

The background features a vibrant, multi-colored abstract design. On the left, there are horizontal, wavy bands of color in shades of red, orange, yellow, and green. On the right, a bright white light source emits a series of colorful rays in shades of blue, green, and yellow, creating a sunburst effect.

cisco *Live!*

Let's go

#CiscoLiveAPJC



The bridge to possible

Deploy Cisco Catalyst Wireless

as an Infrastructure as a Service (IaaS) with AWS

Mark Rebetzke
Technical Solutions Architect – Enterprise Networking
BRKEWN-1104

CISCO *Live!*

#CiscoLiveAPJC

Cisco Webex App

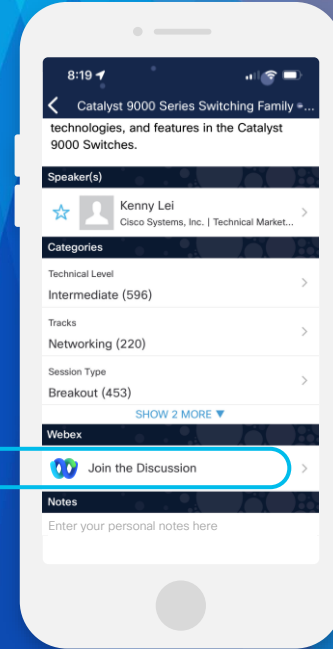
Questions?

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 June 9, 2023.



<https://cislive.ciscoevents.com/cislivebot/#BRKEWN-1104>

About Myself





How is the **network** seen
today?

Under Valued

Do more with Less.

What **band member** would
the network be?



If you're seen, you're
Doing it wrong.

The bassist is the most
Important member in the
band?

You can't hear it, but you know
when its **not there**?

What do **we need** as network
engineers?

#1 Easy to manage

Automated Low Risk Networks

#2 Scalable

Allow to go with the business

#3 Seamless

Transparent to the everyday user, while keeping them secure



The bridge to possible

Deploy Cisco Catalyst Wireless

as an Infrastructure as a Service (IaaS) with AWS

Mark Rebetzke

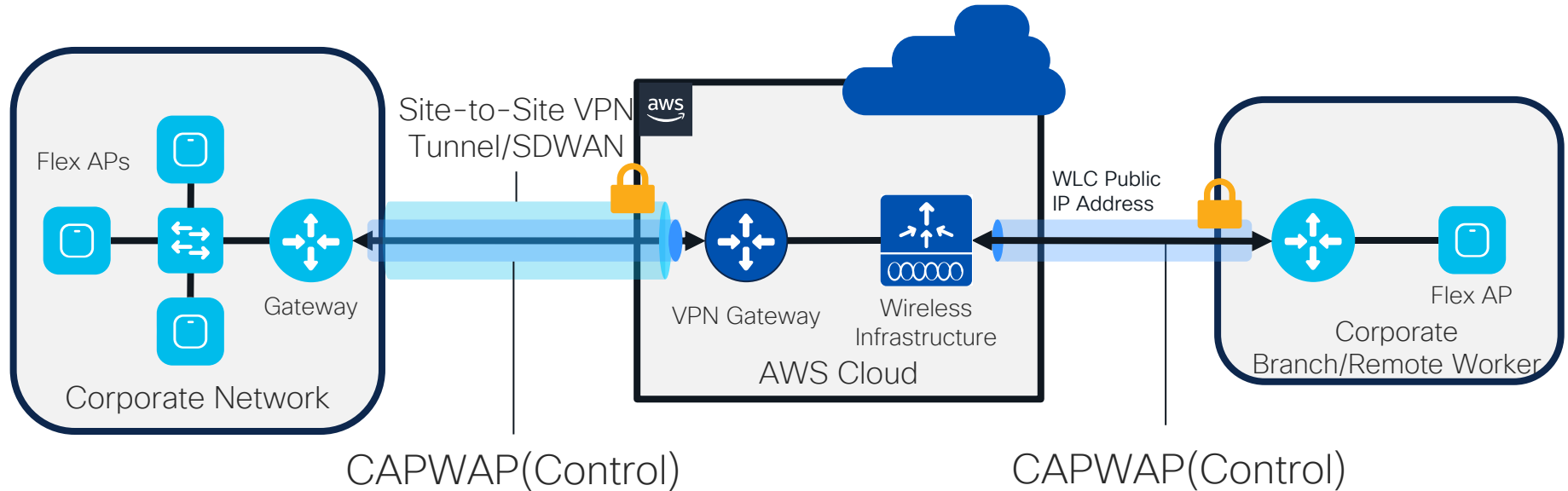
Technical Solutions Architect – Enterprise Networking

BRKEWN-1104



#CiscoLive

Cisco Wireless Infrastructure in AWS



Agenda

- Why are we doing this doing?
- Terraform, and Ansible 101
- How to Deploy Cisco Wireless laaS on AWS
 - Network
 - Security
 - Infrastructure
 - Connection

AWS only supports FlexConnect Deployment C9800-CL

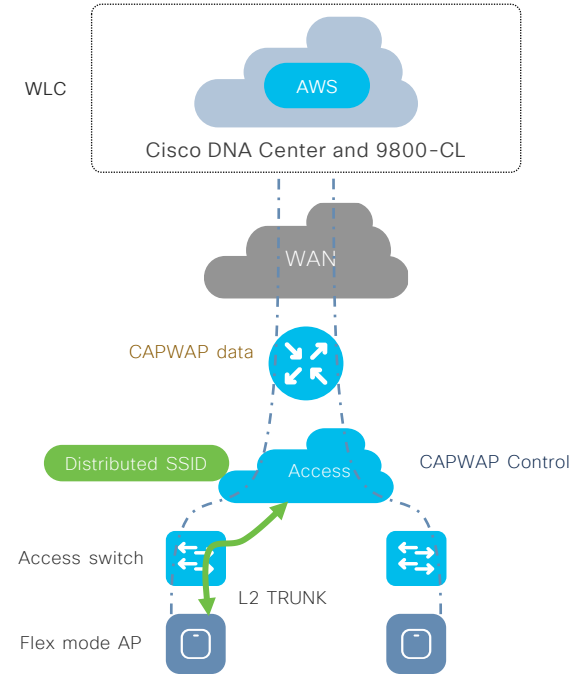
Why Flex Deployment on AWS?

Survivability in Branch Offices

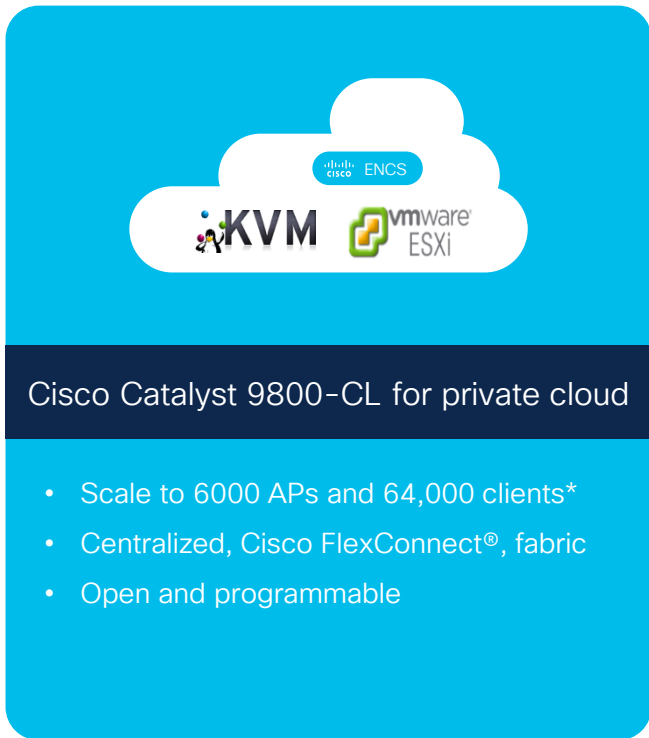
Lower single point of failure

Simple to Deploy

Central Control | Distributed Data (802.1Q)



Cisco Catalyst 9800 Series for private and public cloud



The image shows a blue rounded rectangle representing a private cloud environment. At the top, there is a white cloud icon containing the Cisco ENCS logo, the KVM logo, and the VMware ESXi logo. Below the icon, a dark blue horizontal bar contains the text "Cisco Catalyst 9800-CL for private cloud". At the bottom, a light blue section contains a bulleted list of features.

Cisco Catalyst 9800-CL for private cloud

- Scale to 6000 APs and 64,000 clients*
- Centralized, Cisco FlexConnect®, fabric
- Open and programmable



The image shows a green rounded rectangle representing a public cloud environment. At the top, there is a white cloud icon containing the AWS logo. Below the icon, a dark green horizontal bar contains the text "Cisco Catalyst 9800-CL for public cloud". At the bottom, a light green section contains a bulleted list of features.

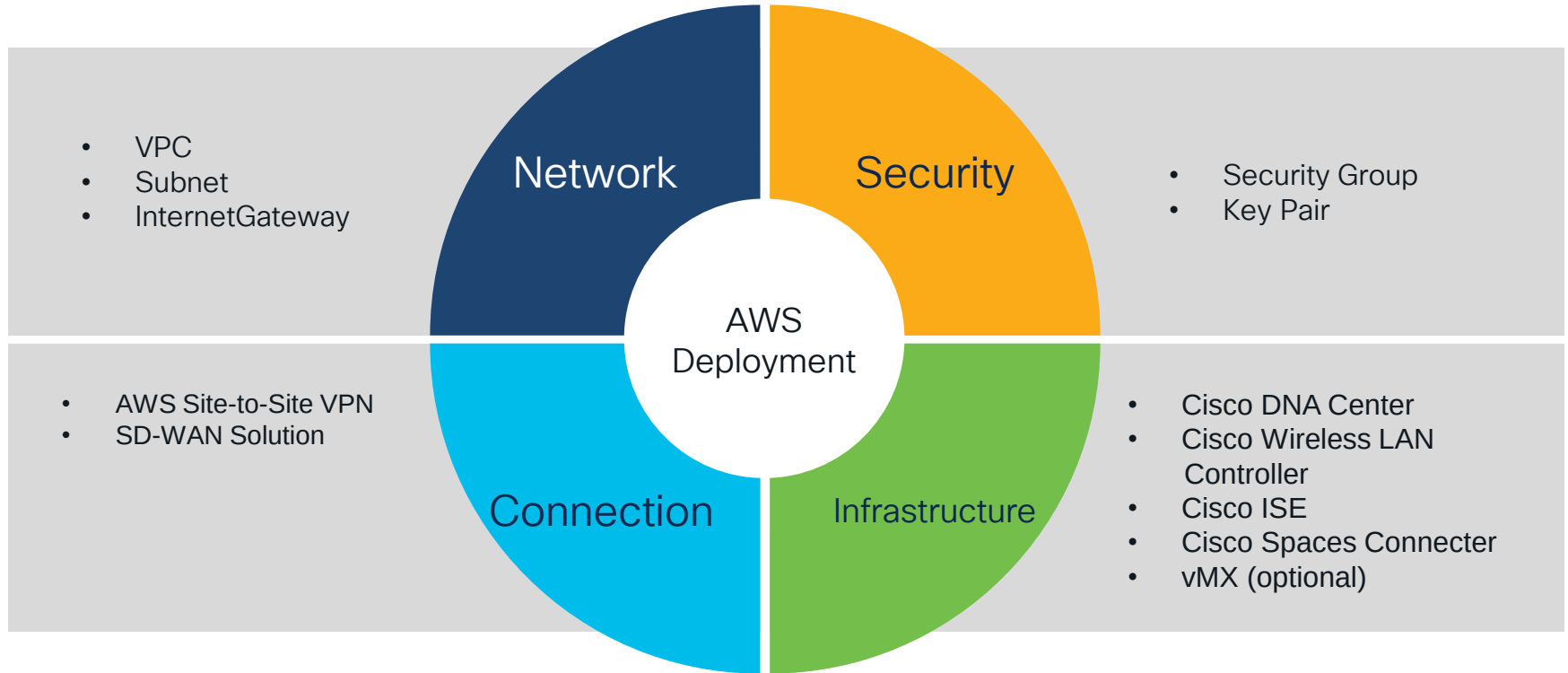
Cisco Catalyst 9800-CL for public cloud

- Scale to 1000 APs and 10,000 clients
- Cisco FlexConnect local switching
- Open and programmable

*Only with Cisco FlexConnect and fabric mode for 6000 AP support

How to Deploy?

Things We need



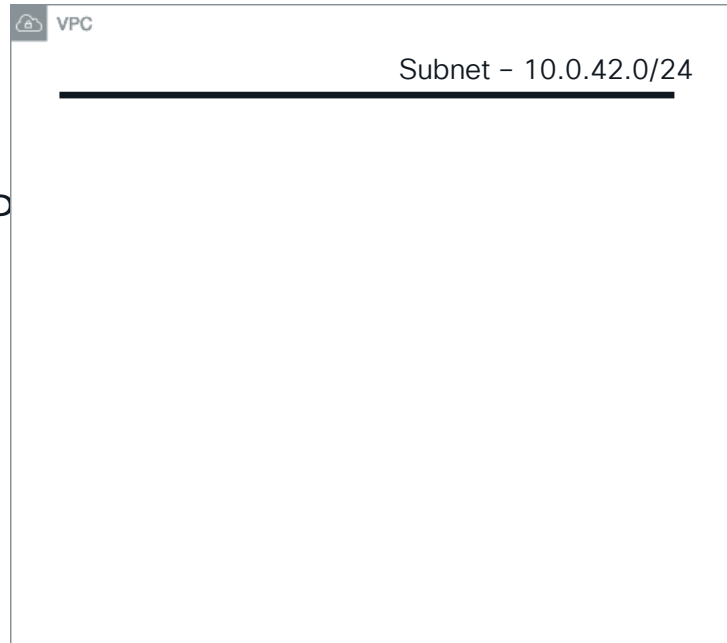
Terraform and Ansible Basics – IaC 101



Terraform



AWS AP



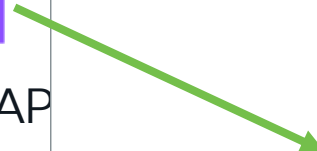
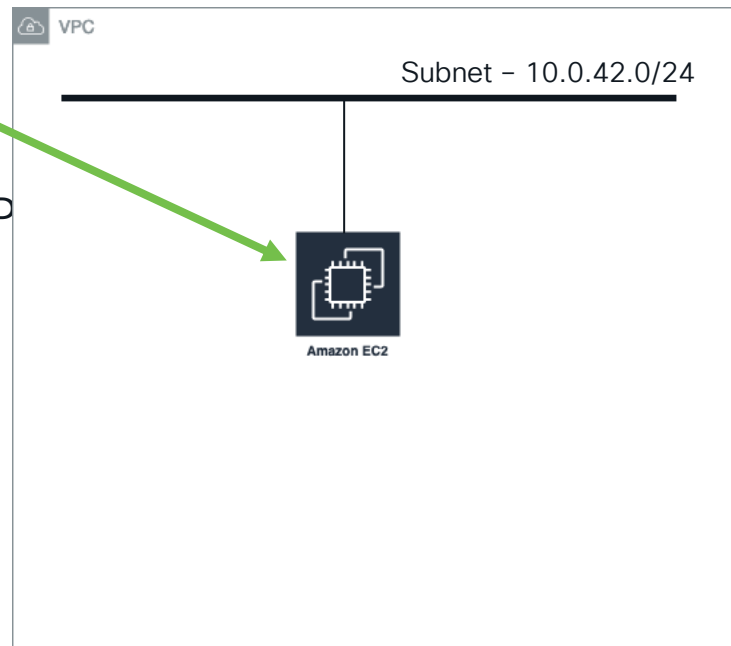


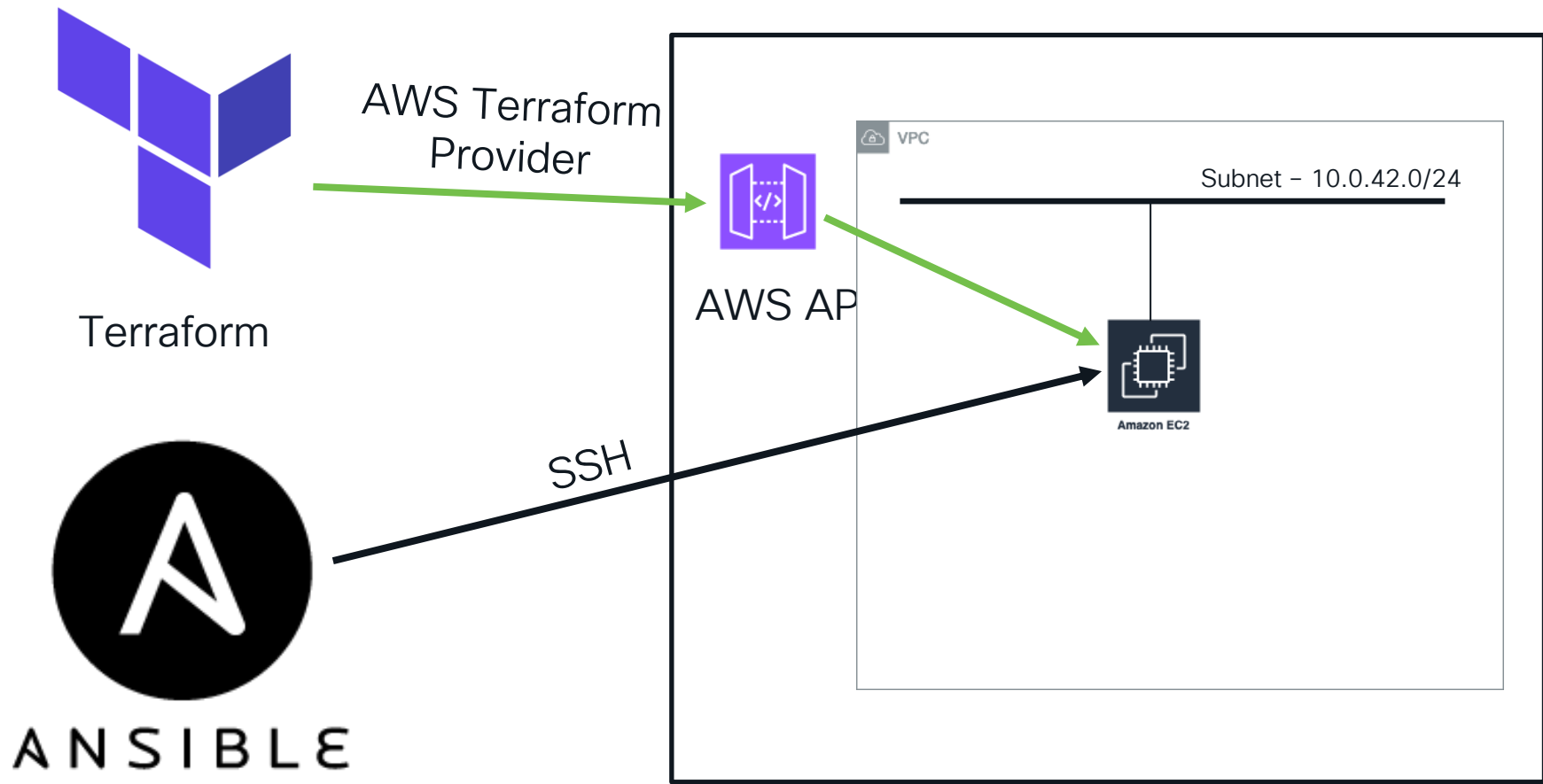
Terraform

AWS Terraform
Provider



AWS AP





Terraform Structure

```
#filename: Main.tf
< Provider>

< Security Group Resource >

< C9800 Resource >

< ISE Resource >

< Cisco Spaces Connector Resource >
```

Terraform Providers

```
#filename: main.tf
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 3.5.0"
    }
  }
}
```

Terraform AWS Provider – <https://registry.terraform.io/providers/hashicorp/aws/latest>

Terraform Variables

```
#filename: variables.tf
variable "[variable name]" {
  type    = string
  default = "[variable value]"
}
```

Terraform Structure – AWS Instance Resource

```
#filename: main.tf
resource "aws_instance" "[Instance Name which is Referenced]" {
  ami                    = var.[variable name]
  instance_type          = var.[variable name]
  subnet_id             = var.[variable name]
  private_ip            = var.[variable name]
  vpc_security_group_ids = [aws_security_group.[variable name].id]
  key_name               = var.[variable name]
  associate_public_ip_address = true
  tags = {
    Name = [AWS Instance Name]
  }
  credit_specification {
    cpu_credits = "unlimited"
  }
  user_data = file([user Data text file])
}
```

Terraform Mode of operation - Init

```
mrebetzk ~ % terraform init
Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v3.5.0

Terraform has been successfully initialized!
```

Terraform Mode of operation – Plan (read-only)

```
mrebetzk ~ % terraform plan
# aws_instance.CiscoLive-C9800-CL will be created
+ resource "aws_instance" "CiscoLive-C9800-CL" {
  + ami              = "ami-01219f5e8df8a426e"
  + host_id          = (known after apply)
  + id               = (known after apply)
  + key_name         = "mre-CiscoLive"
  + private_ip       = (known after apply)
  + subnet_id        = "subnet-0f28d11bdca7596ba"

  <...omitted...>
```

Terraform Mode of operation – Apply

```
mrebetzk ~ % terraform apply
# aws_instance.CiscoLive-C9800-CL will be created
+ resource "aws_instance" "CiscoLive-C9800-CL" {
  + ami              = "ami-01219f5e8df8a426e"
  + host_id          = (known after apply)
  + id               = (known after apply)
  + key_name         = "mre-CiscoLive"
  + private_ip       = (known after apply)
  + subnet_id        = "subnet-0f28d11bdca7596ba"

  <...omitted...>
```

Terraform Mode of operation – Destroy

```
mrebetzk ~ % terraform destroy
# aws_instance.CiscoLive-C9800-CL will be destroyed
+ resource "aws_instance" "CiscoLive-C9800-CL" {
  + ami              = "ami-01219f5e8df8a426e"
  + host_id          = (known after apply)
  + id               = (known after apply)
  + key_name         = "mre-CiscoLive"
  + private_ip       = (known after apply)
  + subnet_id        = "subnet-0f28d11bdca7596ba"

  <...omitted...>
```

Ansible Script for C9800 Config

```
#C9800 Ansible Playbook
- name: Configure Commands
  cisco.ios.ios_command:
    commands:
      - configure terminal
      - username Netconf privilege 15 password 0 netconf
      - username Restconf privilege 15 password 0 restcon
      - netconf-yang
      - ntp server au.pool.ntp.org
      - ip route 0.0.0.0 0.0.0.0 10.0.10.1
      - wireless country AU
      - wlan CiscoLiveTest 1 CiscoLiveTest
      - security wpa psk set-key ascii 0 cisco1234
      - no shutdown
      - interface g1
      - wireless management interface g1
      - end
      - wireless config vwlc-ssc key-size 2048 signature-algo sha256
        password 0 cisco123
```

Ansible 101 – Running Playbook

```
#filename: inventory
[inventory ]
ansible_connection=ansible.netcommon.network_cli
ansible_network_os=cisco.ios.ios
ansible_user=[Username to SSH access IOS-XE Device]
ansible_become=yes
ansible_become_method=enable
ansible_become_password=[password]
ansible_ssh_pass= [password]
host_key_checking = False

[iosxe]
mre-c9800-vmware
...

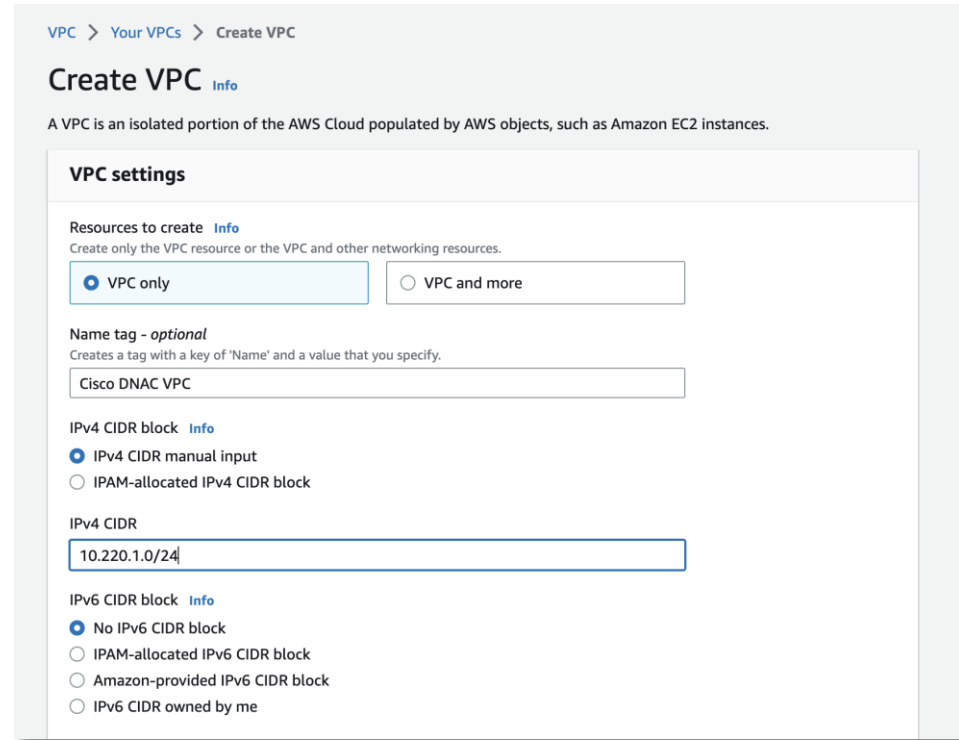
#main.yml
mrebetzk ~ % ansible-playbook -i inventory main.yml
```

AWS Network Constructs



Virtual Private Cloud (VPC)

With Amazon Virtual Private Cloud (Amazon VPC), you can launch AWS resources in a logically isolated virtual network that you've defined. This virtual network closely **resembles a traditional network that you'd operate in your own data centre**, with the benefits of using the scalable infrastructure of AWS.



The screenshot shows the 'Create VPC' page in the AWS Management Console. The breadcrumb navigation at the top reads 'VPC > Your VPCs > Create VPC'. The main heading is 'Create VPC' with an 'Info' link. Below this, a descriptive sentence states: 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances.'

The 'VPC settings' section contains the following options:

- Resources to create** (with an 'Info' link):
 - Radio button selected: **VPC only**
 - Radio button: **VPC and more**
- Name tag - optional** (with an 'Info' link):
 - Text: 'Creates a tag with a key of 'Name' and a value that you specify.'
 - Text input field containing: **Cisco DNAC VPC**
- IPv4 CIDR block** (with an 'Info' link):
 - Radio button selected: **IPv4 CIDR manual input**
 - Radio button: **IPAM-allocated IPv4 CIDR block**
 - Text input field for 'IPv4 CIDR' containing: **10.220.1.0/24**
- IPv6 CIDR block** (with an 'Info' link):
 - Radio button selected: **No IPv6 CIDR block**
 - Radio button: **IPAM-allocated IPv6 CIDR block**
 - Radio button: **Amazon-provided IPv6 CIDR block**
 - Radio button: **IPv6 CIDR owned by me**

Virtual Private Cloud (VPC) – Subnet

A subnet is a range of IP addresses in your VPC. A subnet must reside in a single Availability Zone. After you add subnets, you can deploy AWS resources in your VPC.

Create subnet [Info](#)

VPC

VPC ID
Create subnets in this VPC.

vpc-08dfd2dfdd1638ebf (DNAC-VPC) ▼

Associated VPC CIDRs

IPv4 CIDRs

10.220.1.0/24

Subnet settings

Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

my-subnet-01

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

No preference ▼

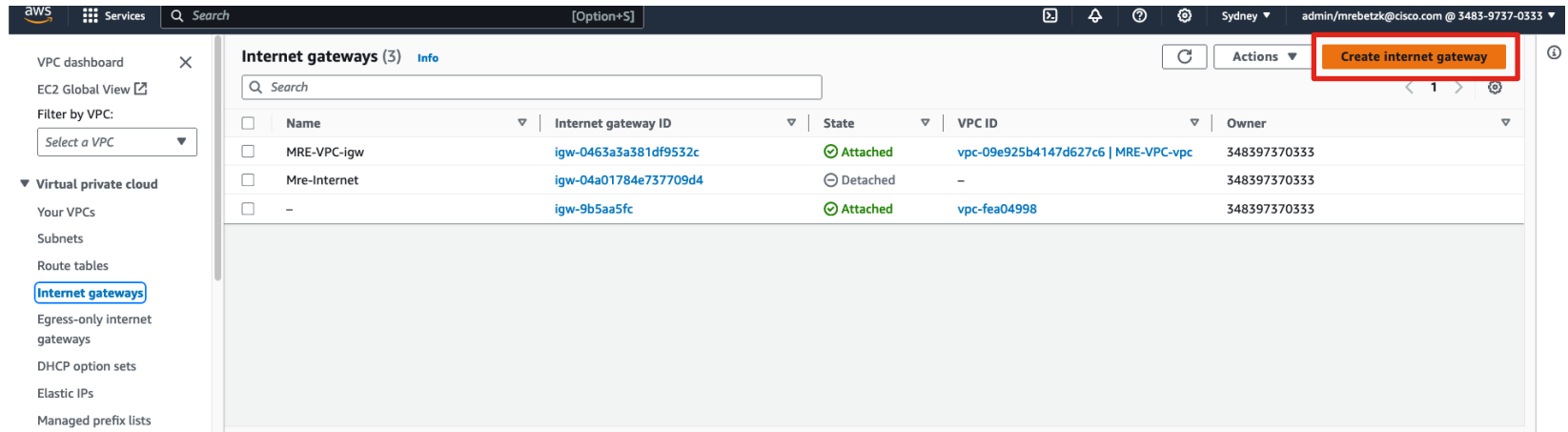
IPv4 CIDR block [Info](#)

10.0.0.0/24

Create an Internet Gateway

Internet Gateway

An internet gateway enables resources in your public subnets (such as EC2 instances) to connect to the internet if the resource has a public IPv4 address or an IPv6 address.



The screenshot shows the AWS Management Console interface for the 'Internet gateways' section. The left sidebar contains navigation links for VPC dashboard, EC2 Global View, and Virtual private cloud resources. The main content area displays a table of internet gateways. A red box highlights the 'Create internet gateway' button in the top right corner of the main content area.

Name	Internet gateway ID	State	VPC ID	Owner
MRE-VPC-igw	igw-0463a3a381df9532c	Attached	vpc-09e925b4147d627c6 MRE-VPC-vpc	348397370333
Mre-Internet	igw-04a01784e737709d4	Detached	-	348397370333
-	igw-9b5aa5fc	Attached	vpc-fea04998	348397370333

Internet Gateway

An internet gateway enables resources in your public subnets (such as EC2 instances) to connect to the internet if the resource has a public IPv4 address or an IPv6 address.

The screenshot shows the AWS Management Console interface for creating an internet gateway. The breadcrumb navigation at the top reads 'VPC > Internet gateways > Create internet gateway'. The main heading is 'Create internet gateway' with an 'Info' link. Below this is a descriptive paragraph: 'An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.' The 'Internet gateway settings' section contains a 'Name tag' field, which is highlighted with a red rectangle and contains the text 'MRE-VP-IGW'. Below this is the 'Tags - optional' section, which includes a table for adding tags with columns for 'Key' and 'Value - optional'. The table currently has one entry with the key 'Name' and the value 'MRE-VP-IGW'. There is an 'Add new tag' button and a note stating 'You can add 49 more tags.' At the bottom of the form are 'Cancel' and 'Create internet gateway' buttons.

aws Services Search [Option+S] Sydney admin/mrebetzk@cisco.com @ 3483-9737-0333

VPC > Internet gateways > Create internet gateway

Create internet gateway [Info](#)

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

MRE-VP-IGW

Tags - optional

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="MRE-VP-IGW"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

Internet Gateway

An internet gateway enables resources in your public subnets (such as EC2 instances) to connect to the internet if the resource has a public IPv4 address or an IPv6 address.

The screenshot shows the AWS Management Console interface for the 'Route tables' section of a VPC named 'MRE-VPC-vpc'. The table lists four route tables, with the first one, 'MRE-VPC-rtb-public', highlighted by a red border. This table is the main route table for the VPC and is associated with two public subnets. The other three route tables are associated with private subnets.

Name	Route table ID	Explicit subnet associati...	Edge associations	Main	VPC	Owner
MRE-VPC-rtb-public	rtb-0b89b6b14b76a2921	2 subnets	-	No	vpc-09e925b4147d627c6 MRE-VPC-vpc	348
MRE-VPC-rtb-private1-ap-southeast-2a	rtb-001ecab9b85fd6af8	subnet-02314aa6bff8834...	-	No	vpc-09e925b4147d627c6 MRE-VPC-vpc	348
MRE-VPC-rtb-private2-ap-southeast-2b	rtb-0fe924fb9f69f16a9	subnet-0b9a4bd1cef622...	-	No	vpc-09e925b4147d627c6 MRE-VPC-vpc	348
-	rtb-01b019614c9ce48dc	-	-	Yes	vpc-09e925b4147d627c6 MRE-VPC-vpc	348

Internet Gateway

An internet gateway enables resources in your public subnets (such as EC2 instances) to connect to the internet if the resource has a public IPv4 address or an IPv6 address.

VPC > Route tables > rtb-0b89b6b14b76a2921

rtb-0b89b6b14b76a2921 / MRE-VPC-rtb-public Actions ▾

Details [Info](#)

Route table ID
rtb-0b89b6b14b76a2921

VPC
vpc-09e925b4147d627c6 | MRE-VPC-vpc

Main
No

Owner ID
348397370333

Explicit subnet associations
2 subnets

Edge associations
-

[Routes](#) | [Subnet associations](#) | [Edge associations](#) | [Route propagation](#) | [Tags](#)

Routes (3) Both ▾ Edit routes

Destination	Target	Status	Propagated
0.0.0.0/0	lgw-0463a3a381df9532c	Active	No
10.0.42.0/24	local	Active	No
10.0.43.0/24	vgw-08839f6414687e949	Active	No

Internet Gateway

An internet gateway enables resources in your public subnets (such as EC2 instances) to connect to the internet if the resource has a public IPv4 address or an IPv6 address.

VPC > Route tables > rtb-0b89b6b14b76a2921 > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.42.0/24	local Q local X	Active	No
Q 10.0.43.0/24 X	Virtual Private Gateway Q vgw-08839f6414687e949 X	Active	No <button>Remove</button>
Q 0.0.0.0/0 X	Internet Gateway Q igw-0463a3a381df9532c X	Active	No <button>Remove</button>

Add route

Cancel Preview Save changes

AWS Security Constructs

Create a Keypair

Create Key Pair for SSH connection

aws Services Key Pair

New EC2 Experience
Tell us what you think

Services (9)
Features (18)
Resources **New**
Blogs (1,682)
Documentation (22,421)
Knowledge Articles (16)
Tutorials (6)
Events (49)
Marketplace (105)

EC2 Dashboard
EC2 Global View
Events
Limits

Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations

Images
AMIs
AMI Catalog

Search results

Key pairs (11) Info

Search

Name

anandg

EC2 > Key pairs > Create key pair

Create key pair [Info](#)

Key pair
A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.

Name

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type [Info](#)

☒ RSA
☐ ED25519

Private key file format

☒ .pem
For use with OpenSSH

☐ .ppk
For use with PuTTY

Tags - optional
No tags associated with the resource.

You can add up to 50 more tags.

Key will be downloaded automatically after creation

```
mrebetzk ~ % cd downloads
mrebetzk downloads ~ % chmod 400 <cert name>.pem
mrebetzk downloads ~ % mv <cert name> ~/.ssh/<cert name>
mrebetzk downloads ~ % ssh -i "<cert name>.pem" Username@<IP Address>
```

Create a Security Group

Security Group

Security Group

Micro segmentation

Security Group

Micro segmentation

Tied to a AWS instance

Security Group

Micro segmentation

Tied to a AWS instance

Stateful Behavior

Create Security Groups

EC2 > Security Groups > Create security group

Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name [Info](#)

Name cannot be edited after creation.

Description [Info](#)

VPC [Info](#)

The recommendation is to limited inbound port access to only essential ports.

Ports needed for Cisco Wireless Stack

Controller Ports

Protocol and Ports	Purpose
UDP 5246-5248	CAPWAP
TCP 22	SSH,SCP
TCP 21	FTP
ICMP	Ping
UDP 161, 162	SNMP/SNMP Traps
TCP 443/80	HTTPS/HTTP
TCP/UDP 49	TACACS+
UDP 53	DNS Server
UDP 1812,1813,1645,1646	Radius
UDP 123	NTP
UDP 514	Syslog

ISE Ports

https://www.cisco.com/c/en/us/td/docs/security/ise/3-3/install_guide/b_ise_installationGuide33.html

Catalyst Center Ports

https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/network-automation-and-management/dna-center/dna-center-va/aws/deploy/1_2/b_cisco_dna_center_va_aws_deployment_guide_1_2.html

Cisco Spaces Connector Ports

https://www.cisco.com/c/en/us/td/docs/wireless/spaces/connector/2-x/config/b_connector/m_ami.html

Add Security Groups with Terraform!

```
resource "aws_security_group" "CiscoLive-securityGroup" {  
  name           = "MRE Cisco Live SecGroup"  
  description    = "Allow inbound CAPWAP Traffic"  
  vpc_id         = var.vpc_id  
  
  ingress {  
    description = "CAPWAP"  
    from_port   = 5246  
    to_port     = 5248  
    protocol    = "tcp"  
    cidr_blocks = [var.cidr_blocks]  
  }  
}
```

Wireless Infrastructure – C9800 WLC



Deploying Cisco Catalyst 9800 WLC

Deployment Options

1. Manual EC2 Instance deployment
2. Terraform

The screenshot shows the Cisco Catalyst 9800-CL Wireless Controller for Cloud configuration interface. At the top, there's a Cisco logo and a 'Continue to Launch' button. Below the header, there are tabs for 'Product Detail', 'Subscribe', and 'Configure'. The main heading is 'Configure this software'. A sub-heading says 'Choose a fulfillment option and software version to launch this software.' There are four dropdown menus: 'Fulfillment option' (set to 'CloudFormation Template'), 'Cisco Catalyst C9800-CL Wireless Controller' (set to 'Cisco Catalyst 9800-CL Wireless Controller'), 'Software version' (set to '17.9.1 (Nov 28, 2022)'), and 'Region' (set to 'US West (N. California)'). To the right of the 'Fulfillment option' dropdown, there's a 'CloudFormation Template' section with the text 'Deploy a complete solution configuration using a CloudFormation template'. Below the 'Software version' dropdown, there's a 'Whats in This Version' section with the text 'Cisco Catalyst 9800-CL Wireless Controller for Cloud running on c5.xlarge' and a 'Learn more' link. On the right side of the page, there's a 'Pricing information' section with a disclaimer: 'This is an estimate of typical software and infrastructure costs based on your configuration. Your actual charges for each statement period may differ from this estimate.' Below this, there's a 'Software Pricing' section with the text 'Cisco Catalyst 9800-CL Wireless Controller for Cloud' and a price of '\$0/hr'.

EC2 and CloudFormation Deployment Guide: https://www.cisco.com/c/en/us/td/docs/wireless/controller/technotes/8-8/b_cisco_catalyst_9800_wireless_controller_aws.html

Configure Catalyst C9800 AWS – Manual

AMI Catalog

← → ↻ 🔒 https://ap-southeast-2.console.aws.amazon.com/ec2/home?region=ap-southeast-2#AMICatalog: 110% 🇺🇸 📧 ⚙️ 📧 admin@mrebetzk@cisco.com @ 3483-9737-0333

aws Services 🔍 Search [Option+S]

EC2 Dashboard ✕
EC2 Global View
Events
▼ Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
New
▼ Images
AMIs
AMI Catalog
▼ Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
▼ Network & Security
Security Groups

EC2 > AMI Catalog

AMI Catalog

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

AMIs Create Template with AMI Launch Instance with AMI

🔍 9800 ✕ ▼

Quickstart AMIs (0) **My AMIs (420)** **AWS Marketplace AMIs (2)** **Community AMIs (33)**

Commonly used AMIs Created by me AWS & trusted third-party AMIs Published by anyone

▼ **Refine results**

Categories
[Infrastructure Software \(2\)](#)

Publisher
☐ Cisco Systems, Inc. (2)

Pricing model
☐ Bring Your Own License (2)

9800 (2 results) showing 1 - 2
Did you mean [9000?](#) Sort By: Relevance ▼

Cisco Catalyst 9800-CL Wireless Controller for Cloud
By [Cisco Systems, Inc.](#) | Ver 17.9.1
★★★★★ 1 AWS review

The Bring Your Own License (BYOL) version of next generation wireless controller (C9800-CL-K9) combines the advantages and flexibility of an AWS public cloud with the customization and features richness customers usually get with on-prem deployments. As part of Cisco's Cloud portfolio, the Catalyst...

Select

AMI Catalog

The screenshot shows the AWS Management Console's AMI Catalog. A modal window is open for the 'Cisco Catalyst 9800-CL Wireless Controller for Cloud' AMI. The modal includes the Cisco logo, the product name, the publisher 'Cisco Systems, Inc.', a 1-star AWS review, and the option to 'Bring Your Own License'. Below this, there are tabs for 'Overview', 'Product details', 'Pricing', 'Usage', and 'Support'. The 'Overview' tab is active, displaying a description of the AMI as a next-generation enterprise-class wireless controller for cloud. It also lists the typical total price as \$0.17/Hr, the latest version as 17.9.1, and the delivery methods as Amazon Machine Image and CloudFormation Template. The categories listed are Network Infrastructure. A 'Continue' button is at the bottom right of the modal. In the background, the AMI Catalog interface is visible, showing a search bar with '9800', a list of AMIs, and a filter sidebar on the left.

Cisco Catalyst 9800-CL Wireless Controller for Cloud

Cisco Systems, Inc. [1 AWS review](#)
Bring Your Own License

Overview | Product details | Pricing | Usage | Support

The Cisco Catalyst 9800-CL is the next generation of enterprise-class wireless controller for cloud that runs open Cisco IOS XE Software and sets the standard for always-on and secure wireless network, bringing the world's most popular wireless networking platform to AWS

Typical total price \$0.17/Hr Total pricing per instance for services hosted on c5.xlarge in us-east-1. See additional pricing information.	Latest version 17.9.1 Delivery methods Amazon Machine Image ¹ CloudFormation Template ¹ Operating systems Other Cisco IOS XE	Categories Network Infrastructure
---	--	--------------------------------------

[Continue](#)

AMI Catalog

aws

Services

Search

[Option+S]

Sydney

admin/mrebetzk@cisco.com @ 3483-9737-0333

EC2 > Instances > Launch an instance

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name

test

Add additional tags

Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q 9800

AMI from catalog

Recents

Quick Start

Amazon Machine Image (AMI)

C9800-CL-17-9-1.20220919233909-36aaa0b8-cf25-45aa-9fcc-16915f24ce71

ami-01219f5e8df8a426e

Verified provider

Browse more AMIs

Including AMIs from AWS, Marketplace and

Summary

Number of instances

1

Software Image (AMI)

[Copied ami-0e0ee675c4041a4fe ...read more]

ami-01219f5e8df8a426e

Virtual server type (instance type)

c5.xlarge

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 16 GiB

Cancel

Launch instance

Review commands

CISCO Live!

#CiscoLiveAPJC

BRKEWN-1104

© 2023 Cisco and/or its affiliates. All rights reserved. Cisco Public

78

Terraform AWS USER DATA

```
#C9800 bootstrap config. Passed via UserData - C9800.txt  
ios-config-1="hostname MRE_C9800_CL"  
ios-config-2="username cisco priv 15 pass cisco123"
```

Day-0 Configuration

Configuration Setup Wizard

1. General Settings

Host Name*

C9800-AWS-3

Country

US

Date

01 Jun 2023

Time / Timezone

10:41:06

Pacific

NTP Servers

Enter NTP Server

Added NTP servers

10.10.105.7

AAA Servers

Enter Radius Server IP

Enter Key

Added AAA servers

Wireless Management Settings

Port Number

GigabitEthernet1

IP Address

10.220.3.20

Next

Day-0 Configuration

3. Advanced Settings

Client Density

◀ ————— Typical ————— ▶
Low High

RF Group Name*

default

Traffic Type

Data and Voice ▼

Virtual IP Address

192.0.2.1

AP Certificate

Generate Certificate

YES ☒

RSA Key-Size

2048 ▼

Signature Algorithm

sha256 ▼

Password*

***** ☐

Create New AP Management User (optional)

New AP Management User

admin

Password

***** ☐

Secret

***** ☐

Previous

Summary

Day-0 Configuration

4. Summary

▼ General Settings

Host Name

C9800-AWS-3

Country

US

Date

01 Jun 2023

Time / Timezone

NTP Servers

Wireless Management

Port Number

▼ Wireless Network Settings

Network Name

◀ ◻ ▶

0

▶ ◻ ▶

10 ▼

▼ Advanced Settings

Client Density

RF Group Name

default

Traffic Type

data and voice

Virtual IPv4 Address

192.0.2.1

It may take a minute to apply the configuration. You will be logged out and asked to login again. Are you sure you want to proceed ?

No

Yes

Previous

Finish

EC2 Serial Connect

EC2 > Instances > i-05b2314698cf2384a

Instance summary for i-05b2314698cf2384a (BRKEWN-1104-Demo-WLC) Info

Updated less than a minute ago



Connect

Instance state ▼

Actions ▼

Instance ID

i-05b2314698cf2384a (BRKEWN-1104-Demo-WLC)

Public IPv4 address

–

Private IPv4 addresses

10.220.3.20

IPv6 address

–

Instance state

Running

Public IPv4 DNS

–

Elastic IP addresses

EC2 > Instances > i-05b2314698cf2384a > Connect to instance

Connect to instance Info

Connect to your instance i-05b2314698cf2384a (BRKEWN-1104-Demo-WLC) using any of these options

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-05b2314698cf2384a (BRKEWN-1104-Demo-WLC)

Serial port

ttyS0

Cancel

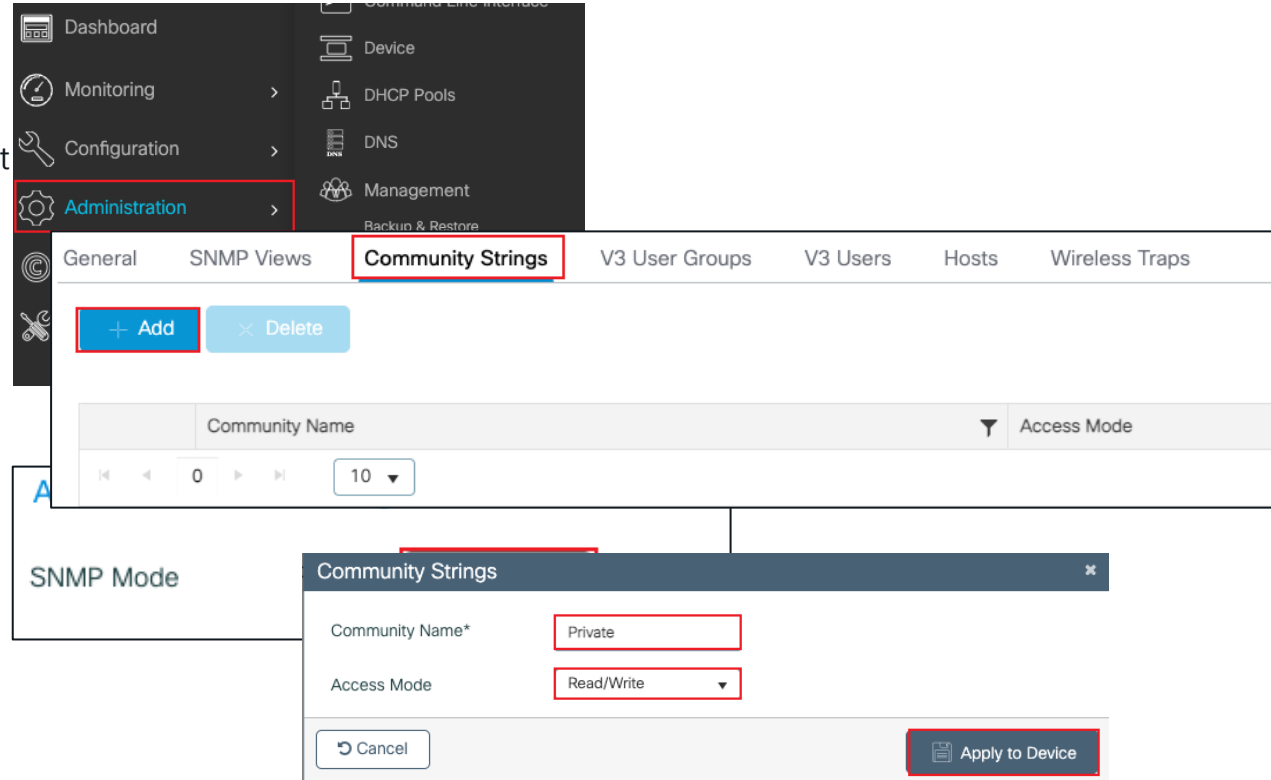
Connect

```
C9800-AWS-3#sh ip int br
Interface      IP-Address      OK? Method Status              Protocol
GigabitEthernet1  10.220.3.20     YES DHCP    up                  up
Vlan1          unassigned      YES unset    administratively down down
C9800-AWS-3#conf t
Enter configuration commands, one per line. End with CNTL/Z.
C9800-AWS-3(config)#int G1
C9800-AWS-3(config-if)#ip address 10.220.3.20 255.255.255.0
C9800-AWS-3(config-if)#exit
```

Configure SNMP

Catalyst Center on-board pre-request

- SSH enabled ✓
- CLI credentials ✓
- Netconf enabled and define the port 830 ✓
- SNMP enabled and configure community strings (?)



Configure Catalyst C9800 AWS – Terraform

Terraform Structure

```
#main.tf
resource "aws_instance" "CiscoLive-C9800-CL" {
  ami                = var.c9800-ami
  instance_type      = var.instance_type
  subnet_id          = var.subnet_id
  private_ip         = var.c9800-staticIPv4
  vpc_security_group_ids = [aws_security_group.CiscoLive-
                           securityGroup.id]

  key_name           = var.key_name
  associate_public_ip_address = true
  tags = {
    Name = "MRE_C9800-CL"
  }
  credit_specification {
    cpu_credits = "unlimited"
  }
  user_data = file("9800.txt")
}
```

Terraform AWS USER DATA

```
#C9800 bootstrap config. Passed via UserData - C9800.txt
ios-config-1="hostname MRE_C9800_CL"
ios-config-2="username cisco priv 15 pass cisco123"
ios-config-3="wireless country AU"
ios-config-4="ntp server au.pool.ntp.org"
```

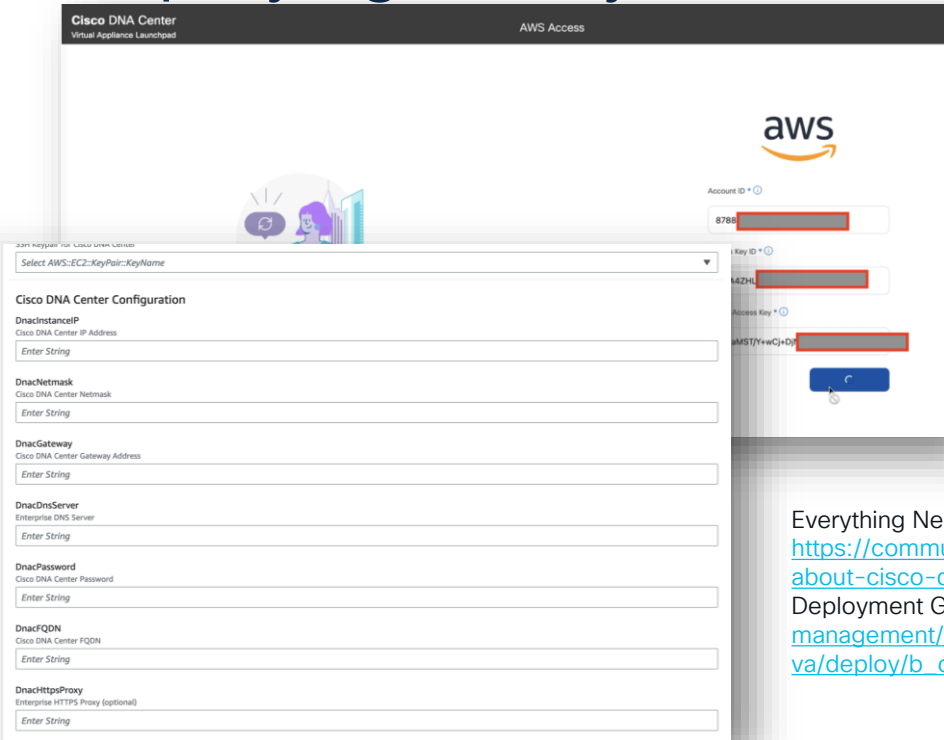
Ansible Script for C9800 Config

```
#C9800 Ansible Playbook
- name: Configure Commands
  cisco.ios.ios_command:
    commands:
      - configure terminal
      - username Netconf privilege 15 password 0 netconf
      - username Restconf privilege 15 password 0 restcon
      - netconf-yang
      - ntp server au.pool.ntp.org
      - ip route 0.0.0.0 0.0.0.0 10.0.10.1
      - wireless country AU
      - wlan CiscoLiveTest 1 CiscoLiveTest
      - security wpa psk set-key ascii 0 cisco1234
      - no shutdown
      - interface g1
      - wireless management interface g1
      - end
      - wireless config vwlc-ssc key-size 2048 signature-algo sha256
        password 0 cisco123
```

Wireless Infrastructure – Catalyst Center



Deploying Catalyst Center on AWS



Cisco DNA Center
Virtual Appliance Launchpad

AWS Access

Account ID * 

878b-

Key ID * 

AKZHL-

Access Key * 

AMSTY+veCjxDy-

Select AWS::EC2::KeyPair::KeyName

Cisco DNA Center Configuration

DnacInstanceIP
Cisco DNA Center IP Address

DnacNetmask
Cisco DNA Center Netmask

DnacGateway
Cisco DNA Center Gateway Address

DnacDnsServer
Enterprise DNS Server

DnacPassword
Cisco DNA Center Password

DnacFQDN
Cisco DNA Center FQDN

DnacHttpsProxy
Enterprise HTTPS Proxy (optional)

Deployment Options

1. CloudFormation Template – Catalyst Center only
2. Launchpad VA App – Catalyst Center and AWS infra

Everything Needed for Cisco DNA Center on AWS:

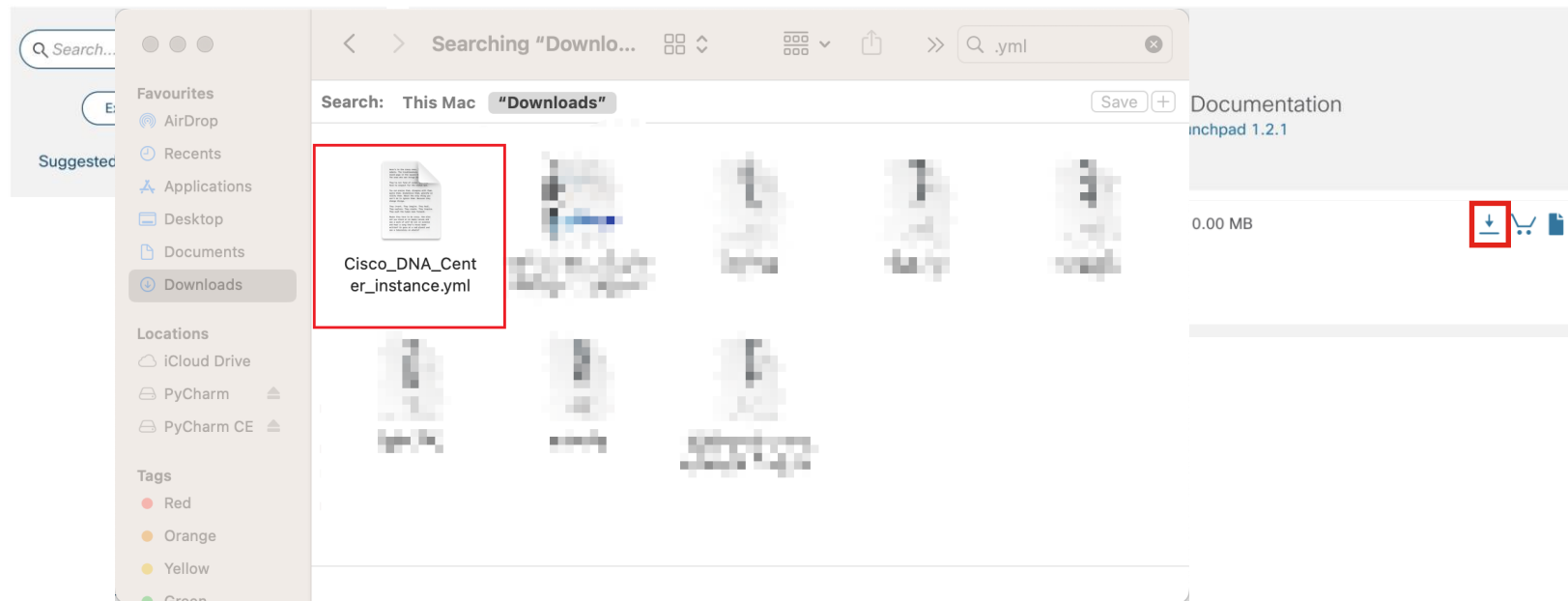
<https://community.cisco.com/t5/networking-knowledge-base/everything-you-need-to-know-about-cisco-dna-center-va-on-aws/ta-p/4731299>

Deployment Guide: https://www.cisco.com/c/en/us/td/docs/cloud-systems-management/network-automation-and-management/dna-center/dna-center-va/deploy/b_cisco_dna_center_va_aws_deployment_guide.html

Cisco Catalyst Center on AWS with CloudFormation

Software Download

[Downloads Home](#) / [Cloud and Systems Management](#) / [Network Automation and Management](#) / [Cisco DNA Center](#) / [DNA Center Software- VA Launchpad 1.2.1](#)



Cisco Catalyst Center on AWS with CloudFormation

The screenshot displays the AWS Management Console interface. At the top, the search bar contains the text 'cloudformation'. Below the search bar, the search results for 'cloudform' are shown. A red rectangle highlights the 'Services' section of the results. Within this section, 'CloudFormation' is listed as a service, with a description 'Create and Manage Resources with Templates'. Below the service name, the 'Top features' are listed: 'StackSets', 'Resource import', 'Stacks', 'Exports', and 'Designer'. To the right of the search results, a panel shows the 'Create stack' button and a list of templates, including 'Template for Cisco Catalyst 9800-CL Wireless Controller for Cloud --AWSMP::36aaa0b8-4ce71::fca11194-78c0-43ed-debf-439f521f4376'.

Cisco Catalyst Center on AWS with CloudFormation

The screenshot shows the AWS CloudFormation console interface. At the top, there's a navigation bar with the AWS logo, 'Services' link, a search bar, and a user profile section. Below the navigation bar, the breadcrumb 'CloudFormation > Stacks' is visible. The main content area is titled 'Stacks (6)' and includes a search bar, a 'Filter status' dropdown set to 'Active', and a 'View nested' toggle. A table lists six stacks, all with a status of 'CREATE_COMPLETE'. The 'Create stack' button in the top right corner of the stack list is highlighted with a red rectangular box.

Stack name	Status	Created time	Description
cisco-dna-center	✔ CREATE_COMPLETE	2023-10-10 11:53:47 UTC+1100	Cisco DNA Center Prerequisite Resources Deployment
syd-ise-stack	✔ CREATE_COMPLETE	2021-08-26 12:39:03 UTC+1000	AWS Cloud Formation Template for Cisco Identity Service Engine. CFT Version : 3.1.518
Demo-5	✔ CREATE_COMPLETE	2021-05-24 17:40:32 UTC+1000	AWS CloudFormation Template for Cisco Catalyst 9800-CL Wireless Controller for Cloud --AWSMP::36aaa0b8-cf25-45aa-9fcc-16915f24ce71::fca11194-78c0-43ed-debf-439f521f4376
demo-9800-2	✔ CREATE_COMPLETE	2021-05-21 17:45:29 UTC+1000	AWS CloudFormation Template for Cisco Catalyst 9800-CL Wireless Controller for Cloud --AWSMP::36aaa0b8-cf25-45aa-9fcc-16915f24ce71::fca11194-78c0-43ed-debf-439f521f4376
Demo-1	✔ CREATE_COMPLETE	2021-05-21 16:50:43 UTC+1000	AWS CloudFormation Template for Cisco Catalyst 9800-CL Wireless Controller for Cloud --AWSMP::36aaa0b8-cf25-45aa-9fcc-16915f24ce71::fca11194-78c0-43ed-debf-439f521f4376
Demo	✔ CREATE_COMPLETE	2021-05-21 14:36:10 UTC+1000	AWS CloudFormation Template for Cisco Catalyst 9800-CL Wireless Controller for Cloud --AWSMP::36aaa0b8-cf25-45aa-9fcc-16915f24ce71::fca11194-78c0-43ed-debf-439f521f4376

Cisco Catalyst Center on AWS with CloudFormation

aws

Services

Search

[Option+S]

Sydney

admin/mrebetzk@cisco.com @ 3483-9737-0333

CloudFormation > Stacks > Create stack

Step 1
Create stack

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

☒ Template is ready

☐ Use a sample template

☐ Create template in Designer

Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

☐ Amazon S3 URL

☒ Upload a template file

Upload a template file

Choose file

Cisco_DNA_Center_Instance.yml

JSON or YAML formatted file

S3 URL: https://s3.ap-southeast-2.amazonaws.com/cf-templates-vmcuw2ka59ob-ap-southeast-2/2023-11-07T213206.304276c-Cisco_DNA_Center_Instance.yml

View in Designer

Cancel

Next

CISCO *Live!*

#CiscoLiveAPJC

BRKEWN-1104

© 2023 Cisco and/or its affiliates. All rights reserved. Cisco Public

94

Cisco Catalyst Center on AWS with CloudFormation

Cisco DNA Center Configuration

DnacInstanceIP
Cisco DNA Center IP Address

DnacNetmask
Cisco DNA Center Netmask

DnacGateway
Cisco DNA Center Gateway Address

DnacDnsServer
Enterprise DNS Server

DnacPassword
Cisco DNA Center Password

DnacFQDN
Cisco DNA Center FQDN

DnacHttpsProxy
Enterprise HTTPS Proxy (optional)

DnacHttpsProxyUsername
HTTPS Proxy Username (optional)

DnacHttpsProxyPassword
HTTPS Proxy Password (optional)

Management IP

Subnet Mask on the aws subnet

Gateway Address (.1) of your aws subnet

DNS Server Address

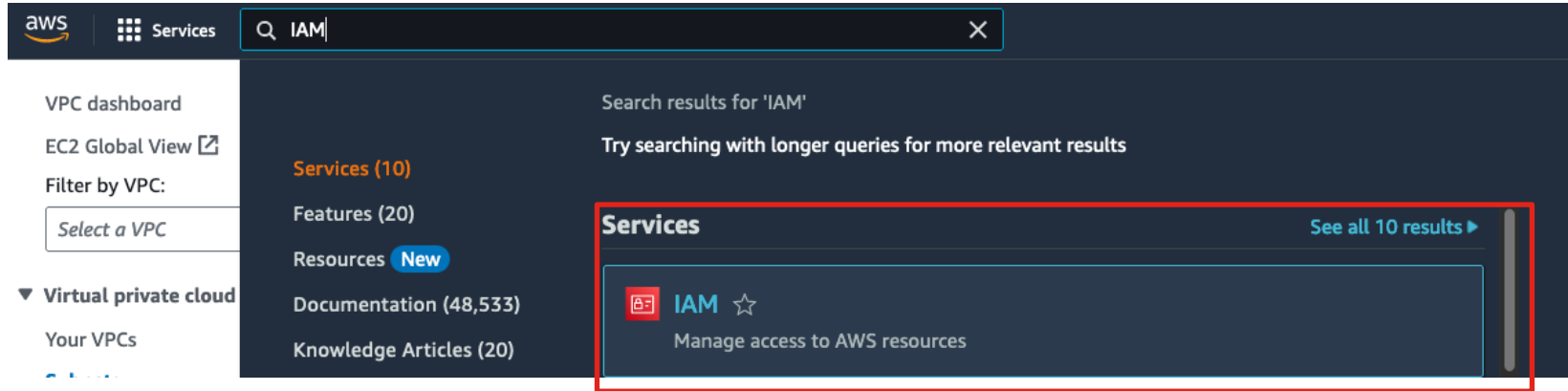
DNAC password (username default maglev)

DNAC FQDN

HTTPs proxy settings(optional)

Configure Catalyst Center – LaunchPad

Cisco Catalyst Center on AWS with Launchpad



Cisco Catalyst Center on AWS with Launchpad

The screenshot shows the AWS IAM console interface. The top navigation bar includes the AWS logo, 'Services', a search bar, and a user profile dropdown. The left sidebar shows the navigation menu with 'IAM > Users > Create user' selected. The main content area is titled 'Specify user details' and contains a 'User details' section. In this section, the 'User name' field is populated with 'cisco-DNA-launchpad'. Below the field, a note states: 'The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . _ - (hyphen)'. There is an unchecked checkbox for 'Provide user access to the AWS Management Console - optional' with a link to 'best practice'. At the bottom right of the form are 'Cancel' and 'Next' buttons.

aws Services Search [Option+S]

Global admin/mrebetzk@cisco.com @ 3483-9737-0333

IAM > Users > Create user

Step 1
Specify user details

Step 2
[Set permissions](#)

Step 3
Review and create

Specify user details

User details

User name

cisco-DNA-launchpad

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . _ - (hyphen)

☐ Provide user access to the AWS Management Console - *optional*
If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

Info If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. [Learn more](#)

Cancel Next

Cisco Catalyst Center on AWS with Launchpad

Step 1
Add permissions

Step 2
[Review](#)

Add permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

Permissions options

- ☐ **Add user to group**
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- ☐ **Copy permissions**
Copy all group memberships, attached managed policies, inline policies, and any existing permissions boundaries from an existing user.
- ☒ **Attach policies directly**
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1/1136)

Search:

Filter by Type: All types

☐	Policy name	Type	Attached entities
☐	AccessAnalyzerServiceRolePolicy	AWS managed	0
☒	AdministratorAccess	AWS managed - job function	3

Cisco Catalyst Center on AWS with Launchpad

AWS Access

Fill the AWS details to connect to your AWS account.

For more details,
check <https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html>



☒ IAM Login ☐ Federated Login

Access Key ID • ⓘ

.....

Secret Access Key • ⓘ

.....

☒ Use MFA authentication ⓘ

Authenticate

Cisco Catalyst Center on AWS with Launchpad

[IAM](#) > [Users](#) > cisco-DNA-launchpad

cisco-DNA-launchpad [Info](#) Delete

Summary

ARN arn:aws:iam::348397370333:user/cisco-DNA-launchpad	Console access Disabled	Access key 1 AKIAVCHQZ7POSVPHYGFY - Active ✔ Used 28 days ago. 28 days old.
Created October 10, 2023, 11:25 (UTC+11:00)	Last console sign-in -	Access key 2 Create access key

Permissions | Groups | Tags | **Security credentials** | Access Advisor

Console sign-in Enable console access

Console sign-in link https://348397370333.signin.aws.amazon.com/console	Console password Not enabled
---	---------------------------------

Multi-factor authentication (MFA) (1) Remove Resync Assign MFA device

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

Cisco Catalyst Center on AWS with Launchpad

aws Services Search [Option+S]

IAM > Users > cisco-DNA-launchpad > Assign MFA device

Step 1
Select MFA device

Step 2
Set up device

Select MFA device [Info](#)

MFA device name

Device name
Enter a meaningful name to identify this device.

MarksiPhone

Maximum 128 characters. Use alphanumeric and '+', '@', '-', '_' characters.

MFA device

Select an MFA device to use, in addition to your username and password, whenever you need to authenticate.

- ☒  **Authenticator app**
Authenticate using a code generated by an app installed on your mobile device or computer.
- ☐  **Security Key**
Authenticate using a code generated by touching a YubiKey or other supported FIDO security key.
- ☐  **Hardware TOTP token**
Authenticate using a code displayed on a hardware Time-based one-time password (TOTP) token.

Cisco Catalyst Center on AWS with Launchpad

AWS Access

Fill the AWS details to connect to your AWS account.

For more details,
check <https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html>



MFA Authentication

Please Select MFA Device • ⓘ

arn:aws:iam::348397370333:mfa/MarkRebetz ▾

MFA Code Verification

Please open your MFA authenticator app and enter the passcode into the designated field labeled "MFA Passcode."

MFA Passcode ⓘ

Cancel

Cisco Catalyst Center on AWS with Launchpad

Global Dashboard

Region
us-east-1

CloudWatch Dashboard

Dashboard

User Activities

Dashboard

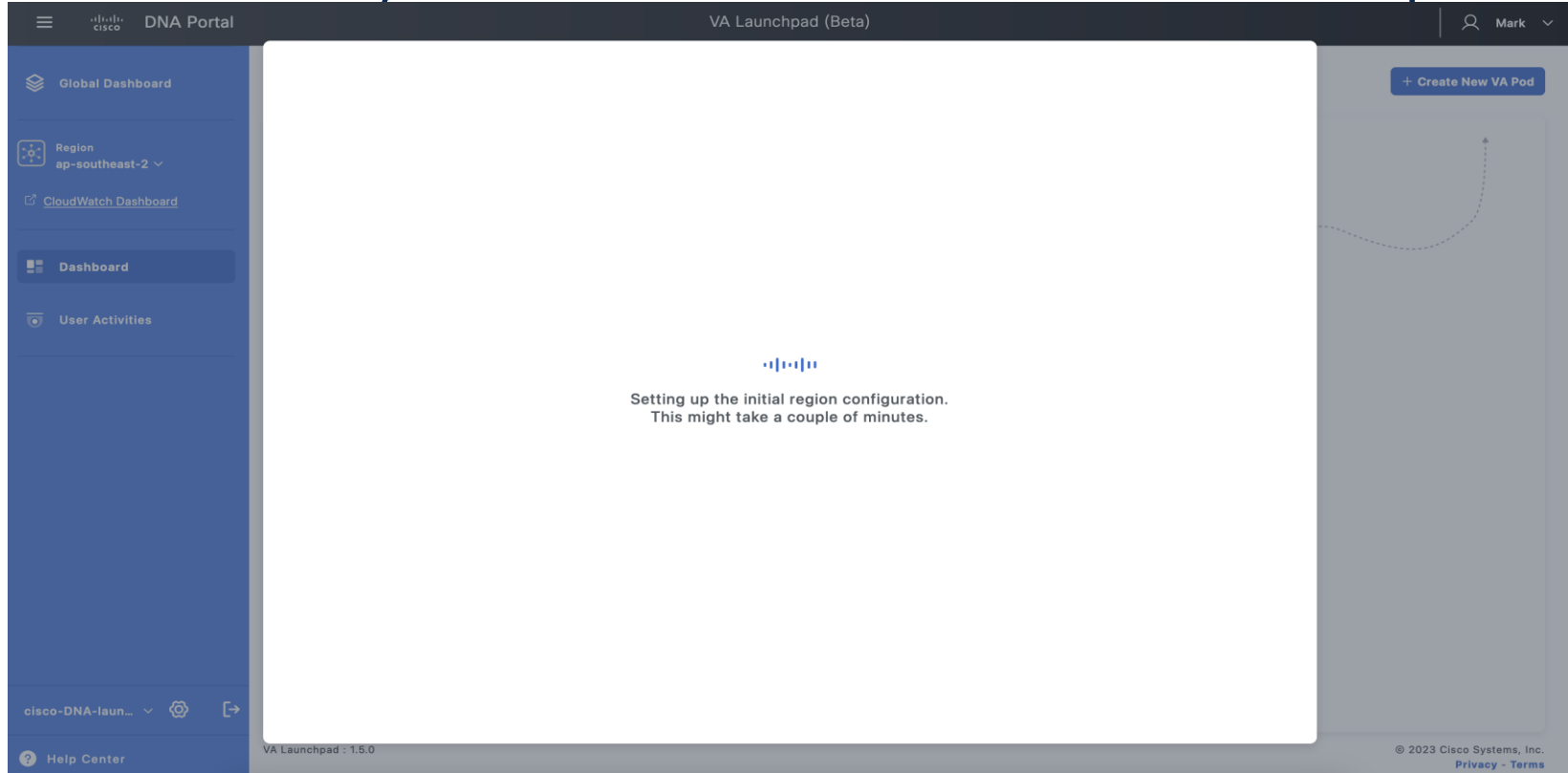
+ Create New VA Pod

No VA Pod(s) created!

You can create new Virtual Appliance (VA) pod by clicking the above button

- Before you begin, ensure that your AWS account is subscribed to [Cisco DNA Center Virtual Appliance - Bring Your Own Licence\(BYOL\)](#) in AWS Marketplace.
- Please make sure you have the following minimum resources to install Cisco DNA Center.
 - Cisco DNA Center Server: 32 vCPU, 256GB RAM, and 4TB storage available.
 - Cloud Backup Server: 2vCPU, 500 GB storage on t3 micro instance.
- VA Pod = AWS hosting environment for Cisco DNA Center Virtual Appliance which include collection of AWS resources such as Cisco DNA Center EC2 instance, EBS storage, backup NFS server, security groups, gateways, routing tables, etc.

Cisco Catalyst Center on AWS with Launchpad



Cisco Catalyst Center on AWS with Launchpad

The screenshot shows the 'Configure AWS Infrastructure' wizard in the Cisco DNA Portal. The left sidebar contains navigation links: Global Dashboard, Region (ap-southeast-2), CloudWatch Dashboard, Dashboard, and User Activities. The main content area is divided into three steps: 1. Configure AWS Infrastructure (With EC2, VPN Details), 2. Configure On-premise (Precheck with AWS), and 3. Network Connectivity Check (Check IPsec tunnel connection). The current step is 'Configure AWS Infrastructure', which includes sections for 'VA POD ENVIRONMENT DETAILS' and 'ON-PREM CONNECTIVITY'. The 'VA POD ENVIRONMENT DETAILS' section has input fields for 'VA Pod Name' (CiscoLive23), 'Availability Zone' (ap-southeast-2a), and 'AWS VPC CIDR' (10.0.44.0/25). The 'ON-PREM CONNECTIVITY' section has a 'Transit Gateway (TGW)' dropdown set to 'VPN GW', a 'VPN DETAILS' section with a 'CGW (Enterprise Firewall/Router)' input field, and a 'VPN Vendor' dropdown. The bottom of the wizard has an 'Exit' button and a 'Next' button. The right sidebar has a '+ Create New VA Pod' button and a graph area. The footer of the wizard shows the copyright notice: © 2023 Cisco Systems, Inc. Privacy - Terms.

Configure AWS Infrastructure
Please enter the following details.

VA POD ENVIRONMENT DETAILS

VA Pod Name • ⓘ

Availability Zone • ⓘ

AWS VPC CIDR • ⓘ
 /25

ON-PREM CONNECTIVITY
Please select the appropriate options for your on-premise network connectivity.

Transit Gateway (TGW) ⓘ

VPN DETAILS
Please complete this to download the sample configuration based on your enterprise gateway. Please note these are samples, and will need modification to use Advanced Algorithms, Certificates, and/or IPv6.

CGW (Enterprise Firewall/Router) • ⓘ

VPN Vendor • ⓘ

© 2023 Cisco Systems, Inc. Privacy - Terms

Cisco Catalyst Center on AWS with Launchpad

- 1 Configure AWS Infrastructure**
With EC2, VPN Details
- 2 Configure On-premise**
Precheck with AWS
- 3 Network Connectivity Check**
Check IPsec tunnel connection

Configure AWS Infrastructure

Please enter the following details.

VA POD ENVIRONMENT DETAILS

VA Pod Name * ⓘ

Availability Zone * ⓘ

AWS VPC CIDR * ⓘ

 /25

Cisco Catalyst Center on AWS with Launchpad

1 Configure AWS Infrastructure

With EC2, VPN Details

2 Configure On-premise

Precheck with AWS

3 Network Connectivity Check

Check IPSec tunnel connection

ON-PREM CONNECTIVITY

Please select the appropriate options for your on-premise network connectivity.

Transit Gateway (TGW)

VPN GW

New VPN GW + New TGW

Existing TGW

VPN DETAILS

Please complete this to download the sample configuration based on your enterprise gateway. Please note these are samples, and will need modification to use Advanced Algorithms, Certificates, and/or IPv6.

CGW (Enterprise Firewall/Router)

144.6.176.223

VPN Vendor

Cisco Meraki

Platform

MX Series

Software

15.12+ (WebUI)

CUSTOMER PROFILE

Small

Medium

Large

BACKUP TARGET

Enterprise Backup (NFS)

Cloud Backup (NFS)

Exit

Next

Cisco Catalyst Center on AWS with Launchpad

1 **Configure AWS Infrastructure**
With EC2, VPN Details

2 **Configure On-premise**
Precheck with AWS

3 **Network Connectivity Check**
Check IPSec tunnel connection

Summary

Review your AWS Infrastructure details and make changes. If you are satisfied with your selection, click the "Start Configuring AWS Infrastructure"

VA POD ENVIRONMENT DETAILS

VA Pod Name	CiscoLive23
Region	ap-southeast-2
Availability Zone	ap-southeast-2a
AWS VPC CIDR	10.0.44.0/25

ON-PREM CONNECTIVITY

Transit Gateway (TGW)	VPN GW
-----------------------	--------

VPN ATTACHMENT

Customer Gateway (CGW)	New VPN GW
------------------------	------------

VPN DETAILS

CGW (Enterprise Firewall/Router)	144.6.176.223
VPN Vendor	Cisco Meraki
Platform	MX Series
Software	15.12+ (WebUI)

OTHER DETAILS

Customer Profile	Medium
Backup Target	Cloud Backup (NFS)

Exit

Back

Start Configuring AWS Infrastructure

Cisco Catalyst Center on AWS with Launchpad

1

Configure AWS Infrastructure

With EC2, VPN Details

2

Configure On-premise

Precheck with AWS

3

Network Connectivity Check

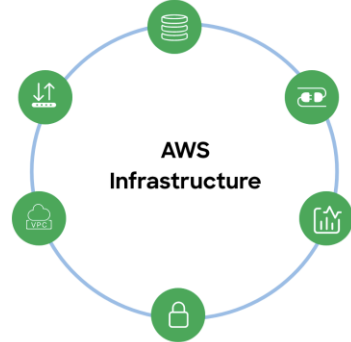
Check IPSec tunnel connection

AWS Infrastructure Configured

- ✓ CiscoLive23
AWS CloudFormation
- ✓ PrivateRouteTable1
AWS EC2
- ✓ PrivateSubnet1
AWS EC2
- ✓ VPC
AWS EC2
- ✓ CiscoLive23-OnPremConnectivity
AWS CloudFormation
- ✓ VpcVpnConnectionPrimary
AWS EC2
- ✓ VpcCustomerGateway
AWS EC2
- ✓ VpcVpnGateway
AWS EC2
- ✓ CiscoLive23-LambdaFunctions
AWS CloudFormation

Exit

Proceed to On-Prem Configuration



A circular diagram with the text "AWS Infrastructure" in the center. Six green circular icons are arranged in a ring around the center, connected by a light blue line. The icons represent: a database cylinder (top), a server rack (top-right), a bar chart (right), a padlock (bottom-right), a cloud with a checkmark (bottom), and a double-headed arrow (left).

Cisco Spaces Connector





Deploying Cisco Spaces Connector

Deployment Options

1. Manual EC2 Instance deployment
2. Terraform

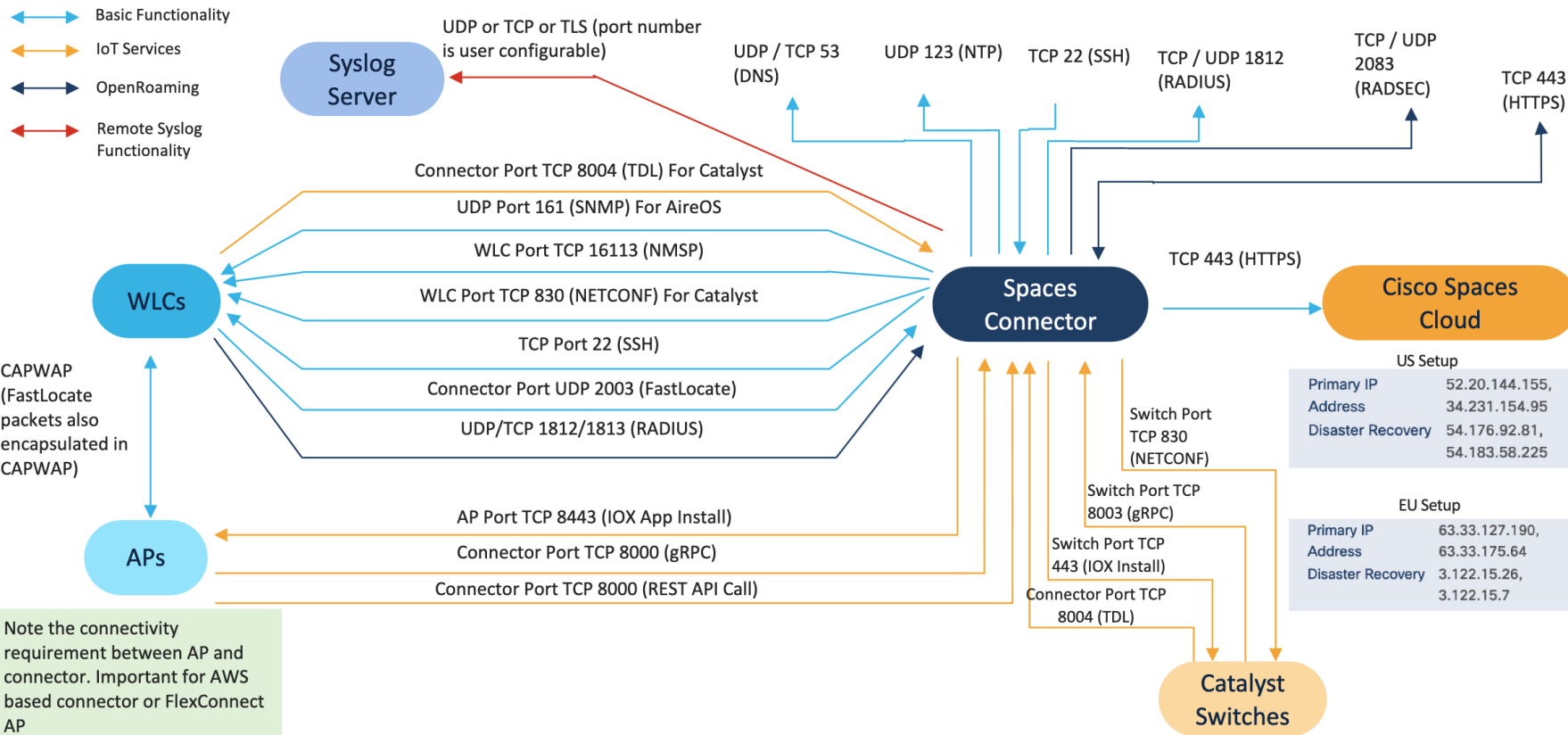
Cisco Spaces Wireless Setup

1. Deploy Spaces connector in AWS
2. Configure the Spaces Connector – Token
3. Add Cisco C9800 WLC
4. Import Catalyst Center Maps

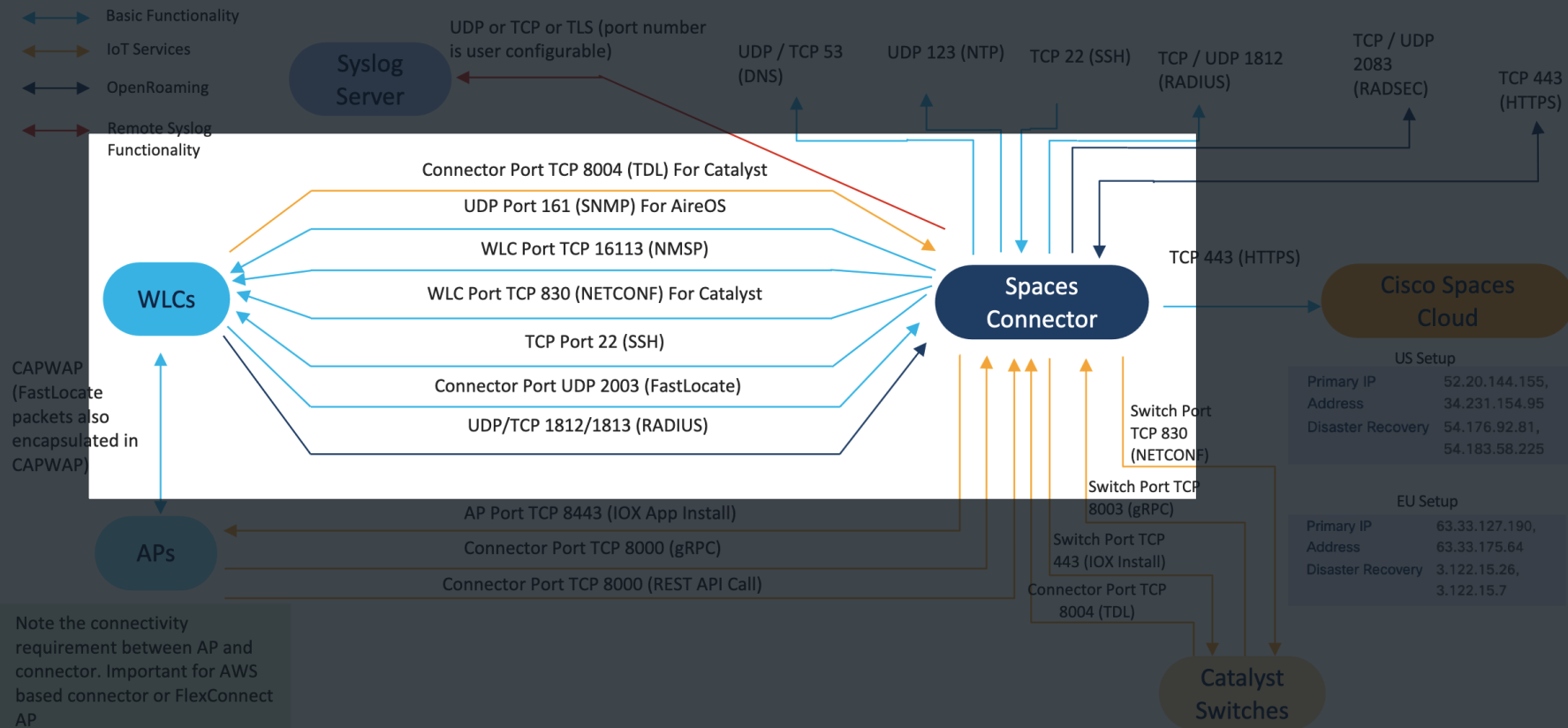
The screenshot shows the AWS Marketplace interface for the Cisco DNA Spaces Connector. The top navigation bar includes links for About, Categories, Delivery Methods, Solutions, AWS IQ, Resources, Your Saved List, Partners, Sell in AWS Marketplace, and Amazon Web S. The main header features the Cisco logo and the product name 'Cisco DNA Spaces Connector' with a 'Continue to Launch' button. Below the header, there are tabs for '< Product Detail', 'Subscribe', and 'Configure'. The 'Configure this software' section prompts the user to 'Choose a fulfillment option and software version to launch this software.' It contains three dropdown menus: 'Fulfillment option' (64-bit (x86) Amazon Machine Image (AMI)), 'Software version' (Cisco DNA Spaces Connector 3 (Mar 14, 2023)), and 'Region' (US West (N. California)). Below these, it shows the 'Ami Id: ami-08f057e7c6fcc89d6' and 'Ami Alias: /aws/service/marketplace/prod-zjxlo4taqlbpe/cisco-dna-spaces-connector-3' with a 'Learn More' link and a 'New' badge. On the right, a 'Pricing information' box provides an estimate of costs, including 'Software Pricing' at \$0/hr and 'Infrastructure Pricing' at \$81.00/month for 1 * m5.large EC2 instance.

EC2 Configuration Guide: https://www.cisco.com/c/en/us/td/docs/wireless/spaces/connector/config/b_connector_30/m_ami3.html

Cisco Spaces Connector Ports Used

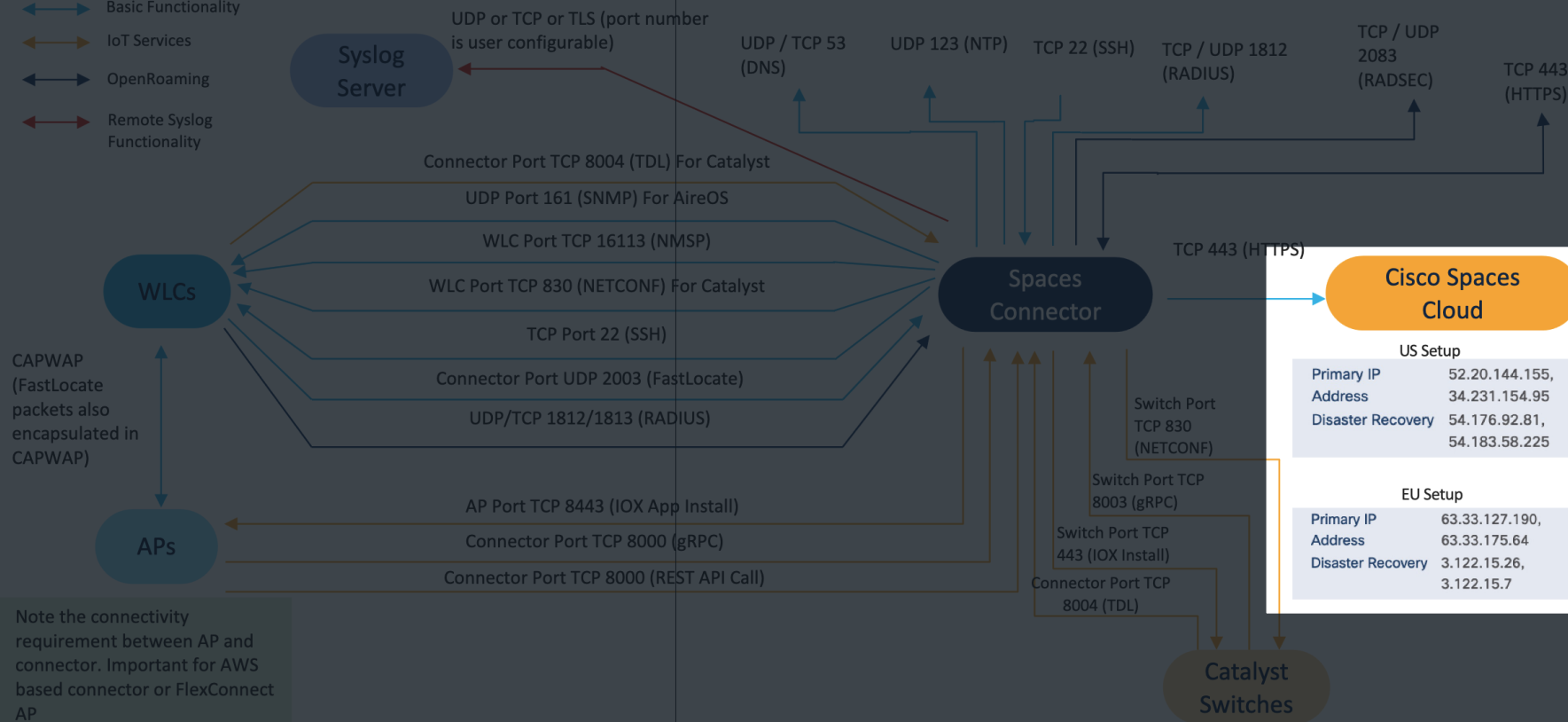


Cisco Spaces Connector Ports Used



Cisco Spaces Connector Ports Used

- Basic Functionality
- IoT Services
- OpenRoaming
- Remote Syslog Functionality



US Setup	
Primary IP Address	52.20.144.155, 34.231.154.95
Disaster Recovery	54.176.92.81, 54.183.58.225

EU Setup	
Primary IP Address	63.33.127.190, 63.33.175.64
Disaster Recovery	3.122.15.26, 3.122.15.7

Configure Spaces Connector – Terraform

Deploy Spaces connector in AWS

```
#main.tf
resource "aws_instance" "CiscoLive-SpacesConnector" {
    ami                        = var.SpacesConnector-ami
    instance_type             = var.spaces_instance_type
    subnet_id                 = var.subnet_id
    private_ip                = var.connector-staticIPv4
    associate_public_ip_address = true

    vpc_security_group_ids    = [aws_security_group.CiscoLive-securityGroup.id]
    key_name                  = var.key_name
    tags = {
        Name = "MRE_SpacesConnector"
    }
    credit_specification {
        cpu_credits = "unlimited"
    }
}
```

Deploy Spaces connector in AWS – Security-Group

```
#main.tf
resource "aws_security_group" "CiscoLive-ConnectorAllTraffic" {
  name           = "MRE Cisco Live ConenctorGroup"
  description    = "Allow All Traffic to only Connector"
  vpc_id         = var.vpc_id

  ingress {
    description    = "All Traffic from 10.0.43.0/24"
    from_port      = 443
    to_port        = 443
    protocol       = "all"
    cidr_blocks    = [var.all_subnets]
  }
}
```

Deploy Spaces connector in AWS – ssh to change pass

```
~ % ssh ec2-user@10.0.42.11 -I ~/.ssh/mre_ciscoLive_aws.pem
New password:
BAD PASSWORD: The password is shorter than 14 characters
Retype new password:
passwd: all authentication tokens updated successfully.
Password changed successfully

Cisco Spaces Connector UI:
https://10.0.42.11
Username log in: spacesadmin
The install is complete, a reboot will occur in 10 seconds...
Connection to 10.0.42.11 closed by remote host.
Connection to 10.0.42.11 closed.
```

Terraform Structure – ssh to change password

Dashboard

Configure Connector

Configure HTTP proxy

Privacy Settings

Manage API Keys

Troubleshoot

⚠️ Configure Token

Without the token, the connector will not be able to start.

Configure Token ×

Connector 3.1

Hostname
mre-spaces-ciscolive

Package
connector3-p84-...

Show More

General Information

Connector Name	Not Available	HA Config Mode	Not VIP Paired
Tenant ID	Not Available		
Connector ID	Not Available		
Instance ID	not_available		
Proxy	Not Available		
NTP Address	169.254.169.123		
NTP Status	active (running)		

Configure the Spaces Connector – Token



Connector 3.1



Dashboard

Configure Connector

Configure HTTP proxy

Privacy Settings

Manage API Keys

Troubleshoot

Configure the token below to activate your connector

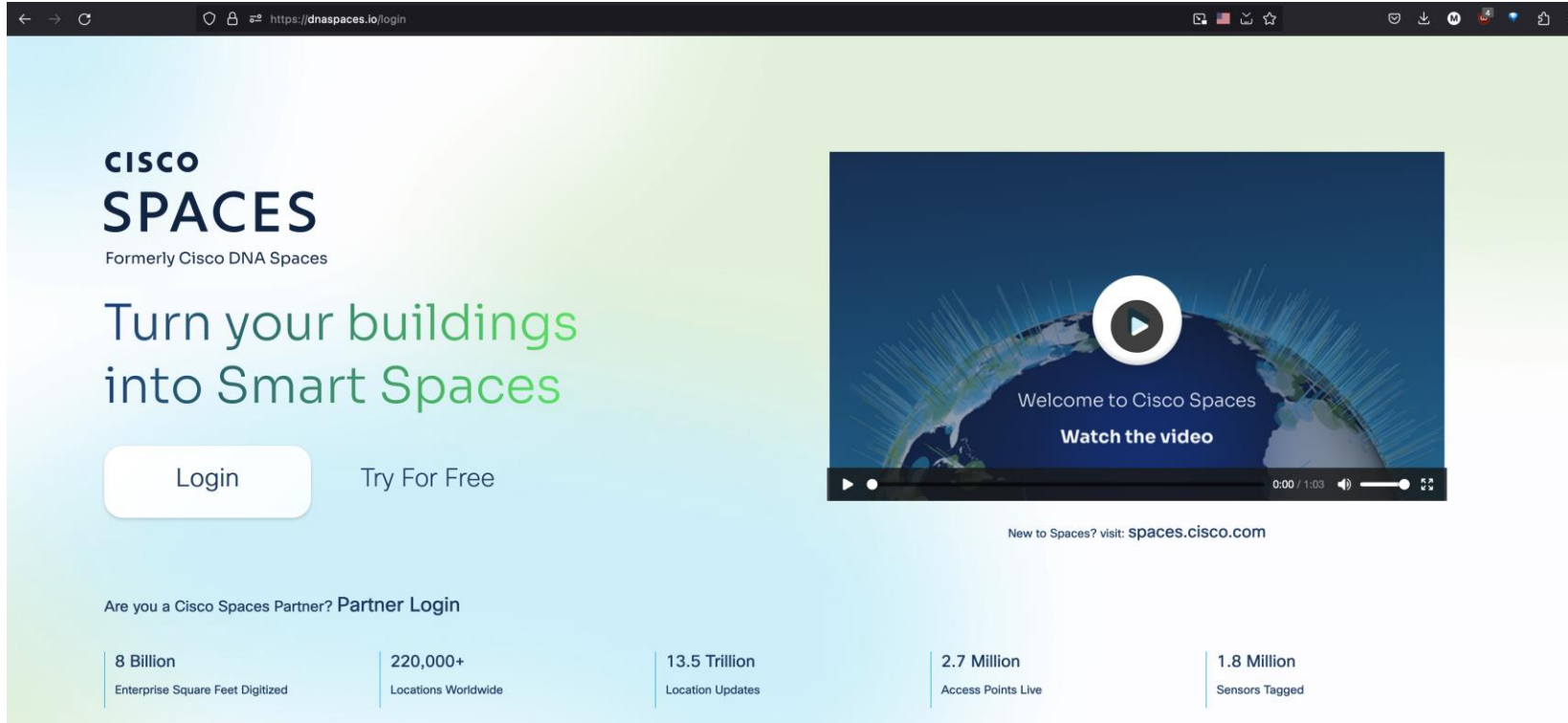
Note: If the machine is behind a proxy, Connector won't be able to interact with the cloud. [Configure Proxy](#) to get the connector working.

Token

Configure

Don't have the token? [Click Here for the Step by Step Guide](#) ▾

Configure the Spaces Connector – Token



The screenshot shows the Cisco Spaces website. The header includes navigation icons and the URL <https://dnaspaces.io/login>. The main content area features the Cisco Spaces logo, the tagline "Formerly Cisco DNA Spaces", and the headline "Turn your buildings into Smart Spaces". Below this are "Login" and "Try For Free" buttons. A video player on the right shows a "Welcome to Cisco Spaces" video with a "Watch the video" button. Below the video player is a link for new users: "New to Spaces? visit: spaces.cisco.com". At the bottom, a statistics section lists five metrics: 8 Billion Enterprise Square Feet Digitized, 220,000+ Locations Worldwide, 13.5 Trillion Location Updates, 2.7 Million Access Points Live, and 1.8 Million Sensors Tagged.

CISCO SPACES
Formerly Cisco DNA Spaces

Turn your buildings into Smart Spaces

Login Try For Free

Welcome to Cisco Spaces
Watch the video

New to Spaces? visit: spaces.cisco.com

Are you a Cisco Spaces Partner? [Partner Login](#)

8 Billion Enterprise Square Feet Digitized	220,000+ Locations Worldwide	13.5 Trillion Location Updates	2.7 Million Access Points Live	1.8 Million Sensors Tagged
---	---------------------------------	-----------------------------------	-----------------------------------	-------------------------------

Configure the Spaces Connector – Token

The screenshot displays the Cisco Spaces dashboard interface. The top navigation bar is dark blue with the Cisco logo and 'Spaces' text on the left, and a help icon and user profile icon on the right. The left sidebar contains a list of navigation items: Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup. The 'Setup' item is highlighted with a red rectangular border. The main content area is titled 'Digitization Stats' and features four status cards: '1 Locations', 'No access points data available', 'No location updates data available', and 'No square foot data available'. Below these cards is a section titled 'SEE Apps' with a search icon and a 'Show/Hide Apps' dropdown. This section contains four app tiles: 'Behavior Metrics' (Gold standard performance metrics for physical spaces), 'Right Now' (Real time metrics), 'Camera Metrics' (Metrics derived from Meraki Camera Data), and 'OpenRoaming' (Stay connected wherever you go, seamlessly and securely).

Configure the Spaces Connector – Token

The screenshot displays the Cisco Spaces dashboard interface. On the left, a sidebar contains navigation links: Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup. The 'Wireless Networks' link under 'IoT Services' is highlighted with a red rectangle. The main content area is titled 'Digitization Stats' and shows a map with 1 location. Below this, there are four status cards: 'No access points data available', 'No location updates data available', 'No square foot data available', and 'No visits data available'. A 'SEE Apps' button is present, followed by a search bar and a 'Show/Hide Apps' dropdown. Four app tiles are displayed: 'Behavior Metrics' (Gold standard performance metrics for physical spaces), 'Right Now' (Real time metrics), 'Camera Metrics' (Metrics derived from Meraki Camera Data), and 'OpenRoaming' (Stay connected wherever you go, seamlessly and securely).

Configure the Spaces Connector – Token

The screenshot shows the Cisco Spaces web interface. On the left is a navigation sidebar with links to Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup. The main content area is titled 'Connect your wireless network' and contains a section 'Connect via Spaces Connector'. This section has a sub-header 'Connect via Spaces Connector' and a description: 'Spaces Connector is an easy way to get your wireless network connected to Cisco Spaces. No need to upgrade Cisco Wireless Controllers or reconfigure your wireless network.' Below this are four numbered steps: 1. Install Spaces Connector OVA, 2. Configure Spaces Connector (highlighted with a red box), 3. Add Controllers, and 4. Import Maps. Step 2 includes a text box for configuration and links for 'Create Connector' and 'View Connectors'. Step 3 includes a text box for controller status and links for 'Add Controllers' and 'View Controllers'. Step 4 includes a text box for map import status and links for 'Import/Sync Maps', 'Map Upload History', and 'Manage Maps'. On the right side of the main content area, there is a 'Need Help?' section with links to 'View Configuration Steps', 'System Requirements', and 'Frequently Asked Questions'.

Connect your wireless network

Connect via Spaces Connector

Spaces Connector is an easy way to get your wireless network connected to Cisco Spaces. No need to upgrade Cisco Wireless Controllers or reconfigure your wireless network.

- 1 Install Spaces Connector OVA**
Download and install Spaces Connector OVA as a virtual machine.
[Download Spaces Connector](#)
- 2 Configure Spaces Connector**
You will need a token to configure Spaces Connector. You need to connect to `https://<your connector IP>/` from a browser to configure the token. You can optionally configure Spaces Connector to connect via HTTPS proxy.

[Create Connector](#)
[View Connectors](#)
- 3 Add Controllers**
Add and associate controllers to your Cisco Spaces Connector(s)

0 controller(s) active

[Add Controllers](#)
[View Controllers](#)
- 4 Import Maps**
Prime/DNAC map requires in order to work Locate & detect, Asset tracker, and IOT services, and proximity Report

0 buildings imported
0 floors imported

[Import/Sync Maps](#)
[Map Upload History](#)
[Manage Maps](#)

Need Help?
Access the below links to view detailed help.
[View Configuration Steps](#)
[System Requirements](#)
[Frequently Asked Questions](#)

Configure the Spaces Connector – Token

The image shows a split-screen view. On the left is the Cisco Spaces dashboard with a sidebar menu containing: Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup (highlighted). The main content area is titled 'Connect your wireless network' and lists four steps: 1. Install Spaces Connector, 2. Configure Spaces Connector, 3. Add Controllers, and 4. Import Maps. On the right is the 'Create Connector' configuration page. It has a title bar with a close button (X). The 'Spaces Connector Name' field is set to 'AWS-Connector'. Under 'Connector Version', 'Version 3.0 (Beta)' is selected with a radio button. Below this, there are three bullet points: 'Support for deploying and managing multiple individual services', 'Enhanced monitoring and troubleshooting of the connector and connector services', and 'Seamless services and system upgrades'. A link to the 'Connector 3.0 Configuration Guide' is provided. At the bottom, there is a checkbox for 'Enable Location Services' which is checked. The page has 'Cancel' and 'Save' buttons at the bottom.

Create Connector

Spaces Connector Name

AWS-Connector

Connector Version

☐ Version 2.x
First generation Connector designed to transfer location data efficiently to Cisco Spaces cloud

☒ **Version 3.0 (Beta)**

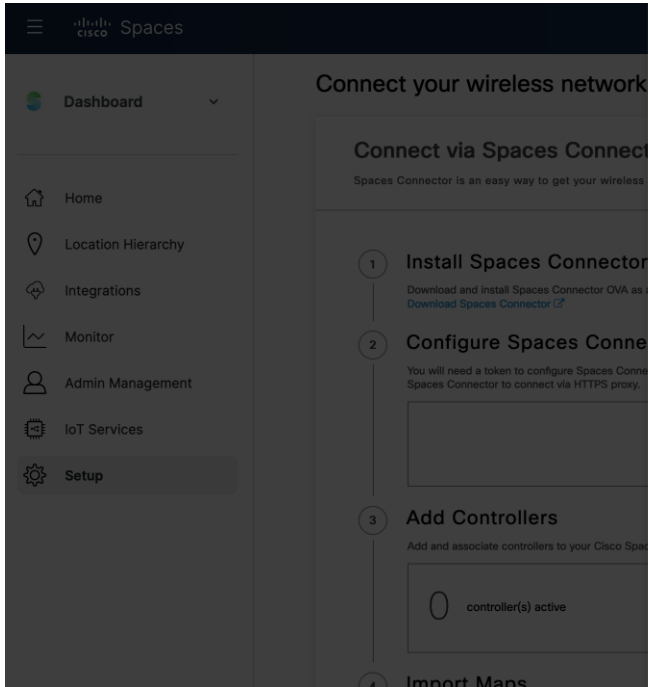
- Support for deploying and managing multiple individual services
- Enhanced monitoring and troubleshooting of the connector and connector services
- Seamless services and system upgrades

• Refer to the Connector 3.0 [Configuration Guide](#) for more details

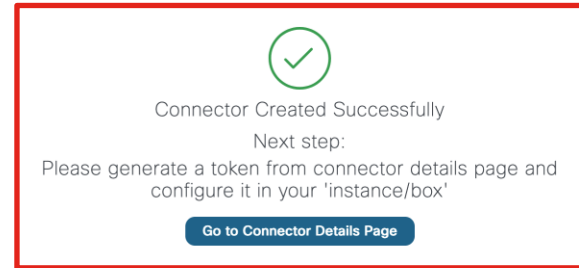
☒ **Enable Location Services**

[Cancel](#) [Save](#)

Configure the Spaces Connector – Token



Create Connector



Configure the Spaces Connector – Token

The screenshot shows the Cisco Spaces AWS-Connector configuration interface. The top navigation bar includes the Cisco Spaces logo and a user profile icon. The left sidebar contains a menu with options: Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup (which is highlighted). The main content area shows the breadcrumb 'Setup > Connectors > AWS-Connector' and the ID '26559748012616850000' with a last modified timestamp of 'Oct 20, 2023, 4:08:31 PM'. A summary section displays metrics: 0 Instances, 0 Active, 0 Inactive, 2 Services enabled, 0 Controller, and 0 Switches. Below this, the 'Configuration' tab is selected, showing a 'Generate Token' button (highlighted with a red box) and a 'Troubleshoot Connector' button. A 'Services' table lists the following:

Service Name	Version	Last Updated	Actions
Service Manager	3.1.1.52	Oct 20, 2023, 4:08:31 PM	
Location	3.1.0.102	Oct 20, 2023, 4:08:31 PM	

An 'Add Services' button is located at the bottom right of the services table.

CISCO *Live!*



© 2023 Cisco and/or its affiliates. All rights reserved. Cisco Public

130

×

[illegible][Copy Token](#) | [View Documentation](#)

To set up your connector instance, follow the steps below.

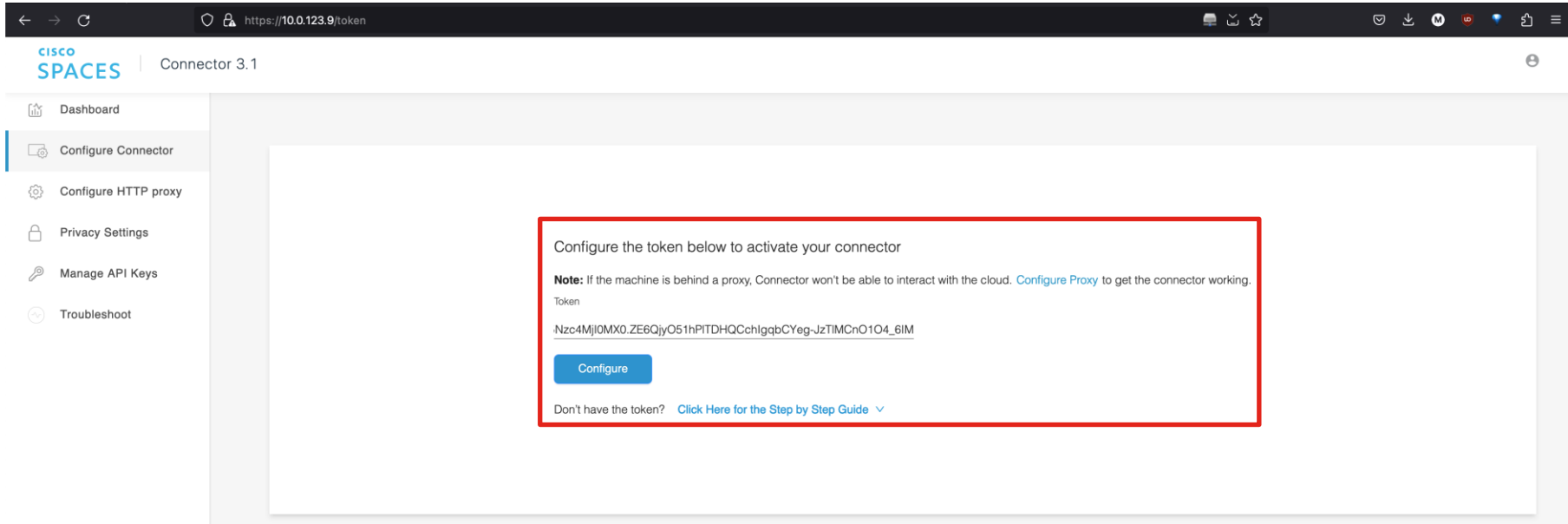
Step 1:

Copy the generated token above.

Step 2:

Login to your connector UI and configure the token. Follow the [documentation](#) if you haven't setup your connector yet.

Configure the Spaces Connector – Token



The screenshot shows a web browser window with the address bar displaying `https://10.0.123.9/token`. The page title is "Cisco SPACES | Connector 3.1". On the left, a sidebar menu contains the following items: "Dashboard", "Configure Connector" (highlighted with a blue bar), "Configure HTTP proxy", "Privacy Settings", "Manage API Keys", and "Troubleshoot". The main content area displays a configuration box for the token, which is highlighted with a red border. Inside this box, the text reads: "Configure the token below to activate your connector". Below this is a **Note**: "If the machine is behind a proxy, Connector won't be able to interact with the cloud. [Configure Proxy](#) to get the connector working." The label "Token" is followed by the token string: `Nzc4Mjl0MX0.ZE6QjyO51hPITDHQCchlqgbCYeg-JzTIMCnO1O4_6lM`. A blue "Configure" button is positioned below the token. At the bottom of the box, it says "Don't have the token? [Click Here for the Step by Step Guide](#)".

Configure the Spaces Connector – Token

 | Connector 3.1

Dashboard

Configure Connector

Configure HTTP proxy

Privacy Settings

Manage API Keys

Troubleshoot



Connector 3.1

Hostnameaws-connectora
Packageconnector3-p84-...

Show More

General Information

Connector Name	aws-connector	HA Config Mode	Not VIP Paired
Tenant ID	19715		
Connector ID	9424810492200851000		
Instance ID	06a288a425bd		
Proxy	Not Available		
NTP Address	Not Available		
NTP Status	Not Available		

Primary Interface

IP Address	10.0.42.11, 13.210.205.201
MAC Address	06:a2:88:a4:25:bd
Gateway	10.0.42.1
DNS Server	ec2-13-210-205-201.ap-southeast-2.compute.amazonaws.com
Domain	Not Available
IP Stack	ipv4

Configure the Spaces Connector – Token

Services

**Service Manager**  3.1.1.52 

Upgrade: Success

Up time	6m 32s ⓘ
Control Channel	Up
CPU Usage (%)	6.15 % ⓘ
Memory Usage (%)	1.23 % ⓘ
Memory Usage	97.4 MB ⓘ
Disk Usage (%)	0 % ⓘ
Disk Size	0.68 MB ⓘ

Show More

**location** 3.1.0.102  

Upgrade: Success

Last Heartbeat	8s ago
Running Status	Up
Up time	2m 43s ⓘ
Outgoing message rate	0.9 msgs/sec ⓘ
Outgoing dropped message rate	0 msgs/sec ⓘ
Incoming NMSP message rate	0 msgs/sec ⓘ
Number of Access Points.	0 Count ⓘ

Show More

Configure the Spaces Connector – Add WLC

Configure the Spaces Connector – Add WLC

The screenshot shows the Cisco Spaces web interface at the URL <https://dnaspaces.io/setup/wirelessnetwork>. The page title is "Connect your wireless network". Below this, there's a section "Connect via Spaces Connector" with a sub-header "Spaces Connector is an easy way to get your wireless network connected to Cisco Spaces. No need to upgrade Cisco Wireless Controllers or reconfigure your wireless network."

The setup process is divided into three numbered steps:

- 1 Install Spaces Connector OVA**
Download and install Spaces Connector OVA as a virtual machine.
[Download Spaces Connector](#)
- 2 Configure Spaces Connector**
You will need a token to configure Spaces Connector. You need to connect to `https://-your connector IP>/` from a browser to configure the token. You can optionally configure Spaces Connector to connect via HTTPS proxy.
1 / 1 connector(s) active
[Create Connector](#)
[View Connectors](#)
- 3 Add Controllers**
Add and associate controllers to your Cisco Spaces Connector(s)
0 controller(s) active
[Add Controllers](#)
[View Controllers](#)

The third step, "Add Controllers", is highlighted with a red border. On the right side, there's a "Need Help?" section with links to [View Configuration Steps](#), [System Requirements](#), and [Frequently Asked Questions](#).

Configure the Spaces Connector – Add WLC

The screenshot displays the Cisco Spaces web interface. On the left is a navigation sidebar with links to Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup. The main content area is titled 'Connect your wireless network' and contains a sequence of steps: 1. Install Spaces Connector OVA, 2. Configure Spaces Connector, 3. Add Controllers, and 4. Import Maps. Step 3 is currently active, showing '0 controller(s) active'. A modal window titled 'Add Controller' is open over step 3. This modal contains the following fields: 'Connectors' (a dropdown menu showing 'aws-connector'), 'Add new controller' (selected with a radio button), 'Controller IP' (text field with '10.0.42.10'), 'Controller Name' (text field with 'aws-c9800'), 'Controller Type' (dropdown menu showing 'Catalyst 9800 Wireless Co...'), 'Netconf Username' (text field with 'Netconf'), and 'Netconf Password' (password field with masked characters and a 'SHOW' button). At the bottom of the modal are two buttons: 'Save & Close' and 'Save & Add Next Controller'.

Add Controller

Connectors

aws-connector

☒ Add new controller
Choose this option to add new controllers

☐ Choose controllers from another connector
Choose this option to add controllers from another connector. Controllers will be removed from original connector

Controller IP

10.0.42.10

Controller Name

aws-c9800

Controller Type

Catalyst 9800 Wireless Co...

Netconf Username

Netconf

Netconf Password

SHOW

Save & Close Save & Add Next Controller

Configure the Spaces Connector – Add WLC

The screenshot shows the Cisco Spaces Setup interface. The left sidebar contains navigation links: Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup (highlighted). The main content area is titled 'Connect your wireless network' and shows a progress bar with four steps: 1. Install Spaces Connector OVA, 2. Configure Spaces Connector, 3. Add Controllers, and 4. Import Maps. Step 3 is currently active, showing '0 controller(s) active' and a 'Add Controller' button. A red box highlights the 'Add Controller' modal window, which contains the following information:

Add Controller

Enter the password [SHOW](#)

Catalyst 9800 Wireless Controller CLI Commands

```
en
conf t
nmsp enable
aaa new-model
username 06a288a425bd mac aaa attribute list cmx_06a288a425bd
aaa attribute list cmx_06a288a425bd
attribute type password a7512bc7c364fd58d3735b1d8741f6782473389672dd03ba026592a2df416643
aaa authorization credential-download wcm_loc_serv_cert local
```

[Test Connectivity](#)

[Save & Close](#) [Save & Add Next Controller](#)

Configure the Spaces Connector – Add WLC

Catalyst 9800 Wireless Controller CLI Commands



```
en

conf t

nmsp enable

aaa new-model

username 005056A5DC5A mac aaa attribute list cmx_005056A5DC5A

aaa attribute list cmx_005056A5DC5A

attribute type password f004b37f72ea65be520f5f212308a6732078e46b33e33fa39b01b92cb81de4b4

aaa authorization credential-download wcm_loc_serv_cert local
```

Configure the Spaces Connector – Add WLC

The screenshot shows the Cisco Spaces web interface. The top navigation bar is dark blue with the Cisco logo and 'Spaces' text. A left sidebar contains a menu with icons and labels: Dashboard, Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup (highlighted). The main content area is titled 'Connect your wireless network' and features a section 'Connect via Spaces Connector' with a sub-header 'Connect via Spaces Connector' and a description: 'Spaces Connector is an easy way to get your wireless network connected to Cisco Spaces. No need to upgrade Cisco Wireless Controllers or reconfigure your wireless network.' Below this, there are three numbered steps: 1. 'Install Spaces Connector OVA' with a link to 'Download Spaces Connector'; 2. 'Configure Spaces Connector' with a description about needing a token and a link to 'Create Connector'; 3. 'Add Controllers' with a description about adding and associating controllers and a link to 'Add Controllers'. Each step has a progress indicator showing '1 / 1' and 'connector(s) active' or 'controller(s) active'. On the right side of the main content area, there is a 'Need Help?' section with links to 'View Configuration Steps', 'System Requirements', and 'Frequently Asked Questions'.

Dashboard

Home

Location Hierarchy

Integrations

Monitor

Admin Management

IoT Services

Setup

Connect your wireless network

Connect via Spaces Connector

Spaces Connector is an easy way to get your wireless network connected to Cisco Spaces. No need to upgrade Cisco Wireless Controllers or reconfigure your wireless network.

- 1 Install Spaces Connector OVA**

Download and install Spaces Connector OVA as a virtual machine.
[Download Spaces Connector](#)
- 2 Configure Spaces Connector**

You will need a token to configure Spaces Connector. You need to connect to [https://<your connector IP>/](#) from a browser to configure the token. You can optionally configure Spaces Connector to connect via HTTPS proxy.

1 / 1 connector(s) active

[Create Connector](#)
[View Connectors](#)
- 3 Add Controllers**

Add and associate controllers to your Cisco Spaces Connector(s)

1 / 1 controller(s) active

[Add Controllers](#)
[View Controllers](#)

Need Help?

Access the below links to view detailed help.

[View Configuration Steps](#)

[System Requirements](#)

[Frequently Asked Questions](#)

Verify on Controller – NMSP

```
aws-c9800#show nmsp status
```

```
NMSP Status
```

```
-----
```

DNA Spaces/CMX IP Address	Active	Tx	Echo	Resp	Rx	Echo	Req	Tx Data	Rx Data	Transport

10.0.42.11	Active	4	4	6	6	TLS				

Configure the Spaces Connector – Catalyst Center Maps

Configure the Spaces Connector – Add Catalyst Center Maps

Import or Sync Maps

Select source map

You must upload a Prime/DNAC map in order to work Detect & Locate, Asset Tracker, and IoT Services seamlessly. Choose any of option below

Prime

Choose this option if you have maps which you have downloaded from prime

DNAC v1.3.1 and above

Choose this option if you have maps which you have DNAC map **version 1.3.1 and above**

I don't have map

Choose this option if you dont not have a map

Cancel Next

Configure the Spaces Connector – Add Catalyst Center Maps

The screenshot displays the Cisco DNA Center interface in the 'Design / Network Hierarchy' view. The top navigation bar shows the Cisco DNA Center logo and the current view. A search bar is present on the right. The left sidebar contains a search hierarchy and a list of sites. The main map area shows a map of Australia with various locations marked. A context menu is open over the map, listing actions: Add Area, Add Building, Import Sites, Export Sites, Import Maps, Import Bulk AP, Export Maps, and View Settings. The map shows locations like Alice Springs, Rockhampton, Brisbane, Armidale, Dubbo, Griffith, Wangaratta, Port Lincoln, Whyalla, Broken Hill, Kalgoorlie, and J. B. T. 4. The 'Great Australian Bight' is also labeled. The bottom right of the map has zoom controls.

Configure the Spaces Connector – Add Catalyst Center Maps

Export Maps

File to be saved to*

DNAC_Map_Archive_10.66.212.109

Export Format ☐ Ekahau Project ☒ Prime

Ekahau Export is allowed for site that contains only buildings

Do you still want to continue with data export? Click **Export** to proceed. The file will be automatically downloaded once export is complete.

Cancel

Export

Configure the Spaces Connector – Add Catalyst Center Maps

Import or Sync Maps

×



Map uploaded successfully

Map provisioning started, check Upload History section for upload status

Continue

Configure the Spaces Connector – Manage Location Hierarchy

The screenshot displays the Cisco Spaces web interface. The top navigation bar includes the Cisco logo, the word "Spaces", and user profile icons. A search bar is located on the left side of the main content area. Below the search bar, a sidebar lists a hierarchy of locations under the "MarkRebetzke" group, including "250 Victoria Square Adelaide...", "ACT", "Ekahau Import", "Global", "Melbourne", "NSW", "PI migration", "SA", "Shepparton", "WA", and "Xavier College". The main content area features a header for "MarkRebetzke" with a briefcase icon and a summary of counts: 11 Campus, 2 Group, 12 Building, 27 Floor, 5 Zone, and 1 Alert. Below this, there are tabs for "Map", "Location Info", "Network Devices", and "Metadata". The "Map" tab is active, showing a map of Australia with several blue location pins. The pins are numbered 1 through 7, with pin 1 located in the southwest, pin 2 in the south, pin 3 in the southeast, and pins 4 through 7 in the northeast. The map also shows state boundaries and major cities like Sydney and Melbourne.

Configure the Spaces Connector – Add Maps

The screenshot displays the Cisco Spaces web interface. The top navigation bar includes a hamburger menu, the 'Spaces' logo, and user profile icons. The left sidebar contains a 'Dashboard' dropdown and a list of navigation items: Home, Location Hierarchy, Integrations, Monitor, Admin Management, IoT Services, and Setup (which is highlighted). The main content area is divided into four numbered steps:

- 1 Install Spaces Connector OVA**
Download and install Spaces Connector OVA as a virtual machine.
[Download Spaces Connector ?](#)
- 2 Configure Spaces Connector**
You will need a token to configure Spaces Connector. You need to connect to `https://<your connector IP>/` from a browser to configure the token. You can optionally configure Spaces Connector to connect via HTTPS proxy.

1 / 1 connector(s) active

[Create Connector](#)
[View Connectors](#)
- 3 Add Controllers**
Add and associate controllers to your Cisco Spaces Connector(s)

1 / 1 controller(s) active

[Add Controllers](#)
[View Controllers](#)
- 4 Import Maps**
Prime/DNAC map requires in order to work Locate & detect, Asset tracker, and IOT services, and proximity Report

12 buildings imported
27 floors imported

[Import/Sync Maps](#)
[Map Upload History](#)
[Manage Maps](#)

On the right side, there is a 'Need Help?' section with the text 'Access the below links to view detailed help.' and three links: [View Configuration Steps](#), [System Requirements](#), and [Frequently Asked Questions](#), each with an external link icon.

Wireless Infrastructure – Identity Services Engine (ISE)





Deploying Cisco ISE

Deployment Options

1. Manual EC2 Instance deployment
2. Terraform
3. CloudFormation Template

The screenshot shows the AWS Partner Solutions page for the 'Cisco ISE on AWS' solution. The page is titled 'What you'll build' and lists the components of the solution:

- The Partner Solution sets up the following:
 - A highly available architecture that spans two Availability Zones.*
 - A virtual private cloud (VPC) configured with public and private subnets, according to AWS best practices, to provide you with your own virtual network on AWS.*
 - In the public subnets, managed NAT gateways to allow outbound internet access for resources in the private subnets.*
 - In the private subnets:
 - One Cisco ISE primary and one Cisco ISE secondary node deployed to Amazon Elastic Compute Cloud (Amazon EC2) instances.
 - A Network Load Balancer attached to the private subnets. The load balancer distributes Remote Authentication Dial-In User Service (RADIUS) and Terminal Access Controller Access-Control System Plus (TACACS+) requests among Cisco ISE nodes.
 - Amazon Route 53 for Domain Name Service (DNS) resolution between Cisco ISE instances.
 - AWS Lambda for health checks, deployment, and failover of Cisco ISE instances.
 - AWS Step Functions for state machines to deploy and manage failover of Cisco ISE

The diagram on the right illustrates the architecture across two Availability Zones (Zone 1 and Zone 2). Each zone contains a VPC with a public subnet and a private subnet. The public subnets have NAT gateways. The private subnets have Cisco ISE primary and secondary nodes. A Network Load Balancer is attached to the private subnets. The diagram also shows various AWS services used in the solution, including Route 53, AWS Lambda, Step Functions, Systems Manager, CloudWatch, EventBridge, and Amazon SNS.

Buttons at the bottom of the diagram include 'Enlarge image', 'View deployment guide for details', and 'View the source code for this Partner Solution'.

EC2 and CloudFormation: <https://aws.amazon.com/marketplace/pp/prodview-uvsybra7r3iug>

Partner Solution: <https://aws.amazon.com/solutions/partners/cisco-ise-on-aws/>

Deployment Guide: https://www.cisco.com/c/en/us/td/docs/security/ise/3-1/install_guide/b_ise_InstallationGuide31/m_ISEaaS.html

Terraform Structure

```
#main.tf
resource "aws_instance" "CiscoLive-ISE" {
    ami                = var.ISE-ami
    instance_type      = var.ise_instance_type
    subnet_id          = var.subnet_id
    vpc_security_group_ids = [aws_security_group.CiscoLive-securityGroup.id]
    key_name            = var.key_name
    tags = {
        Name = "MRE_ISE"
    }

    credit_specification {
        cpu_credits = "unlimited"
    }

    user_data = file("ise.txt")
}
```

Terraform AWS USER DATA

```
#ISE bootstrap config. Passed via UserData - ISE.txt
hostname=mreISECiscoLive
primarynameserver=8.8.8.8
dnsdomain=example.com
ntpserver=time.nist.gov
timezone=CST6CDT
password=<password>
ersapi=yes
openapi=yes
pxGrid=no
pxgrid_cloud=no
```

External Connection



Connecting to On Prem Networks

VPN can be deployed using:

AWS Site-to-Site VPN

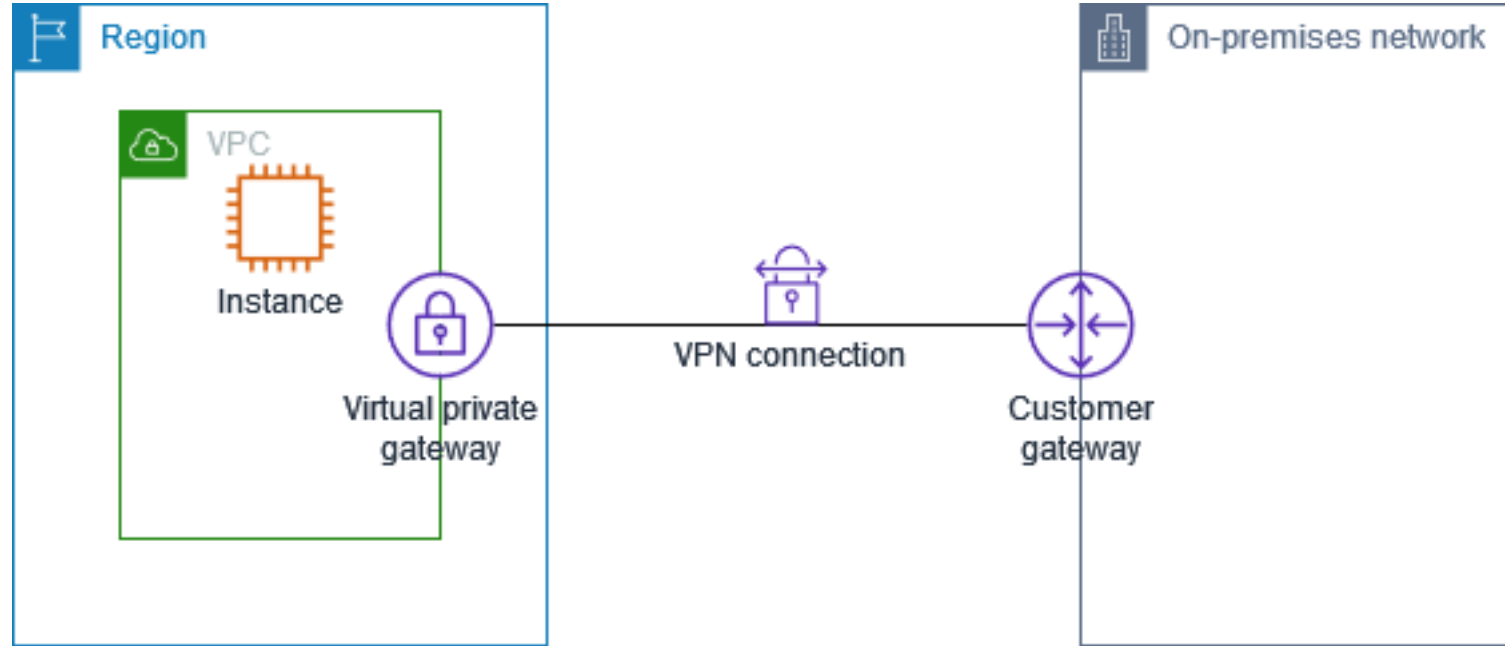
- Builds site-to-site VPN tunnel to on-prem router
- Either manual configuration or automation via 3rd party tools (ex. Ansible, etc.)
- Vendor agnostic

Cisco SD-WAN Solutions (Recommended)



- Use either:
 - C8000v – Cisco SD-WAN
 - Meraki vMX – Meraki AutoVPN
- Automatically configure VPN tunnels from AWS to On-Prem networks
- Push corporate security / access policies to the cloud

AWS Site-to-Site VPN



 Services

VPC dashboard 

EC2 Global View  New

Filter by VPC:

Select a VPC 

▶ Virtual private cloud

▶ Security

▶ DNS firewall

▶ Network Firewall

▼ Virtual private network (VPN)

Customer gateways

Virtual private gateways

Site-to-Site VPN connections

Client VPN endpoints

AWS Site-to-Site VPN – Customer Gateway

A customer gateway is a resource that you create in AWS that represents the customer gateway device in your on-premises network.

When you create a customer gateway, you provide information about your device to AWS.

VPC > Customer gateways > Create customer gateway

Create customer gateway [Info](#)

A customer gateway is a resource that you create in AWS that represents the customer gateway device in your on-premises network.

Details

Name tag - *optional*

Creates a tag with a key of 'Name' and a value that you specify.

Value must be 256 characters or less in length.

BGP ASN [Info](#)

The ASN of your customer gateway device.

Value must be in 1 - 2147483647 range.

IP address [Info](#)

Specify the IP address for your customer gateway device's external interface.

Certificate ARN

The ARN of a private certificate provisioned in AWS Certificate Manager (ACM).

Device - *optional*

Enter a name for the customer gateway device.

AWS Site-to-Site VPN – Virtual private gateways

A virtual private gateway is the **VPN concentrator** on the **Amazon side of the Site-to-Site VPN connection**. You create a virtual private gateway and attach it to a virtual private cloud (VPC) with resources that must access the Site-to-Site VPN connection.

VPC > Virtual private gateways > Create virtual private gateway

Create virtual private gateway [Info](#)

A virtual private gateway is the VPN concentrator on the Amazon side of the site-to-site VPN connection.

Details

Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.

Value must be 256 characters or less in length.

Autonomous System Number (ASN)

☒ Amazon default ASN

☐ Custom ASN

Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs. Name tag helps you track your resources more easily. We recommend adding Name tag.

Key	Value - optional	
<input type="text" value="Name"/>	<input type="text" value="MRE-VPC-GW"/>	<input type="button" value="Remove"/>

You can add 49 more tags.

AWS Site-to-Site VPN – Virtual private gateways

A virtual private gateway is the **VPN concentrator** on the **Amazon side of the Site-to-Site VPN connection**. You create a virtual private gateway and attach it to a virtual private cloud (VPC) with resources that must access the Site-to-Site VPN connection.

VPC > Virtual private gateways > vgw-0f5e3acd32041b2ab

vgw-0f5e3acd32041b2ab / MRE-VPC-GW

Details

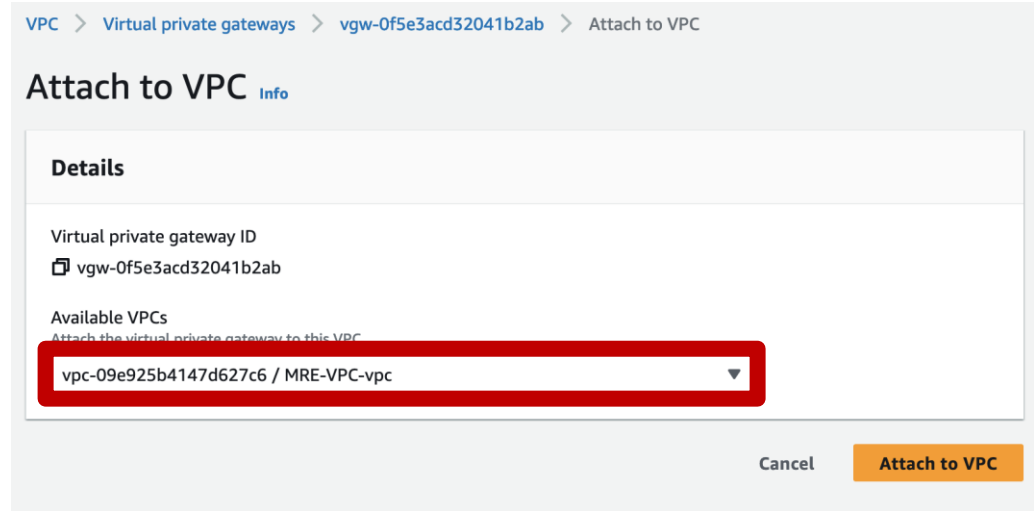
Virtual private gateway ID vgw-0f5e3acd32041b2ab	State Detached	Type ipsec.1	VPC -
Amazon ASN 64512			

Actions

- Attach to VPC
- Manage tags
- Delete virtual private gateway

AWS Site-to-Site VPN – Virtual private gateways

A virtual private gateway is the **VPN concentrator** on the **Amazon side of the Site-to-Site VPN connection**. You create a virtual private gateway and attach it to a virtual private cloud (VPC) with resources that must access the Site-to-Site VPN connection.



The screenshot shows the AWS Management Console interface for attaching a virtual private gateway to a VPC. The breadcrumb navigation at the top reads: VPC > Virtual private gateways > vgw-0f5e3acd32041b2ab > Attach to VPC. The main heading is 'Attach to VPC' with an 'Info' link. Below this is a 'Details' section containing the following information:

- Virtual private gateway ID: vgw-0f5e3acd32041b2ab
- Available VPCs: Attach the virtual private gateway to this VPC. A dropdown menu is shown with the selected option: vpc-09e925b4147d627c6 / MRE-VPC-vpc. This dropdown is highlighted with a red rectangular border.

At the bottom right of the dialog are two buttons: 'Cancel' and 'Attach to VPC'.

AWS Site-to-Site VPN – Virtual private gateways

A virtual private gateway is the **VPN concentrator** on the **Amazon side of the Site-to-Site VPN connection**. You create a virtual private gateway and attach it to a virtual private cloud (VPC) with resources that must access the Site-to-Site VPN connection.

vgw-0f5e3acd32041b2ab / MRE-VPC-GW

Actions ▼

Details			
Virtual private gateway ID vgw-0f5e3acd32041b2ab	State Attached	Type ipsec.1	VPC vpc-09e925b4147d627c6 MRE-VPC-vpc
Amazon ASN 64512			

AWS Site-to-Site VPN – VPN Connection

VPC > VPN connections > Create VPN connection

Create VPN connection [Info](#)

Select the resources and additional configuration options that you want to use for the site-to-site VPN connection.

Details

Name tag - optional
Creates a tag with a key of 'Name' and a value that you specify.

Value must be 256 characters or less in length.

Target gateway type [Info](#)

☒ Virtual private gateway
☐ Transit gateway
☐ Not associated

Virtual private gateway

Customer gateway [Info](#)

☒ Existing
☐ New

Customer gateway ID

Routing options [Info](#)

☐ Dynamic (requires BGP)
☒ Static

Static IP prefixes [Info](#)

10.0.43.0/24 X

AWS Site-to-Site VPN – VPN Connection

VPC > VPN connections > vpn-0e4aa63b015e41714

vpn-0e4aa63b015e41714 / MRE-VPN-Wifi

[Download configuration](#) Actions ▾

Details

VPN ID vpn-0e4aa63b015e41714	State Available	Virtual private gateway vgw-0f5e3acd32041b2ab	Customer gateway cgw-09655bd9c57b9feae
Transit gateway -	Customer gateway address [REDACTED]	Type ipsec.1	Category VPN
VPC -	Routing Static	Acceleration enabled False	Authentication Pre-shared key
Local IPv4 network CIDR 0.0.0.0/0	Remote IPv4 network CIDR 0.0.0.0/0	Local IPv6 network CIDR -	Remote IPv6 network CIDR -
Core network ARN -	Core network attachment ARN -	Gateway association state associated	Outside IP address type PublicIpv4

AWS Site-to-Site VPN – VPN Connection

Download configuration ✕

Choose the sample configuration you wish to download based on your customer gateway. Please note these are samples, and will need modification to use Advanced Algorithms, Certificates, and/or IPv6.

Vendor
The manufacturer of the customer gateway device (for example, Cisco Systems, Inc).

Cisco Meraki ▼

Platform
The class of the customer gateway device (for example, J-Series).

MX Series ▼

Software
The operating system running on the customer gateway device (for example, ScreenOS).

15.12+ (WebUI) ▼

IKE version
The IKE version you are using for your VPN connection.

ikev1 ▼

Cancel Download

AWS Site-to-Site VPN – Meraki Limitation

```
! IPSEC Tunnel #1 !
```

```
! AWS hosted VPN solution is a route-based solution, since Cisco Meraki  
only supports policy-based solution you will need to limit to a single  
SA. So please make sure to  
! select "yes" for just one subnet, if you have more than one subnet,  
consolidate them into a single subnet before proceeding with the VPN  
configuration.
```

Meraki AWS Site-to-Site VPN – Download Config – IKEv1

```
! IPSEC Tunnel #1 !
```

On the Cisco Meraki Dashboard, go to Security Appliance -->
Configure --> Site-to-Site VPN

Local networks: Select "yes" under "Use VPN" for the subnet you want
to include in the source encryption domain.

Under Organization-wide settings --> Non-Meraki VPN peers

Name: ipsec-vpn-0e4aa63b015e41714-0

Public IP: 52.62.61.139

Private subnets: <vpc_subnet>/<vpc_subnet_mask>

IPsec policies: Click "Default", select "AWS" under the Preset menu
and "Update"

Preshared secret: XXXXXXXXXXXXXXXXXXXXwbDyS4CQiJFd

Availability: All networks

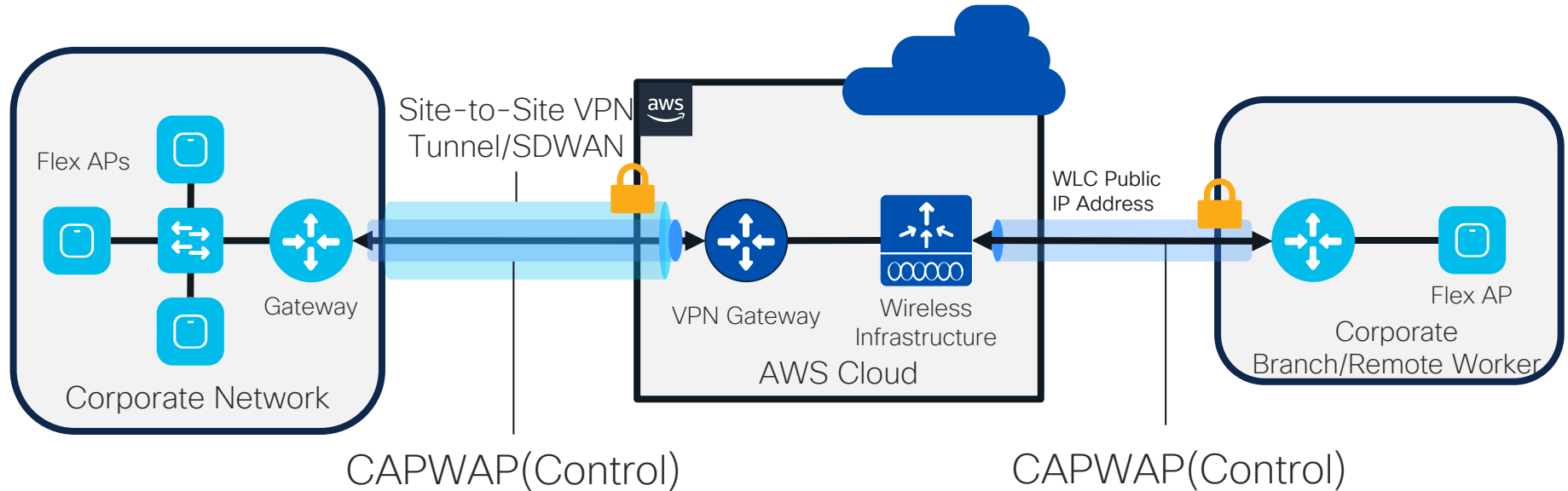
AWS Site-to-Site VPN – Confirming from AWS

Tunnel details									
Static routes									
Tags									
 This VPN connection is not using both tunnels. This mode of operation is not highly available and we strongly recommend you configure your second tunnel.									
Tunnel state									
Tunnel number	Outside IP address	Inside IPv4 CIDR	Inside IPv6 CIDR	Status	Last status change	Details	Certificate ARN		
Tunnel 1	52.62.61.139	169.254.95.188/30	–	 Down	September 5, 2023, 17:13:58 (UTC+10:00)	–	–		
Tunnel 2	52.63.131.164	169.254.175.220/30	–	 Up	September 5, 2023, 19:00:09 (UTC+10:00)	–	–		

Summary

You don't notice it, but you know
when its **not there**

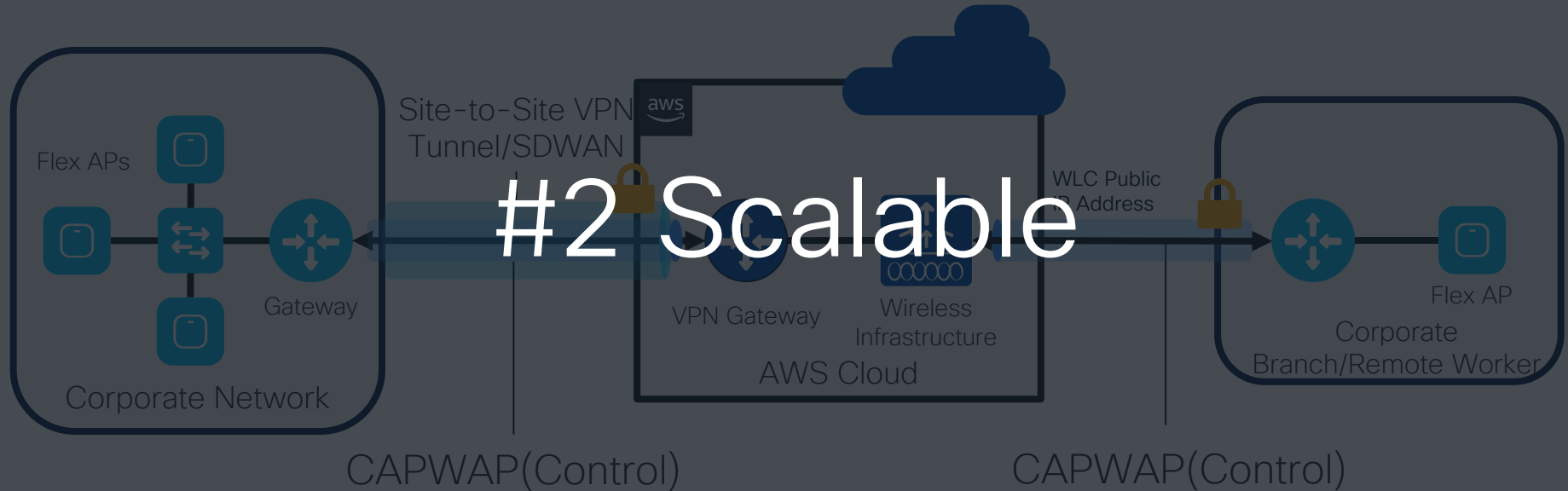
Cisco Wireless Infrastructure in AWS



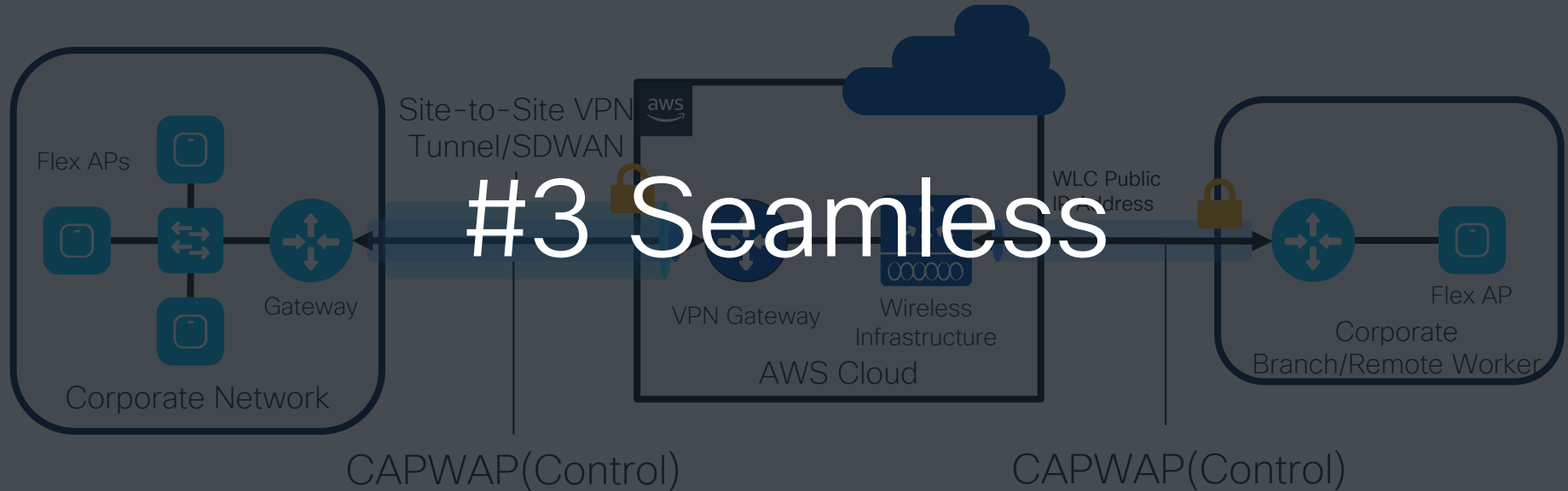
Cisco Wireless Infrastructure in AWS



Cisco Wireless Infrastructure in AWS



Cisco Wireless Infrastructure in AWS



Where do I get started:
Github for this session –
<https://github.com/markrebetzke>

Devnet/Developer –
<https://developer.cisco.com/>

MeetMe – 7/12/23 – Thursday 2:30–4PM – The
Hub
near World of Solutions

Session Surveys – Important!

We would love to know your feedback on this session!

- Complete a minimum of four session surveys and the overall event surveys to claim a Cisco Live T-Shirt
- If you want me back – please provide feedback 😊



Continue your education



- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

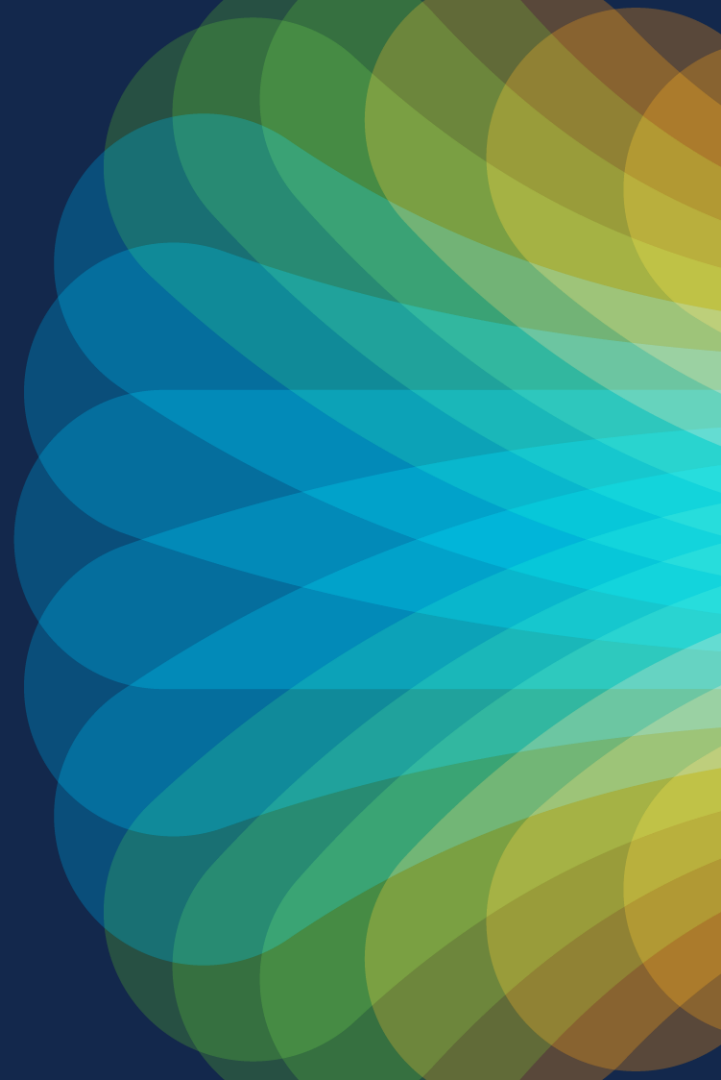


The bridge to possible

Thank you



#CiscoLive

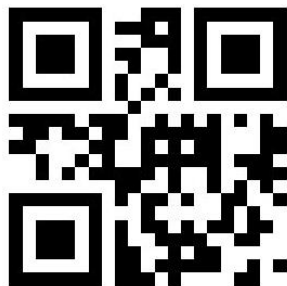
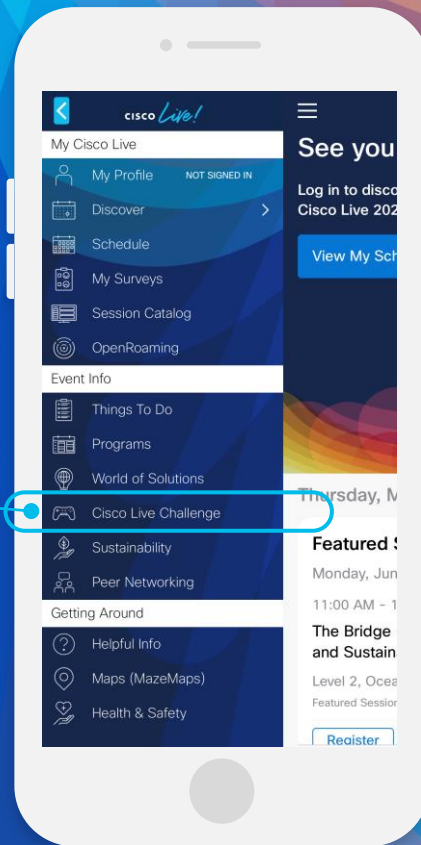


Cisco Live Challenge

Gamify your Cisco Live experience!
Get points for attending this session!

How:

- 1 Open the Cisco Events App.
- 2 Click on 'Cisco Live Challenge' in the side menu.
- 3 Click on View Your Badges at the top.
- 4 Click the + at the bottom of the screen and scan the QR code:



The background is a vibrant, abstract graphic. It features a large, stylized cloud shape on the left side, composed of overlapping, semi-transparent layers of orange, red, and yellow. To the right of the cloud, a bright, multi-colored sunburst or starburst pattern radiates outwards, transitioning through a spectrum of colors including yellow, green, blue, and purple. The overall effect is energetic and celebratory.

cisco *Live!*

Let's go

#CiscoLive