to Physical or Virtual Domains (infra:RsDomP)         aci_aet_to_menduleBind EPS to Physical or Virtual Domains (infra:RsDomP)</li> </ul>
Clisco.ucs Collections in the Cloud Namespace Collections in the Cloudscale_ch	aci_acp_coc_epg module - Manage top level Application Profile (AP) objects (fv:Ap)     aci_bd module - Manage Bridge Domains (BD) objects (fv:BD)     aci_bd_dhcp_label module - Manage DHCP Labels (dhcp:Lbl)

$\leftrightarrow$ $\rightarrow$ C $($ github.com/CiscoDevNet/ansible-ac			白 ☆ 🕫 😐 🍕 🌲 🔳 🌘
Search or jump to 7	ill requests Issues Marketplace Explore		Q +- 🚭
GiscoDevNet/ansible-aci		⊙ Wa	ntch 31 - 양 Fork 70 - ☆ Star 101 -
Code O Issues 21 1 Pull requests 1	8 🕑 Actions 🕕 Security 🗹 Insights		
🐉 master 👻 🤔 4 branches 🔊 8 ta	igs G	o to file 🛛 Add file 👻 Code 🗸	About
			Cisco ACI Ansible Collection
tomvrugt [bugfix] Fix aci_interface_:	selector_to_switch_policy_leaf_profile e 🔤 🗙 5b	9be22 4 hours ago 🕚 407 commits	
.github	[ignore] Add codeql-analysis workflow file	last month	
Changelogs	[ignore] Update Changelog for new release (v2.2.0)	7 months ago	D Readme
i meta	Add changelog, bump version and update runtime.y		
plugins	[bugfix] Fix aci_interface_selector_to_switch_polic	/_leaf_profile err 4 hours ago	
tests	[bugfix] Fix aci_interface_selector_to_switch_polic	/_leaf_profile err 4 hours ago	31 watching     70 forks
🗅 .gitignore	Add vscode and collection tarball to gitignore.	3 years ago	
CHANGELOG.rst	[ignore] Update Changelog for new release (v2.2.0)	7 months ago	Releases 8
	Add Cisco line to License file.	3 years ago	© v2.2.0 (Latest)
C README.md	Update README.md	2 years ago	
🗋 galaxy.yml	[ignore] Update Changelog for new release (v2.2.0)	7 months ago	+ 7 releases
C requirements.txt	Add a requirements file	2 years ago	
			Contributors 28
			- 🥵 😔 🌒 🧶 💿 🙆 🥥
ansible-aci			🕘 🌚 🐲 📢
			+ 17 contributors
consists of a set of modules and ro	Ansible collection for managing and automating les for performing tasks related to ACI.	your cisco aci environment. It	
This collection has been tested and	I supports ACI 3.2+. Modules supporting new fea	tures introduced in ACI API in	Languages
specific ACI versions might not be	supported in earlier ACI releases.		<ul> <li>Python 100.0%</li> </ul>
Note: This collection is not compat	ble with versions of Ansible before v2.8.		

#### https://github.com/CiscoDevNet/ansible-aci

# Where to start?

"I am rarely happier than when spending entire day programming my computer to perform automatically a task

... that it would otherwise take me a good ten seconds to do by hand."

Douglas Adams • Author of The Hitchhikers' Guide to the Galaxy

cisco il



# Where should you start?

- Not Alone!
  - Everyone starts somewhere
- Start small, learn, adapt and expand!
- Identify small, time consuming, repetitive tasks
- For example

- Configuring vPC interfaces for servers...



cisco ile

# Configuring vPC Interfaces - Part 1

### Interface Policies & Profiles

CISCO



# Configuring vPC Interfaces – Part 2

Switch Policies & Profiles







# Warning: Code Ahead

https://github.com/cisco-apjc-cloud-se/brkdcn-1810-aci-ansible

cisco live!

### Initial Ansible Playbook Version 1

- Ansible playbook to build "standard" vPCs
- Ad-hoc approach
  - Prompts user for variables at run time
- This example prompts user for variables
- 5 Sequential Tasks
  - 1. Create a Virtual Port Channel (VPC) Interface Policy Group.
  - 2. Create an Interface Leaf Profile (aka Interface Selector)
  - 3. Create an Interface Access Port Selector and bind to Interface Policy Leaf Profile
  - 4. Create an Interface Access Port Block and bind to Interface Port Selector
  - 5. Associate Interface Selector to existing Leaf Profile

– name: BRK1810 – Part 1 – Create VPC Interfaces for New Server Playbook hosts: localhost gather\_facts: false - name: server\_name prompt: Please enter new server name? private: no - name: leaf profile prompt: Please enter existing leaf switch pair profile name? private: no – name: port number prompt: Please enter leaf switch interface port number (assumes same port on both switches)? private: no - name: Create a Virtual Port Channel (VPC) Interface Policy Group cisco.aci.aci\_interface\_policy\_leaf\_policy\_group: host: 10.67.29.5 username: admin password: C!sco123 validate certs: no output\_level: debug timeout: 120 annotation: orchestrator:Ansible lag\_type: node policy\_group: "{{server\_name | capitalize }}-VPC-PolGrp" state: present # present, absent, query delegate to: localhost - name: Create an Interface Leaf Profile delegate\_to: localhost - name: Create an Interface Access Port Selector and bind to Interface Policy Leaf Profile delegate\_to: localhost - name: Create an Interface Access Port Block and bind to Interface Port Selector delegate\_to: localhost - name: Associate Interface Selector to Switch Policy Leaf Profile delegate to: localhost

#CiscoLiveAPJC BRKDCN-1810

### Initial Ansible Playbook Version 1

- Ansible playbook to build "standard" vPCs
- Ad-hoc approach
  - Prompts user for variables at run time
- This example prompts user for variables •
- 5 Sequential Tasks
  - Create a Virtual Port Channel (VPC) 1 Interface Policy Group.
  - Create an Interface Leaf Profile 2 (aka Interface Selector)
  - Create an Interface Access Port Selector 3 and bind to Interface Policy Leaf Profile
  - Create an Interface Access Port Block and 4 bind to Interface Port Selector
  - Associate Interface Selector to existing 5. Leaf Profile

	Playt	DOOK (YAML THE _Pla
tasks: - name: Create a Vi		· Group
<pre>cisco.aci.aci_int host: 10.67.29. username: admin password: C!sco validate_certs: output_level: d timeout: 120 annotation: orc lag_type: node policy_group: " # aep: whatever # link_level_po # cdp_policy: w # lldp_policy: # port_channel_ state: present delegate to: loca</pre>	<pre>interface_policy_leaf_policy_group: 5 123 no lebug inestrator:Ansible if{server_name   capitalize }}-VPC-PolGrp" rattachedentityprofile ilicy: whateverlinklevelpolicy whateverldppolicy policy: whateverlacppolicy # present_absent_query libuet</pre>	Module
- name: Create an I	interface Leaf Profile	Teels (biddes
<pre>cisco.aci.aci_int     delegate_to:_loca</pre>	errace_policy_leat_profile:=	Task (nidder
<ul> <li>name: Create an I cisco.aci.aci_acc delegate_to:_loca</li> </ul>	<pre>interface Access Port Selector and bind to ess_port_to_interface_policy_leaf_profile lhost</pre>	Interface Policy Leaf Profile
<ul> <li>name: Create an I cisco.aci.aci_acc delegate to: loca</li> </ul>	interface Access Port Block and bind to Ir ress_port_block_to_access_port:= rlhost	Task (hidder

#CiscoLiveAPJC BRKDCN-1810

# ACI Collection Modules (Tasks)

Introduction to ACI Ansible Modules

- ACI modules need credentials
  - APIC host IP, user credentials, connection information
- State parameter
  - Present, Absent or Query
- Specific module parameters (i.e. policy\_group name)
- Variables are substituted as strings
  - "{{variable\_name}}"
  - Optional Jinja filters (i.e. capitialize) to transform or standardise variables



### Initial Ansible Playbook Version 2

- Initial version 92 lines and inefficient
- Don't Repeat Yourself (DRY)
  - Same APIC details needed in each task
  - Moved to separate variable and referenced as an anchor and alias variable aci info
- Version 2 now only 71 lines

- name: BRK1810 - Part 1 - Create VPC Interfaces for New Server Playbook hosts: localhost gather\_facts: false aci\_info: &aci\_info host: 10.67.29.5 username: admin password: C!sco123 validate\_certs: no output\_level: debug timeout: 120 annotation: orchestrator:Ansible ars prompt: – name: server\_name prompt: Please enter new server name? private: no - name: leaf\_profile prompt: Please enter existing leaf switch pair profile name? private: no - name: port number prompt: Please enter leaf switch interface port number (assumes same port on both switches)? private: no - name: Create a Virtual Port Channel (VPC) Interface Policy Group <<: \*aci\_info lag type: node policy\_group: "{{server\_name | capitalize }}-VPC-PolGrp" state: present # present, absent, query delegate to: localhost - name: Create an Interface Leaf Profile delegate to: localhost - name: Create an Interface Access Port Selector and bind to Interface Policy Leaf Profile cisco.aci.aci access port to interface policy leaf profile: -delegate to: localhost - name: Create an Interface Access Port Block and bind to Interface Port Selector cisco.aci.aci access port block to access port: delegate to: localhost - name: Associate Interface Selector to Switch Policy Leaf Profile delegate\_to: localhost 26 BRKDCN-1810

#Ciscol iveAPJC

### Initial Ansible Playbook Version 3

- Including passwords is risky and bad practice!
  - Could prompt for password
- Create encrypted password using Ansible Vault
  - Note: Must store in separate file for anchor to work.



Confirm New Vault password:

\$ANSIBLE\_VAULT;1.1;AES256

New Vault password:

enc\_password: !vault |

Encryption successful

# Running an ACI Playbook

- Command to run our playbooks **ansible-playbook** 
  - ansible-playbook -i {inventory file} {Playbook file}
  - ansible-playbook add\_new\_server\_playbook-v3.yaml --ask-vault-pass
- Check mode (--check)
  - Run through playbook without making changes
  - ansible-playbook -i {inventory file} {Playbook file} --check
- Verbose (-v, -vvv)
  - Verbose output good for troubleshooting/debugging
  - ansible-playbook -i {inventory file} {Playbook file} -vvvv



cisco live!

#CiscoLiveAPJC BRKDCN-1810

# Infrastructure-as-Code (laC)

cisco live!

# Infrastructure-as-Code (IaC)



Infrastructure as Code (IaC) is the managing and provisioning of infrastructure through code instead of through manual processes.

- Really should be Intent-as-Code
- Code split into *intent* (variables) and *implementation logic* (playbooks/tasks)
- Managed through code repository tools (i.e. GitHub/GitLab/BitBucket etc.)
  - Sharing/Collaboration
  - Rollback/Versioning



### Infrastructure as Code is a process, not a single tool or application

# Managing ACI Switch Port Configuration with IaC

- Code re-worked to split *intent* from *implementation logic*
- For efficiency, use nested loops of tasks
- Intent (source of truth) abstracted input variable file
  - Cluster -> Server -> Switch Ports
- State included for removing configuration
- Easily add (or remove) interfaces/servers/clusters...





# Input Data (Business Intent)

cluster\_details.yaml

- Custom YAML file
  - Written to work for you...
  - Captures business intent as data
- Simplifies Day 2 changes
  - Add a Server -> Add a new section
  - Remove a Server -> Set state to Absent, run the playbook - then remove code itself
  - Move a Server -> Add a 2<sup>nd</sup> Port block, then remove the 1<sup>st</sup>

	- name: MELDC1ESX1	
	description: Production VMware ESXi Cluster	
	- name: MELDC1ESX1-1	
	description: MELDC1ESX1-1 Production VMware ESXi Server	
	state: present	
	switch_ports:	
	- name: INBAND-MGT	
	state: absent	
	leaf_profile: L101-L102	
	description: Inband Mgmt Uplink Port-Channel	
	lag_type: node # Required for VPC - assumes same port number on both	
	interface type: vpc	
	policy group:	
	port blocks:	
	- name: E1-5	
	state: present	
	start: 5	
	stop: 5	
	- name: VM-DATA	
	state: absent	
	leaf profile: L101-L102	
	description: VM Data Uplink Port-Channel	
	lag_type: node # Required for VPC - assumes same port number on both	
	interface_type: vpc	
	policy_group:	
	port_blocks:	
	- name: E1-6	
	state: present	
	- name: 00B	
	state: present	
	leaf_profile: L101	
	description: Server OOB Management Port	
	<pre>lag_type: leaf # Single 00B management port</pre>	
	interface_type: switch_port	
	port_blocks:	
	- name: E1-31	
	state: present	
	start: 31	
	- name: MELDC1ESX1-2 -	
	– name: MELDC1ESX1-3 ∞	
	- name: MELDC1ESX2	
	description: Development VMware ESXi Cluster	
	servers:	
	- name: MELDC1ESX2-1⇔	
214 >	- name: MELDC1ESX2-2 =	

#### iac\_cluster\_management\_playbook.yaml

<pre></pre>	
loop_var: cluster	
manage_cluster.yaml	
<pre>1 ### Iterate through each clusters' server definitions ### 2 - name: "Manage Cluster {{cluster.name   upper }} Servers" 3 include_tasks: manage_server.yaml 4 loop: "{{cluster.servers}}" 5 loop_control: 6 loop_var: server 7</pre>	
manage_server.yaml	
<pre>1 ### Iterate through each servers' switch port definitions ### 2 - name: "Manage {{server.name   upper }} Switch Ports" 3 include_tasks: manage_switch_port.yaml 4 loop: "{{server.switch_ports}}" 5 loop_control: 6 loop_var: switch_port 7</pre>	
manage switch port.vaml	

– name: Set APIC Credential Information		
aci_info: &aci_info		
host: 10.67.29.5		
username: admin		
<pre>password: "{{vault_password}}"</pre>		
output_level: debug		
timeout: 120		
annotation: orchestrator:Ansible		
<ul> <li>name: Check Server state before creating s</li> </ul>	witch ports	
<pre>switch_port_state: "{% if server.state==</pre>	'absent' %}absent{% else %}{{switch_	port.state}}{% endif %}"
– name: "Manage {{server.name upper}} {{swit	ch_port.name upper}} Interface Polic	y Group "
<<: *aci_info		
<pre>policy_group: "{{server.name   upper }}-</pre>	-{{switch_port.name   upper }}-POL-GP	P"
<pre>lag_type: "{{switch_port.lag_type}}"</pre>		
<pre>description: "Policy group for {{server.</pre>	<pre>name   upper }} interface group {{sw</pre>	/itch_port.name   upper }}"
<pre>state: "{{switch_port_state}}"</pre>		
<pre>aep: "{{item.aep   default(omit) }}"</pre>		
link_level_policy: "{{item.link_level_po	licy   default(omit) }}"	
<pre>cdp_policy: "{{item.cdp_policy   default</pre>	(omit) }}"	
<pre>lldp_policy: "{{item.lldp_policy   defau</pre>	<pre>lt(omit) }}"</pre>	
<pre>port_channel_policy: "{{item.port_channe</pre>	l_policy   default(omit) }}"	
delegate_to: localhost		
<pre>- name: "Manage {{server.name upper}} {{swit</pre>	ch_port.name upper}} Interface Selec	tor/Leaf Profile"
> cisco.aci.aci_interface_policy_leaf_profil		
delegate_to: localhost		
<pre>- name: "Manage {{server.name upper}} {{swit</pre>	ch_port.name upper}} Interface Acces	s Port Selector and bind to Interface Selector"
<pre>&gt; cisco.aci.aci_access_port_to_interface_pol</pre>	icy_leaf_profile:∞	
delegate_to: localhost		
– name: "Manage {{server.name upper}} {{swit	ch_port.name upper}} Access Port Blo	ocks and bind to Port Selector"
cisco.aci.aci_access_port_block_to_access_	port:	
<: *ac1_1nto		
interface_profile: "{{server.name   uppe	r }}-{{switch_port.name   upper }}-1	NIF-SEL"
access_port_selector: "{{server.name   u	<pre>pper }}-{{switcn_port.name   upper }</pre>	
port_blk: "{{port_block.name   upper }}"		
<pre>trom_port: "{{port_block.start}}" to orthogonal #///port_block.start}"</pre>		
to_port: "{{port_block.stop}}"		
### Set port_block.state:absent if switc	n_port absent ##	
state: "(% if switch_port_state== absent	**/absenti* else */{{port_block.sta	ite}}{% endir %}
delegate_to: localnost		
loop: "{{switcn_port.port_blocks}}"		
loop_control:		
	ch port paralupporll Interfere Cales	tor to Loof Switch Profile Accessition"
- name: manage ((server.name upper)) {{swit	<pre>cn_port.name(upper); Interface Selec</pre>	tor to Lear Switch Profile Association"
cisco.aci.aci_interface_selector_to_switch	_policy_lear_profile :	
<: *acl_into		
intenfore colorter, "((source to the second se	er)) (fouiteb next next 1	
interface_selector: {{server.name   upp	er //port.name   upper }}-	INTI-SEL
state: "Jewitch port state		
<pre>state: "{{switch_port_state}}" delegate to: localbost</pre>		

cisco live!

#CiscoLiveAPJC BRKDCN-1810

© 2022 Cisco and/or its affiliates. All rights reserved. Cisco Public 34

# Problems

- What if User 1 forgets to push changes back?
- What if two users run at the same time?
- Only run the playbook
   AFTER updates have been pushed to repository
  - Ideally to run from a common control server





# GitOps with Ansible AWX / Automation Platform (Tower)

### **Basic Automation Pipeline**

- AWX/Tower are RedHat Ansible "managed" control servers
  - AWX is OSS, Tower/Automation Platform is paid
  - Installed on Kubernetes
  - GUI & API driven
  - RBAC, Audit trails etc.

- Ansible collections & python libraries
   automatically downloaded
- GitOps
  - Git Push triggers webhook
  - AWX/Tower pulls code & executes
  - Multiple Jobs sequentially





## AWX – Project Definition

E							🜲 🕜 👻 🛓 admin	•
	Jobs P Schedules Activity Stream	rojects > BRKDCN-1810-Part Details	2-laC					5
	Workflow Approvals		tails Access Job Templates	Notifications Sche	edules			
R	Resources 🗸	Last Job Status	Successful	Name	BRKDCN-1810-Part2-laC	Description	BRKDCN Introduction to Infrastructure as Code with ACI and Ansible	
	Credentials	Organization	Default	Source Control Type	Git	Source Control Revision	0d8131f 🏢	
	Projects Inventories Hosts	Source Control URL 🔊	https://wwwin-github.cisco.com /richwats/brkdcn-1810-ansible- aci-part2.git	Cache Timeout	0 Seconds	Project Base Path 💿	/var/lib/awx/projects	
A	Access 🗸	Playbook Directory 😨 Enabled Options	_8_brkdcn_1810_part2_iac	Created	06/10/2022, 2:27:22 pm by admin	Last Modified	17/10/2022, 12:40:36 pm by adm	'n
	Organizations	Edit Sync [	Delete		Email 💌	Q Add Delete		1-
Back to Projects     Det	tails Access Job Templates Notifications	Schedules			Username 1	First Name 1	Last Name 🗍	Role
Name 🔻	Q			1-1of1 🔻 < >	🗆 admin			System Administrator
Name 1		Туре	Options		🗆 anmangla	Anshul	Mangla	System Administrator
BRKDCN-1810-Email-Ap	oproval-Notification	Email	💽 Start 🂽	Success 💽 Failure				
			1-1of1items 👻 < <	1 of1page > >>	richwats	Richard	Watson	System Administrator
cisco	ive!		#	CiscoLiveAPJC E	3RKDCN-1810 © 20	022 Cisco and/or its affilia	ites. All rights reserved. Cisco	1-3 of 3 items + « < 1 Public 37

### AWX – Job Template

							🜲 😧 👻 💄 admin 👻
Jobs Schedules	1	Templates > BRKDCN-1810-Part2 Details	-Manage-Cluster-Interfaces				Ŀ
Workflow Approvals		Back to Templates     Deta	ails Access Notifications S	Schedules Jobs Si	ırvey		
Resources	~	Name	BRKDCN-1810-Part2-Manage- Cluster-Interfaces	Description	Manage server cluster interfaces on ACI fabric	Job Type 🔊	run
Templates Credentials		Organization	Default	Inventory 💿	Demo Inventory	Project 💿	BRKDCN-1810-Part2-laC
Projects		Execution Environment 🔊	AWX EE (latest)	Playbook 💿	iac_cluster_management_playbook .yaml	Fork: ⑦	0
Inventories Hosts		Verbosity 💿	0 (Normal)	Timeout 🔊	0	Show Changes 🔊	Off
Access	•	Job Slicing 🗇	1	Webhook Service 🔊	GitHub	Webhook URL 😨	http://awx.mel.ciscolabs.com /api/v2/job_templates/9/github/
Organizations		Webhook Key 💿	krU3VYVGZQqeDkmjld5LcfUdfDl gRZNaNzhUlmw6C0xF6segG4	Created	06/10/2022, 2:46:56 pm by admin	Last Modified	06/10/2022, 4:34:19 pm by admin
Users		Enabled Options 💿	Webhooks				
		Credentials 💿	Vault: BRKDCN-1810				
Administration	~	Variables <sup>®</sup> YAML JSON	]				×

### GitHub - Webhook Configuration

C Enterprise Search or jump to	¢ +• ⊜•		
EngIT: GitHub Upgrade Friday November 11th			
📮 richwats / brkdcn-1810-ansil	ble-aci-part2 Public	01	Jnwatch - 1 ☆ Star 0 약 Fork 0
🗘 Code 🕢 Issues 🛛 Pull reque	ests 凹 Projects 🕮 Wiki 🕕 Secu	urity └─ Insights   \$\$ Settings	
	Options	Webhooks / Manage webhook	
	Collaborators	We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd li	ke
	Security & analysis	to receive (JSON, x-ww-form-urlencoded, etc). More information can be found in our developer documentation.	
	Branches	Payload URL *	
	Hooks	http://awx.mel.ciscolabs.com/api/v2/job_templates/9/github/	
	Notifications	Content type application/json	
	Integrations	Secret	
	Deploy keys	If you've lost or forgotten this secret, you can change it, but be aware that any integrations using this secret will need to be	
	Custom tabs	updated. – Change Secret	
	Autolink references	Which events would you like to trigger this webhook?	
		• Just the push event.	
		O Send me everything.	
		O Let me select individual events.	
		Active We will deliver event details when this hook is triggered.	
		Update webhook Delete webhook	

39

🗰 Atom File Edit View	Selection Find Pack						0 4			* ◎ 豊 令 Q	🔚 🔸 Mon 17 Oct 1	
richwats/brkdcn-181	0-ansible-a 🗙 🔤 AWX	Jobs	× 😁 APIC (10.67.2	9.5) × +								
	🔿 🔒 awx.mel.c	iscolabs.cc	om/#/jobs						11	0% ☆		
										. 0 -	💄 admin 🔻	
Jobs Schedules	Job	S									Ð	
Activity Stream Workflow Approvals			Name 🔻	Q	Delete	Cancel jo	bbs			1-20 of 33 👻	< <b>&gt;</b>	
Resources	~		Name		State	us 🕽	Туре	Start Time	Finis	h Time ↓	Actions	
Templates Credentials Projects			34 – BRKDCN-1810- es	Part2-Manage-Cluster-Into	erfac 💽 S	Successful	Playbook Run	17/10/2022, 12:57:28 pn	n 17/10 aci-ansible	)/2022, 12:59:47 pm	ą	
Inventories Hosts			35 – BRKDCN-181C <sup>&gt;</sup> ♥	Project	cluste 1 ## Se 2 clust 3 - n 4 d	er_details.yaml rver Cluste ers: ame: MELDC1 escription:	er Details ## ESX1 Production VMware F	SXi Cluster				ר <b>℃</b> Stage All
Access Organizations Users	•	•	32 – BRKDCN-1810 es	<ul> <li></li></ul>		ervers: – name: ME descript state: p switch_p – name	LDC1ESX1-1 ion: MELDC1ESX1-1 Pr iresent iorts: : INBAND-MGT	oduction VMware ESXi Server		cluster_details.yar	<b>1</b> ↓ Uns hanges	
Teams Administration	•		33 – BRKDCN-1810	<ul> <li>iac_cluster_management_play</li> <li>manage_clusteryaml</li> <li>manage_serveryaml</li> <li>manage_switch_port.yaml</li> <li>BRADME md</li> </ul>		stat leaf desc lag_ inte	e: present profile: L101-L102 ription: Inband Mgm type: node # Require erface_type: vpc	<b>t Uplink Port-Channel</b> d for VPC – assumes same port		See All Sta	ged Changes ster switch port	
Credential Types Notifications	,		31 – BRKDCN-1810				<pre>.cy_group: aep: whateverattache link_level_policy: w cdp_policy: whatever lldp_policy: whateve</pre>			2+		н
Management Jobs Instance Groups Instances	3		29 – BRKDCN-1810 es				<pre>port_channel_policy: _blocks: name: E1-5 state: present start: 5</pre>			Find the commit to commit	Unc	38 do 1h 1h 2h
				luster_details.yaml ③ 0 🛦 0 ③ 0					-8 2 Spaces	YAML ∲ master OF	etch 🎧 GitHub ≺	¢- Git (1)

cisco live!



CISCO

	AWX   38 - BRKDCN-1810-1	art2 × (* APIC (10.87.29.5) × +		
← → C ŵ	O 👌 awx.mel.ciscolabs.com/#/jo		110% ☆	
			<b>≜ 0 -</b>	💄 admin 👻
Jobs Schedules Activity Stream Workflow Approvals	Jobs > 38 - BRKDCN Output	1810-Part2-Manage-Cluster-Interfaces Details Output		G
Resources Templates Credentials Projects Inventories Hosts	BRKDCN-1810-P	ert2-Manage-Cluster-Interfaces Successful Plays 1 Tasks 135 Hosts 1 Elap ssing complete. Reload output DMEONE "accidentally" removes all interface		± =
Access Organizations Users Teams	<ul> <li>412 ok:</li> <li>413</li> <li>414 TASK</li> <li>415 ok:</li> <li>416</li> <li>417 TASK</li> <li>418 ok:</li> </ul>	<pre>[localhost] [Manage MELDC1ESX2-3 00B Interface Selector/Leaf Profile] ************************************</pre>		
Administration Credential Types Notifications Management Jobs Instance Groups Instances	419 420 TASK 421 ok: 422 423 TASK 424 ok: 425 426 PLAY 427 loca	<pre>[Manage MELDC1ESX2-3 00B Access Port Blocks and bind to Port Selector] **** 14:26:17 [localhost] =&gt; (item={'name': 'E1-33', 'state': 'present', 'start': 33, 'stop': 33}) [Manage MELDC1ESX2-3 00B Interface Selector to Leaf Switch Profile Association] *** 14:26:18 [localhost] RECAP ************************************</pre>	red=0	1

# Continuous Integration / Continuous Deployment (CI/CD) Pipelines

*"To err is human, but to really foul things up you need a computer. "* 

Paul R. Ehrlich

cisco ile

# Continuous Integration / Continuous Deployment (CI/CD) Pipelines

### Software Development Lifecycle

- Move fast, make changes quickly BUT always test and get approval (if needed) before deploying
- Known as a Continuous Integration / Continuous Deployment (CI/CD) pipeline
- May require a separate toolset (i.e. Jenkins/Flux/Argo)
  - AWX/Tower, GitHub, GitLab also include some native capabilities
- How to best adapt for Infrastructureas-Code?
  - What testing can we run?







# Where to next?

.

.

cisco live!

# How to learn more?

DevNet - developer.cisco.com

- All developer / programmability resources are in one place
- Links to documentation, interactive labs & shared code
- <u>https://developer.cisco.com/iac/</u>
- <u>https://developer.cisco.com/automation-ansible/</u>
- ACI Simulator OVA



cisco / ille

# Cisco DevNet Automation Bootcamp for ACI

https://developer.cisco.com/automation-bootcamp/

- Self-study prerequisite materials
- 9 days of focused training delivered over 8 weeks
- 5-day virtual instructor-led training and labs that focus on the foundational concepts
- Dynamic, hands-on, 4-day virtual instructor-led, deep-dive labs that build your confidence and skills

49

Three 2-hour office hours support sessions conducted using a Webex space



# Summary

- 1. Visit DevNet to learn through hands on labs! *(developer.cisco.com)*
- 2. Identify small, repetitive tasks for quick wins
- 3. Slowly move towards Infrastructure-as-Code (or Intent-as-Code) operations
- 4. Adopt CI/CD pipelines for greater control over IaC
- 5. Talk to your Cisco Partner or CX to help\*

\* Thursday 12:30-12:45 - CXLTS-1031 Accelerate Automation with Nexus-as-Code for ACI

cisco

# Session Surveys

We would love to know your feedback on this session!

- Complete the session surveys in the Cisco Events mobile app. You'll earn some points in the Cisco Live Game and potentially win a prize.
- Complete a minimum of four session and the overall event surveys to claim a Cisco Live cable bag.

## Continue your education

Visit the Cisco Showcase for related demos



Book your one-on-one Meet the Expert meeting



Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs



Visit the On-Demand Library for more sessions at <u>www.CiscoLive.com/on-demand</u>





# **Cisco Learning and Certifications**

From technology training and team development to Cisco certifications and learning plans, let us help you empower your business and career. www.cisco.com/go/certs

#### Pay for Learning with Cisco Learning Credits

(CLCs) are prepaid training vouchers redeemed directly with Cisco.



**Cisco U.** IT learning hub that guides teams and learners toward their goals

#### **Cisco Digital Learning**

Subscription-based product, technology, and certification training

#### **Cisco Modeling Labs**

Network simulation platform for design, testing, and troubleshooting

#### **Cisco Learning Network**

Resource community portal for certifications and learning



**Cisco Training Bootcamps** Intensive team & individual automation and technology training programs

#### **Cisco Learning Partner Program**

Authorized training partners supporting Cisco technology and career certifications

#### Cisco Instructor-led and Virtual Instructor-led training

Accelerated curriculum of product, technology, and certification courses

<u>ଜ</u>= Certify

#### Cisco Certifications and Specialist Certifications

Award-winning certification program empowers students and IT Professionals to advance their technical careers

#### **Cisco Guided Study Groups**

180-day certification prep program with learning and support

### **Cisco Continuing**

Education Program

Recertification training options for Cisco certified individuals

CISCO The bridge to possible

# Thank you

cisco life!

#CiscoLiveAPJC







#CiscoLiveAPJC