Zero to Cloud-Native with IBM Cloud

Part 3: IBM Cloud Account, Network and Security Configuration

Kevin Collins
Technical Sales Leader
IBM Cloud Enterprise Containers – Americas

Resource Groups

Introduction

One of the first things that you need to do after you have designed and architected your application, is planning how to setup your IBM Cloud Account. We will be leveraging a single IBM Cloud account and segregate resources with a resource group. A resource group is a way for you to organize your account resources in customizable groupings so that you can quickly assign users access to more than one resource at a time. Any account resource that is managed by using IBM Cloud Identity and Access Management (IAM) access control belongs to a resource group within your account.

Tutorial - Resource Groups

The first part of the tutorial is creating a new resource group for the zero to cloud native on ibm cloud tutorial resources. We will place all the resources we create during the tutorial into this resource group.

1) After logging into cloud.ibm.com click on Manage and then select access (IAM)
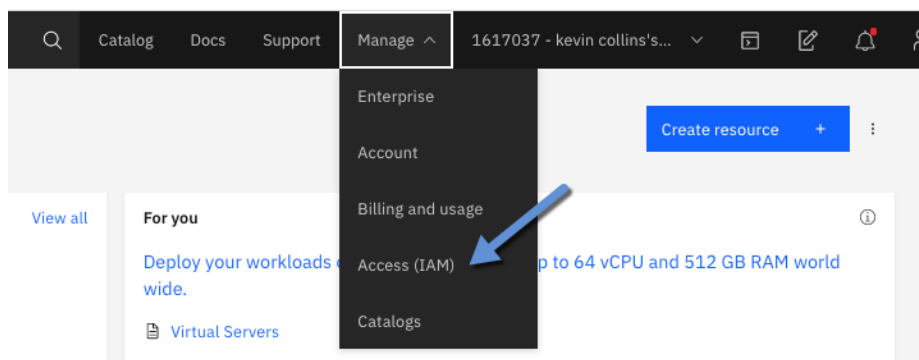
2) Click on Resource Groups and then Create. Enter a name, I will be use **zero-to-cloud-native**

Virtual Private Cloud - Network Design

Overview

After you have your resource group created, we can start deploying services for our application. The first consideration to think about is the network design. In IBM Cloud, we offer two network infrastructure designs, one being Virtual Private Cloud and the other being classic infrastructure. VPC on IBM Cloud is that latest and greatest cloud model that provides advanced security through network isolation, more granularity for network configuration and security and improved compute and storage provisioning. For both CloudPak Provisioner and the tutorial we will be using a virtual private cloud. Generally speaking, you will want one virtual private cloud for each environment that you plan to use. For CloudPak provisioner, I had one environment for dev and one for production. My very simplistic VPC design resulted in two VPCs, not surprisingly one for dev and one for production. The reason for this is to provide strong network isolation and control between the two environments.

Virtual Private Clouds

Region: Dallas ⌄						↻ ⚙ Create +

Status	Virtual Private Cloud	Resource Group	Subnets	Default ACL	Default Security Group	
● Available	getroks-cp-dev-vpc	getRoks	3	strongman-bacterium-utter-detail-probing-circula...	blighted-backtrack-squid-easiest-extended-aliens	⋮
● Available	getroks-cp-ocp-vpc	getRoks	3	preshow-blaming-mossy-eatable-moaning-neuron	veto-detoxify-unimpeded-parka-overgrown-privatiz...	⋮

There are several things you need to setup for your VPC.

- The first thing that you need to setup is a subnet for each availability zone that you plan to use.

- The second is an access control list for the VPC. You can use an access control list (ACL) to control all incoming and outgoing traffic in IBM Cloud Virtual Private Cloud. An ACL is a built-in, virtual firewall, similar to a security group. In contrast to security groups, ACL rules control traffic to and from the *subnets*, rather than to and from the *instances*.

- The third is a security group. Security groups give you a convenient way to apply rules that establish filtering to each network interface of a virtual server instance, based on IP address. When you create a security group, you configure it to create the network traffic patterns you want. By default, a security group denies all traffic. As rules are added to a security group, it defines the traffic that the security group permits.

Tutorial – Creating a Virtual Private Cloud

From the IBM Cloud Catalog, click on the hamburger menu, select VPC Infrastructure and then VPC.
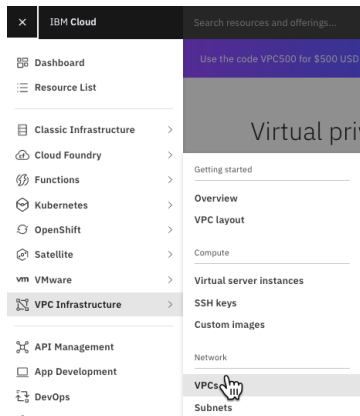
On the next screen click create, to create a new VPC.

On the next screen enter the following information:

Name – name of your VPC. Name this something meaningful in the case you plan to have multiple VPCs in your account. I will use zero-to-cloud-native-dal3 in this tutorial.

Resource Group – select the resource group name you create above. I am using zero-to-cloud-native in this tutorial.

Default security group – I recommend to keep Allow SSH and Allow Ping enabled for dev systems as it will allow us to ssh and ping into pods / services while debugging. For a production system, consider disabling these settings.

Classic Access – in this tutorial we will not be using resources in classic infrastructure so you can uncheck this option.

Default address prefixes – leave this option checked as we do want to create a default prefix for each zone.

Next, you will need to create a new subnet. In this tutorial, we will be deploying a multizone clusters across three zones which means we will need to create one subnet for each zone. In this example I will be using Dallas as my multizone region. When you create a VPC you only have an option of creating a single subnet during the VPC creation. We will start by creating a subnet in on the multizone availability zones and then go back and create the other two.

Subnet Name – when I create subnets, I include the vpc name and location in the subnet name so I can easily find the subnets if needed. I will start with dallas 3 and will name my first subnet zero-to-cloud-native-dal3

Location – pick one of the locations in the MZR that you are planning on using. I will start by using Dallas 3.

IP Range – being a rather simply tutorial, we won't need that many IP addresses for our cluster and application. I will select 64 addresses.

Public Gateway – make sure you select Attached. This will allow us to access the OpenShift Console. For a production level cluster, consider un-attaching the public gateway if you do not require Internet connectivity.

After entering all the information, click Create virtual private cloud on the right hand side of the screen.
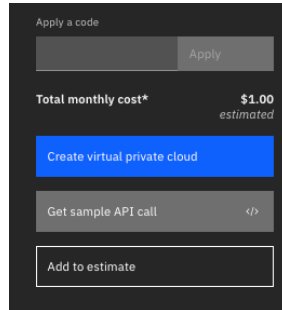
The next screen will take you to a list of virtual private clouds. Click on the virtual private cloud you just created. We will need to create additional subnets to support our multizone cluster and we will need to create access rules for our application in the security group section.

The first thing we will want to do is to create our subnets for the additional two availability zones for our multizone cluster.

Scrolling down to our subnets in this VPC section. Click on create to create another subnet.

Follow the same steps you did when you created your first subnet when you created your VPC. I will be using these settings for the second subnet I'm creating in Dallas 2.

Name: zero-to-cloud-native-dal2
Resource Group: zero-to-cloud-native
Location: Dallas 2
IP Range – Number of addresses: 64
Public Gateway: Attached
*leave all the other settings with the default values.

Once you have entered all the settings, click on Create Subnet to create this new subnet.

Repeat the same process of the third subnet. I will be using these settings for the third and final subnet in Dallas 1.

**Name:** zero-to-cloud-native-dal1

Resource Group: zero-to-cloud-native

Location: Dallas 1

**IP Range – Number of addresses:** 64

Public Gateway: Attached

*leave all the other settings with the default values.

Now that we have created our three subnets in each zone for or multizone region, we can now update the security group to support our application. On the VPC navigation tree on the left, click on Security Groups which will bring up a list of all the security groups in your account. On the right-hand side, click on the security group for the VPC you just created.
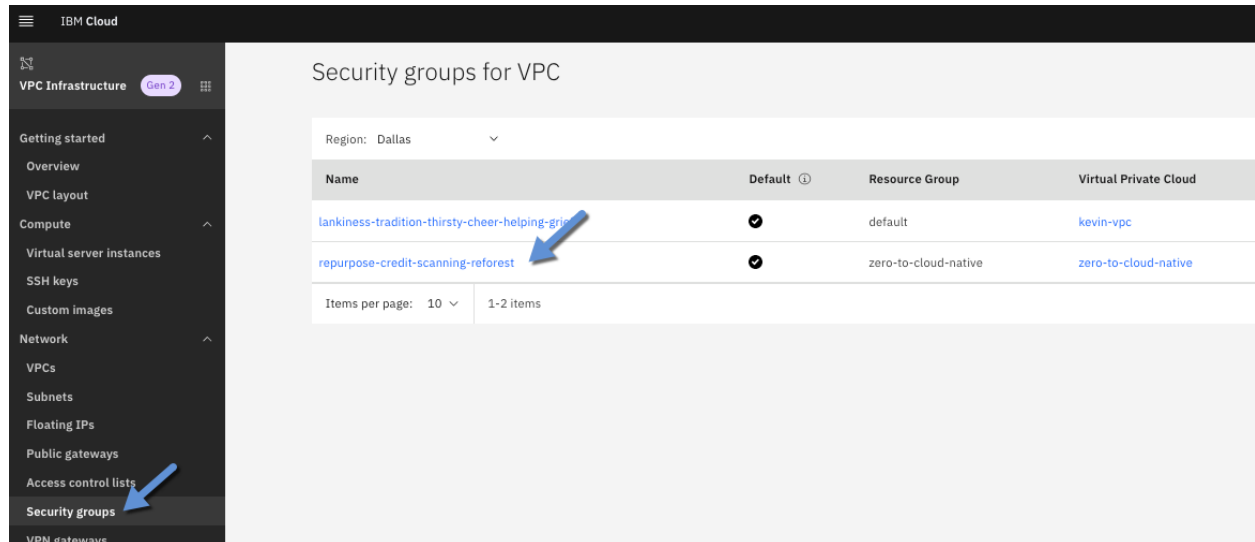
By default, when you create a VPC all outbound traffic will be allowed and all inbound traffic will be blocked. Since we did enable SSH and Ping those will also be allowed for inbound traffic. For our cloud native application, we will need to create three additional rules, one for the OpenShift console, second for web application and third for the API interface.

Click on create in the Inbound rules section to create a new rule.
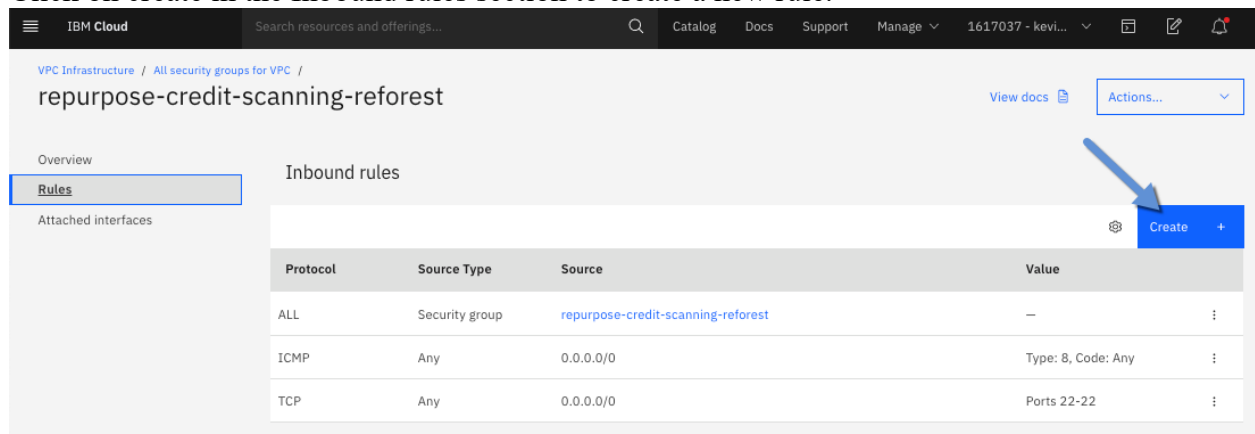
Starting with the ports for OpenShift, enter 30000 as the Port min and 32767 as the Port max and click save.

New Inbound Rule ✕

Protocol

TCP ⌄

Port   **Port min**   **Port max**

◯ Any   ⦿ Port range   30000   32767

Source type

⦿ Any   ◯ IP address   ◯ CIDR block   ◯ Security group

Create the same process to port 80 (web application) and port 8080 (API Interface)
After creating the additional rules, your Inbound rules should look like this.

Inbound rules

        ⚙   Create   +

| Protocol	Source Type	Source	Value	
ALL	Security group	repurpose-credit-scanning-reforest	—	⋮
ICMP	Any	0.0.0.0/0	Type: 8, Code: Any	⋮
TCP	Any	0.0.0.0/0	Ports 22-22	⋮
TCP	Any	0.0.0.0/0	Ports 30000-32767	⋮
TCP	Any	0.0.0.0/0	Ports 8080-8080	⋮
TCP	Any	0.0.0.0/0	Ports 80-80	⋮

Custom Domains

When developing your cloud-native application, you will most likely want a domain that is easily readable rather than the default domain you get when you create an OpenShift cluster like the following:

```
getroks-nonprod-ocp-73aebe0673131e634c608c4167edcc2aeb-0000.us-south.containers.appdomain.cloud
```

One of the services we have in IBM Cloud is called Cloud Internet Services which provides reliability, performance, and security for Internet facing applications, websites, and services using Cloudflare's 165+ Global Points of Presence (PoPs). It includes Domain Name Service (DNS), Global Load Balancer (GLB), Distributed Denial of Service (DDoS) protection, Web Application Firewall (WAF), Transport Layer Security (TLS), Rate Limiting, Smart Routing, and Caching. For the tutorial, we will mainly be using Cloud Internet Services for a Domain Name Service.

Custom Domain Tutorial

If you don't already have a domain that you would like to use, then the first thing we will need to do is create one. From the IBM Cloud catalog, search from Domain and then click on Domain Name Registration Service.
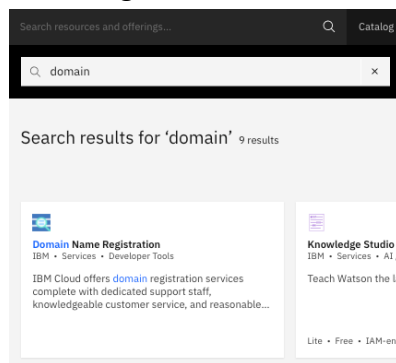
On the next screen, there are no options, on the bottom right corner, click create.
The following screen will allow you the option to create a new domain. Click on Register, enter the domain you want, I will be using zero-to-cloud-native.com. Click on Check Availability to make sure the domain you want is available and then click Continue.

Enter your registrant information and click Order Now.

Complete zero-to-cloud-native.com Registration

Registrant

First Name
[   ]
* Required field

Last Name
[   ]
* Required field

☐ Use Default Company Address

Company Name
[   ]
* Required field

Street Address
[   ]
* Required field

[   ]

City
[   ]

[Cancel] [Order Now]

After about 5 minutes, you will see the domain you request show up in the domain view. Expand the domain you created and not the custom name servers to use later.

Domains

Domains ▲	Registrar	Status	Contacts	Verified	Lock Domain	DNS	Expiration
˅ zero-to-cloud-native.com	SoftLayer	Active	View / Edit	Verifying	Locked ˅	Edit DNS	2021-08-25

Filter ˅ Transfer ˅ Renew ˅ Register ˅

All Domains ▼   Items per page: 25 ▼ | 1-1 of 1 items   1 of 1 pages   ‹ 1 ▼ › ↻ | ⚙

Custom Name Servers
Point this domain to custom name servers
ns1.softlayer.com

Registered Name Servers
Register Name Server(s) for others to point to this domain
There are no NS records currently registered with this domain.

Transfer Code
*)+gd7(?B?3b

The next step will be connecting the domain you just created with Cloud Internet Services. Please note, you will need to verify your domain before you can complete the process of registering the domain with IBM Cloud Internet Services. Look for an email from IBM with instructions on verifying your information.

Cloud Internet Services Tutorial

The next thing we will need to do is create an instance of Cloud Internet Services from the IBM Cloud catalog. To do so, click on the Catalog of IBM Cloud, search for Internet Services, and click the Internet Services tile. We will used Cloud Internet Service to manage the DNS and Internet security for the domain we just created.

IBM Cloud

Catalog

Catalog

IBM Cloud catalog

Featured

Services

Software

Consulting

Category

- [ ] Networking
- [ ] Storage
- [ ] Developer Tools
- [ ] Integration
- [ ] Security

Search resources and offerings...

Internet Services

Search results for 'Internet Services' 10 results

Internet Services
IBM • Services • Networking

Using Cloudflare, Cloud Internet Services (CIS) provides Domain Name Service (DNS), Global Load Balancer (GLB), DDoS protection, Web Application...

Free • IAM-enabled

HashiCorp Consul
Third party • Software

HashiCorp Consul is configuring services

Helm charts • OVA Im vCenter Server • Free

NGINX Ingress Controller
Third party • Software • Developer Tools

Hyper Protect Crypt
IBM • Services • Secu

This brings up a screen to select the pricing plan you want for Internet Services. Select the Standard plan as being a tutorial and no production system we don't need the all the advanced security options provided by the other plans. For a production system, we strongly recommend that you select one of the Enterprise plans.

At the bottom of the screen, enter a name for the service and the resource group you created earlier to place the service in. I will be using Internet Services-zero-to-cloud-native as the name of my service. Click create after entering selecting the plan and entering the name for your service.

This will create your service and will bring you to the overview page of your IBM Cloud Internet Services instance you just created. To get started ... click Let's get started.

This will bring up a new page to setup your domain. Enter the domain name you created earlier and click next.

Setup your domain

×

● Connect   ○ DNS records   ○ Domain
   domain   management

Connect your domain

Enter your domain name in the following input field to
connect it to CIS. This does not affect your domain's web
traffic until you change your name servers in the next step.
To complete the connection, you must have admin access to
the DNS provider or registrar for the domain. You can always
change the connected domain at a later time.

Domain name

zero-to-cloud-native.com

You can skip the setup your DNS records, just click next.

Setup your domain

×

⊘ Connect   ● DNS records   ○ Domain
   domain   management

Setup your DNS records (optional)

Upload a DNS file in the BIND format. Conflicting records will
not be overwritten. Depending on the number of records in
the file, the upload may take a few minutes.

Import records   ⬆

Next, on the delegate domain management screen, note the new NS records. Before you click next, you will need to update the domain you created with these records. Note, you can only update the DNS records name server records once per hour. If you get an error stating so, wait an hour from when you created the domain. Open a new browser or tab and navigate to the IBM Cloud Domain Service like you did in the previous step. Under the domain you created select Unlock from the Lock Domain drop down.

Next click on Add/ Edit NS and enter the NS specified on the Cloud Internet Services screen. Enter the new name servers and click Associate.

Now, go back to Cloud Internet Services and click the final Next button.

Setup your domain

⊘ Connect domain   ⊘ DNS records   ● Domain management

Delegate domain management

Delegate your domain's management system to CIS by changing your name server (NS) records. After delegation, traffic redirects to CIS over 24 hours and we apply all of our security and performance benefits.

Start by going to your domain's current DNS provider or registrar (usually where you purchased the domain) and find a section titled "custom DNS" or "DNS server settings." The section's title varies from provider to provider. When there, replace the existing NS records with the new NS records and save changes.

The domain will be deleted automatically if not activated within 60 days.

Learn more about delegating domain management.

New NS records

ns060.name.cloud.ibm.com

ns135.name.cloud.ibm.com

Old NS records

ns1.softlayer.com

ns2.softlayer.com

⚠ Name server records are not updated. ✕
August 25, 2020 11:58 AM

After you click the final next button, you will be taken to the Overview page of IBM Cloud Internet Services. Your domain names are now being managed by the service.   We will come back to IBM Cloud Internet Services later on to create a DNS entry for our zero-to-cloud-native application.

Tutorial – IBM Cloud Certificate Manager

The next thing we will need to do is create a certificate for the domain we just created. To do so, click on Catalog, search for Certificate Manager, and then click the Certificate Manager tile.
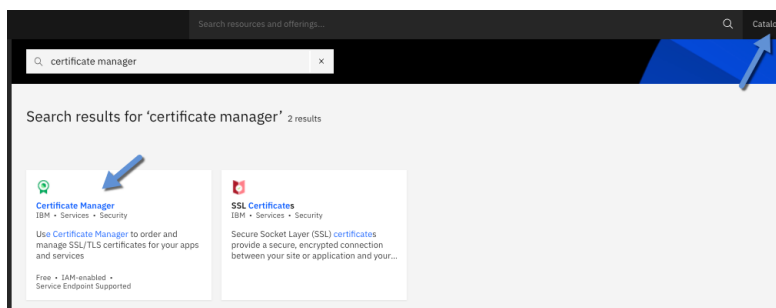
On the next screen, enter details for a new instance of IBM Cloud Certificate Manager.
Select a Region – best practice is to keep all resources in the same region. I will use Dallas.

Service Name – so we can distinguish different instances give the service a meaningful name. I will use Certificate Manager – zero-to-cloud-native.

Endpoints – you can optionally block access to the public network and use private only endpoints. Being a simply tutorial, I will keep the default Public and Private.

After entering these values, click Create.   This brings you to a screen showing your certificates. As we have just created the instance of Certificate Manager you will not see any certificates and will need to create one. To do so, click the Order button.
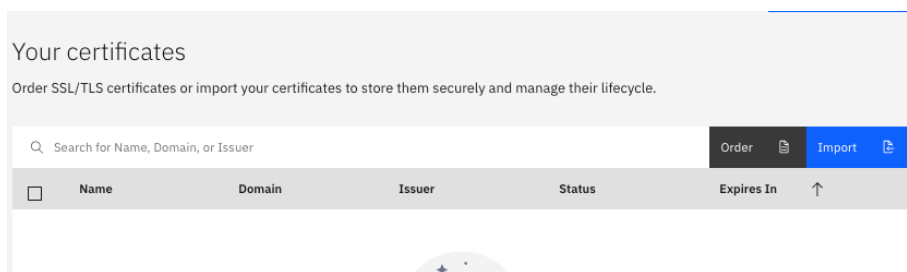
Because we are using Cloud Internet Services as our DNS provider, click Continue under IBM Cloud Internet Services (CIS) tile.
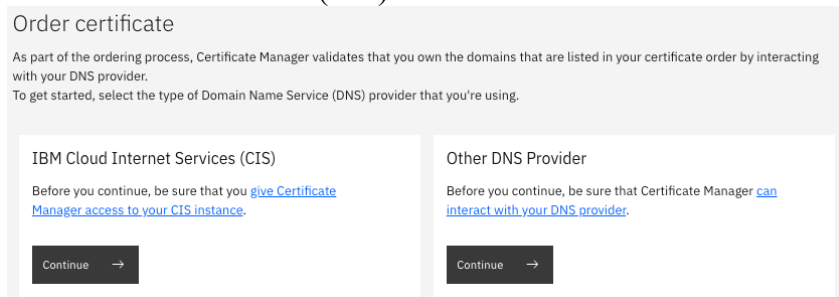
On the next screen, you will see a warning message saying there are no IBM Cloud Internet Services instances that Certificate Manager has permission to access.
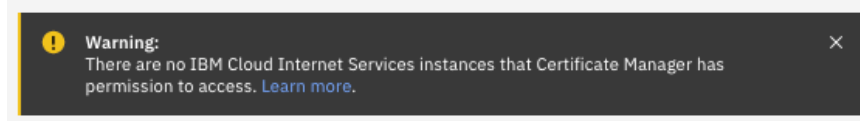
To fix this, we will need to authorize certificate manager to interact with Cloud Internet Services.

1. Navigate to **IBM Cloud > Manage > Access (IAM) > Authorizations**.
2. Click **Create** and assign a source and target service. The source service is granted access to the target service based on the roles that you set in the next step.
- Source: Certificate Manager
- Target: Internet Services
3. Specify a service instance for both the source and the target.

4. Assign the **Reader** role to allow Certificate Manager to view the CIS instance and its domains. Then, click **Authorize**.

First, select the source service to see which services it can be authorized to access. Then, select from the available target services, and assign all of the roles that are needed to give the source service access.

Source service ⓘ

| Certificate Manager | × ∨ | in | Account | × ∨ |

Source service instance

| Certificate Manager - zero-to-cloud-nativ | × ∨ |

Target service ⓘ

| Internet Services | × ∨ | in | Account | × ∨ |

Service instance

| string equals | ∨ | Internet Services-zero-to-cloud-native (3 | × ∨ | ⓘ |

Domain Id

| string equals | ∨ | | | ⓘ |

CIS functional scope

| string equals | ∨ | CIS functional scope | ∨ | ⓘ |

sub-scope

| string equals | ∨ | sub-scope | ∨ | ⓘ |

Service access ⓘ

☐ Reader   (6)   As a reader, you can perform read-only actions within a service such as viewing service-specific resources.

☐ Writer   (10)   As a writer, you have permissions beyond the reader role, including creating and editing service-specific resources.

☑ Manager   (14)   As a manager, you have permissions beyond the writer role to complete privileged actions as defined by the service. In addition, you can create and edit service-specific resources.

Cancel   Authorize

Now, when you go back to the Certificate Manager order certificate screen you will not see that warning message.

Next on the Order Certificate screen, enter the details for your certificate.
Name: zero-to-cloud-native

Automatic certificate renewal – recommendation is to turn this to yes. The certificate manager service will automatically renew your certificate and notify you. You can also automate deployments through a CI/CD process to deploy the renews certificates.

Order certificate

Be sure that this instance of Certificate Manager has the Manager IAM service access role for the domains in your IBM Cloud Internet Services instance. Learn more.

Certificate details   Domains

Name

zero-to-cloud-native

Description (optional)
Never add passwords or personal information (PII) to the description.

Certificate authority

Let's Encrypt   ⌄

Signature algorithm
The cryptographic algorithm used by the CA to sign this certificate.

sha256WithRSAEncryption   ⌄

Key algorithm
The encryption algorithm and key size used to generate the private and public keys for this certificate.

rsaEncryption 2048 bit   ⌄

Automatic certificate renewal ⓘ   ●✓

⚠ When a certificate is renewed you are notified through your notification channels that it is ready for you to deploy. Learn how to automate deployments.

Next, click on the Domains tab and select the IBM Cloud Internet Services (CIS) instance. Select both Add Domain and Add Wildcard, and click Order.

This will order your new certificates that will take a couple of minutes. We won't be using the certificates right away so you don't have to wait for the order to complete.

Tutorial – Key Protect

The last tutorial in this section is to create an instance of Key Protect that we will be using through the tutorial. IBM Key Protect for IBM Cloud helps you provision encrypted keys for apps across IBM Cloud services. As you manage the lifecycle of your keys, you can benefit from knowing that your keys are secured by FIPS 140-2 Level 3 certified cloud-based hardware security modules (HSMs) that protect against the theft of information.

For highly regulated workloads, IBM is the only cloud provider to offer FIPS 140-2 Level 4 encryption where you keep your own key. IBM Cloud Hyper Protect Crypto Services integrates with Key Protect to enable Keep Your Own Keys (KYOK) for IBM Cloud, so your organization has more control and authority over its data. Check out the Hyper Protect Crypto Services offering details page.

For this tutorial we will be using BYOK with IBM Cloud Key Protect for all of the IBM Cloud services including our OpenShift cluster and workloads.

The first thing we need to do is to create an instance of Key Protect. To do so, go to the IBM Cloud, search for Key Protect, and click the Key Protect tile.
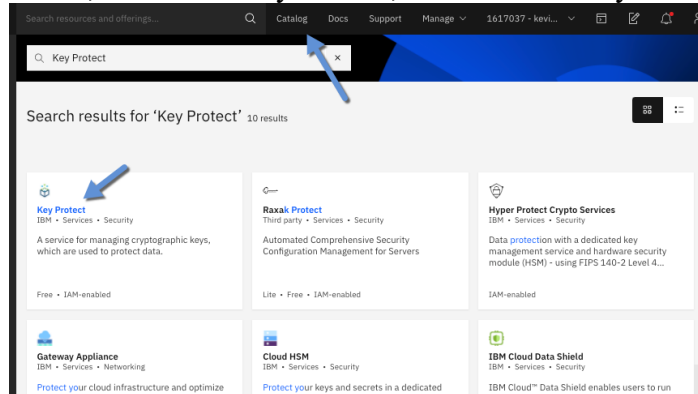
Just like you did with the other services so far, there are a couple of settings to enter to create the service.

Select a Region – best practice is to keep all resources in the same region. I will use Dallas.

Service Name – so we can distinguish different instances give the service a meaningful name. I will use Key Protect–zero-to-cloud-native.

Resource Group – to keep everything nicely organized, I will choose the zero-to-cloud-native resource group.

Endpoints – you can optionally block access to the public network and use private only endpoints. Being a simply tutorial, I will keep the default Public and Private.

After entering all your settings, click Create.

The next thing we need to is add an encryption key to Key Protect. Navigate to your Key Protect instance from your IBM Cloud resource list and click Add Key.

On the next screen you have the option of either having the service create a key or you can import your own key. For this tutorial, we will just have Key Protect create our own Root key, but if you can easily import your own key if you have one.

Give your key a meaningful name such as zero-to-cloud-native-root-key and click Create Key.

Congratulations!!
If you have made it this far, you have done the longest step of the zero-to-cloud-native tutorial series. We have setup a solid and secure enterprise environment to start deploying our application. To recap, we have just done the following:

- Created a secure Virtual Private Cloud opening only the ports our application will use.
- Created a custom domain with IBM Cloud Domain Services.
- Created an instance of Cloud Internet Services and configured it as the DNS provider of our domain.
- Created a wildcard certificate for our newly created domain using IBM Cloud Certificate Manager with automated renewal.
- Created an instance of IBM Cloud Key Protect where we will store our encryption keys.

And we did all of this on IBM Cloud!