

# Déployer et exécuter des applications IOx sur IE3400

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Configuration](#)

[Réseau](#)

[Activer IOx](#)

[Pour commencer](#)

[Via Local Manager](#)

[Se connecter à Local Manager](#)

[Déployer](#)

[Activer](#)

[Démarrer](#)

[Utiliser IOxclient](#)

[Préparer ioxclient](#)

[Déployer](#)

[Activer](#)

[Démarrer](#)

[Utiliser l'interface de ligne de commande IOS-XE](#)

[Déployer](#)

[Activer](#)

[Démarrer](#)

## Introduction

Ce document décrit comment déployer, activer et démarrer une application IOx sur IE3400.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Compréhension de base de Cisco IOx et IOS
- IE3400 8P2S ou 8T2S exécutant IOS XE version >17.2.1

## Configuration

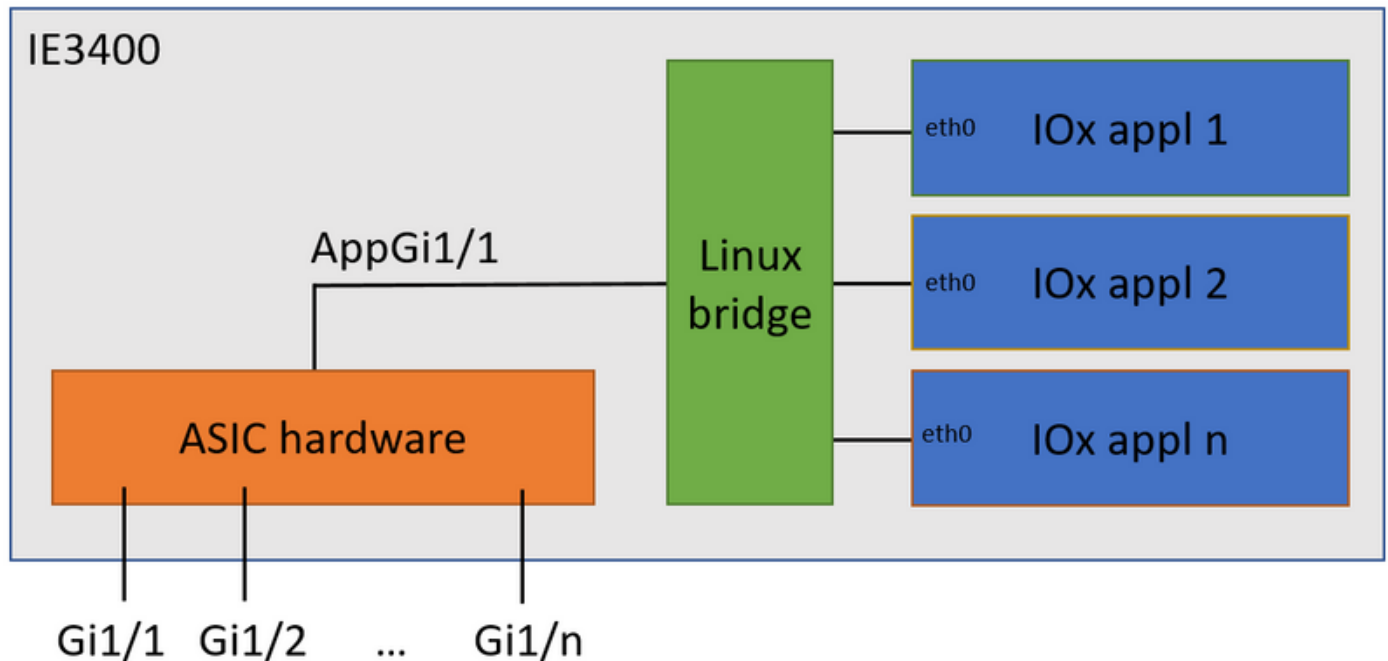
Depuis la version 17.2.1 de l'IOS, les commutateurs de la gamme robuste IE3400 prennent en

charge l'hébergement d'applications IOx. Cela vous permet d'exécuter vos propres codes/applications/conteneurs personnalisés sur le périphérique de périphérie. Comme pour la plupart des plates-formes IoT qui prennent en charge l'hébergement d'applications, il convient de garder à l'esprit certaines caractéristiques qui sont expliquées plus en détail dans ce document.

## Réseau

L'IE3400 dispose d'une interface virtuelle supplémentaire pour la connectivité aux applications IOx, appelée AppGigabitEthernet1/1

L'architecture simplifiée de cette interface est illustrée dans cette image.



Comme vous pouvez le voir, l'interface appelée AppGigabitEthernet1/1 fournit la connectivité aux applications. Il existe différentes options disponibles car AppGigabitEthernet1/1 peut être configuré comme interface physique régulière en mode d'accès ou de liaison. Un autre nom courant pour ce port est le port KR.

Actuellement, une interface d'application IOx doit avoir un ID de VLAN configuré même si l'interface AppGigabitEthernet1/1 est configurée en mode d'accès, car le trafic reçu par virteth2/L2br est toujours balisé.

Dans ce document, AppGigabitEthernet1/1 en mode d'accès dans VLAN 1 est configuré :

```
ie3400#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ie3400(config)#interface AppGigabitEthernet1/1
ie3400(config-if)#switchport mode access
ie3400(config-if)#switchport access vlan 1
ie3400(config-if)#end
```

Afin d'atteindre le commutateur et les applications IOx, l'adresse IP SVI pour VLAN 1 doit être configurée :

```
ie3400#conf t
```

```
Enter configuration commands, one per line. End with CNTL/Z.
ie3400(config)#interface Vlan1
ie3400(config-if)#ip address 192.168.0.30 255.255.255.0
ie3400(config-if)#end
```

## Activer IOx

Une fois le réseau configuré, activez IOx sur le périphérique.

Sur IE3400, une carte SD est nécessaire pour stocker les applications et les données IOx. Avant d'activer IOx, assurez-vous que la carte SD est formatée avec le système de fichiers ext4 :

```
ie3400#format sdflash: ext4
Format operation may take a while. Continue? [confirm]
Format operation will destroy all data in "sdflash:". Continue? [confirm]
format completed with no errors

Format of sdflash: complete
```

Ensuite, activez IOx avec cette commande :

```
ie3400#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ie3400(config)#iox
Warning: Do not remove SD flash card when IOx is enabled or errors on SD device could occur.

*Feb 21 12:49:18.310: %UICFGEXP-6-SERVER_NOTIFIED_START: R0/0: psd: Server iox has been notified
to start
*Feb 21 12:49:48.165: %IM-6-IOX_ENABLEMENT: R0/0: ioxman: IOX is ready.
```

Afin d'accéder à distance à la fonctionnalité IOx, en utilisant, par exemple, Local Manager ou ioxclient, assurez-vous que le serveur Web est activé et qu'un utilisateur est configuré pour l'accès :

```
ie3400#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ie3400(config)#ip http secure-server
ie3400(config)#username admin privilege 15 password 0 secret
```

## Pour commencer

Il existe plusieurs méthodes pour déployer des applications IOx sur le IE3400. Ce document décrit les éléments suivants :

- Utilisation du gestionnaire local - Interface utilisateur graphique
- Utilisation d'ioxclient - CLI distante
- Utilisation de IOS-XE - CLI sur périphérique

L'une des méthodes est suffisante pour effectuer toutes les tâches, ils sont simplement documentés ici pour l'exhaustivité et pour atteindre un large public.

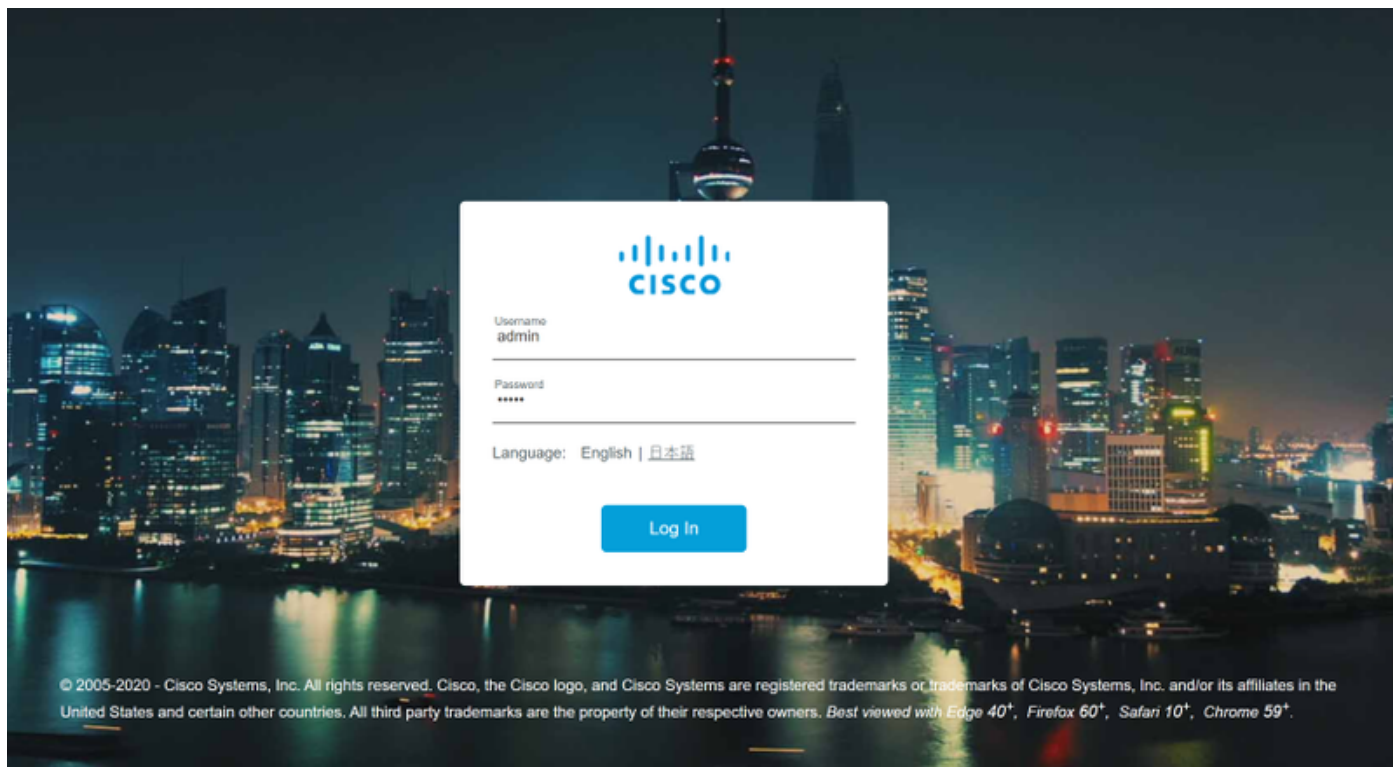
### Via Local Manager

Local Manager est une interface graphique à utiliser pour gérer et déployer des applications IOx sur des plates-formes compatibles IOx. Local Manager s'exécute sur le périphérique IOx lui-

même, dans ce cas, le IE3400.

## Se connecter à Local Manager

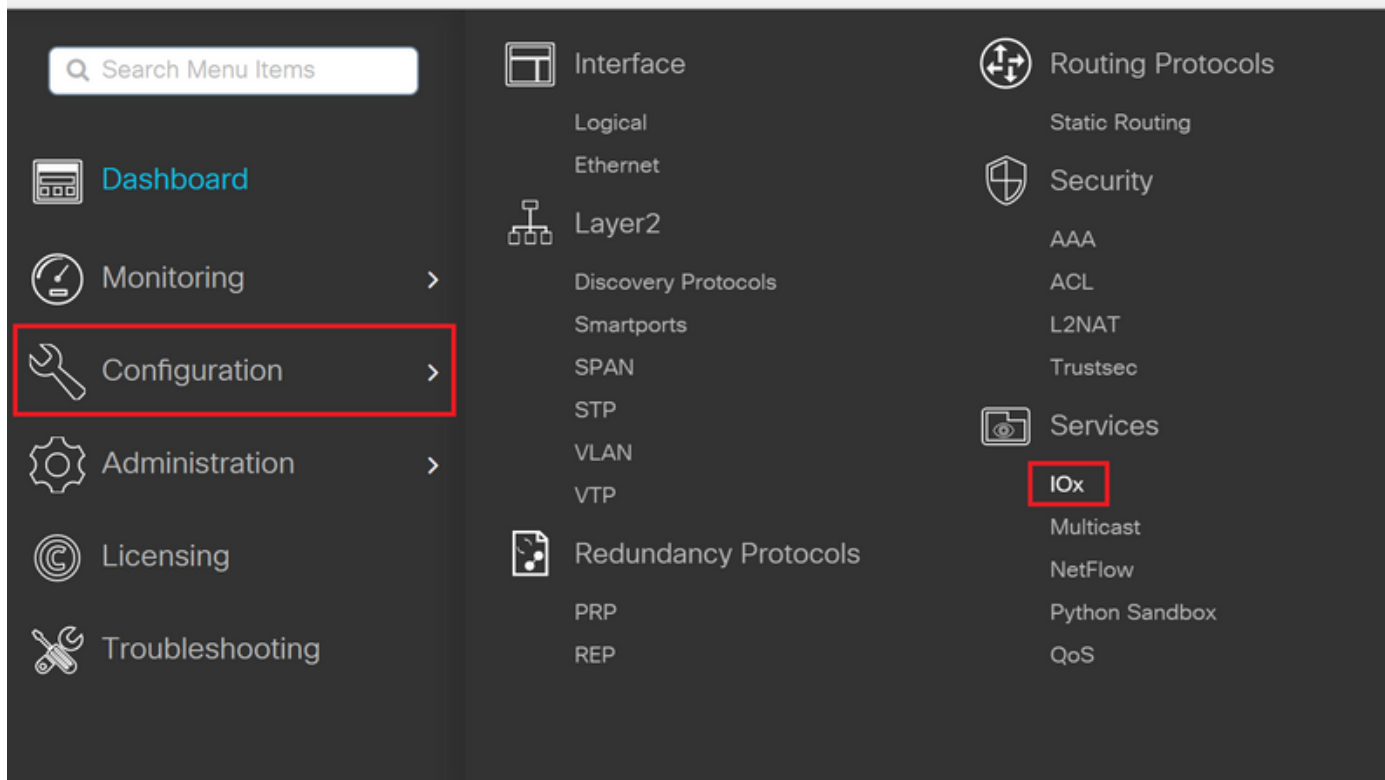
Si vous avez suivi la procédure d'activation du serveur Web et d'ajout d'un utilisateur, vous devriez pouvoir accéder à l'interface Web d'IE3400 à l'aide de l'adresse IP SVI à l'aide de `https://<svi ip>/` (dans cet article : <https://192.168.0.30/>):



Vous pouvez vous connecter à l'aide de l'utilisateur créé comme expliqué ci-dessus, puis naviguez jusqu'à **Configuration > Services > IOx**, comme illustré dans l'image.

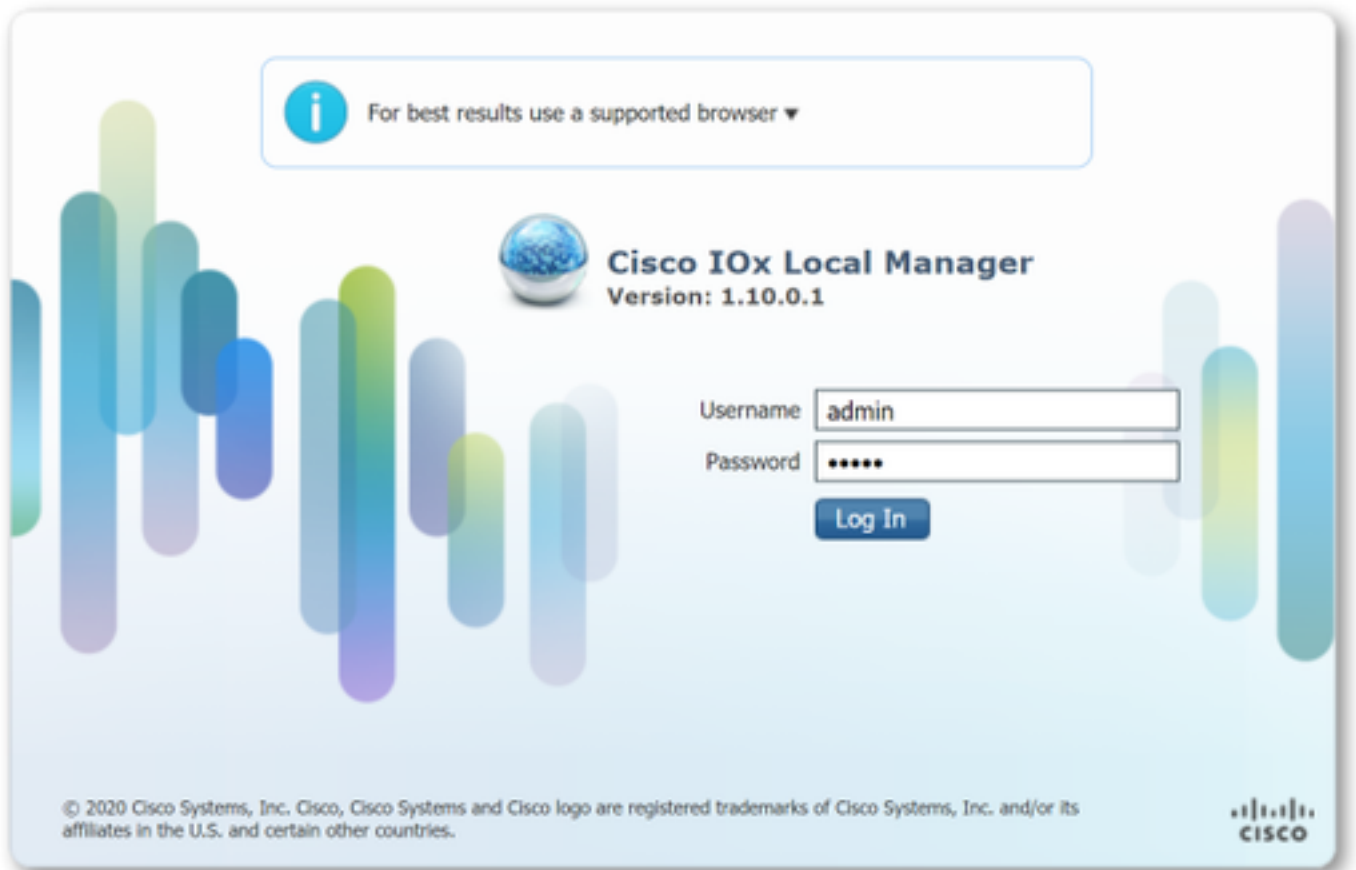


Cisco IE-3400-8T2S  
17.2.1



Si vous préférez accéder directement à Local Manager, vous pouvez utiliser l'URL suivante : <https://<svi ip>/iox/login> (dans cet article <https://192.168.0.30/iox/login>).

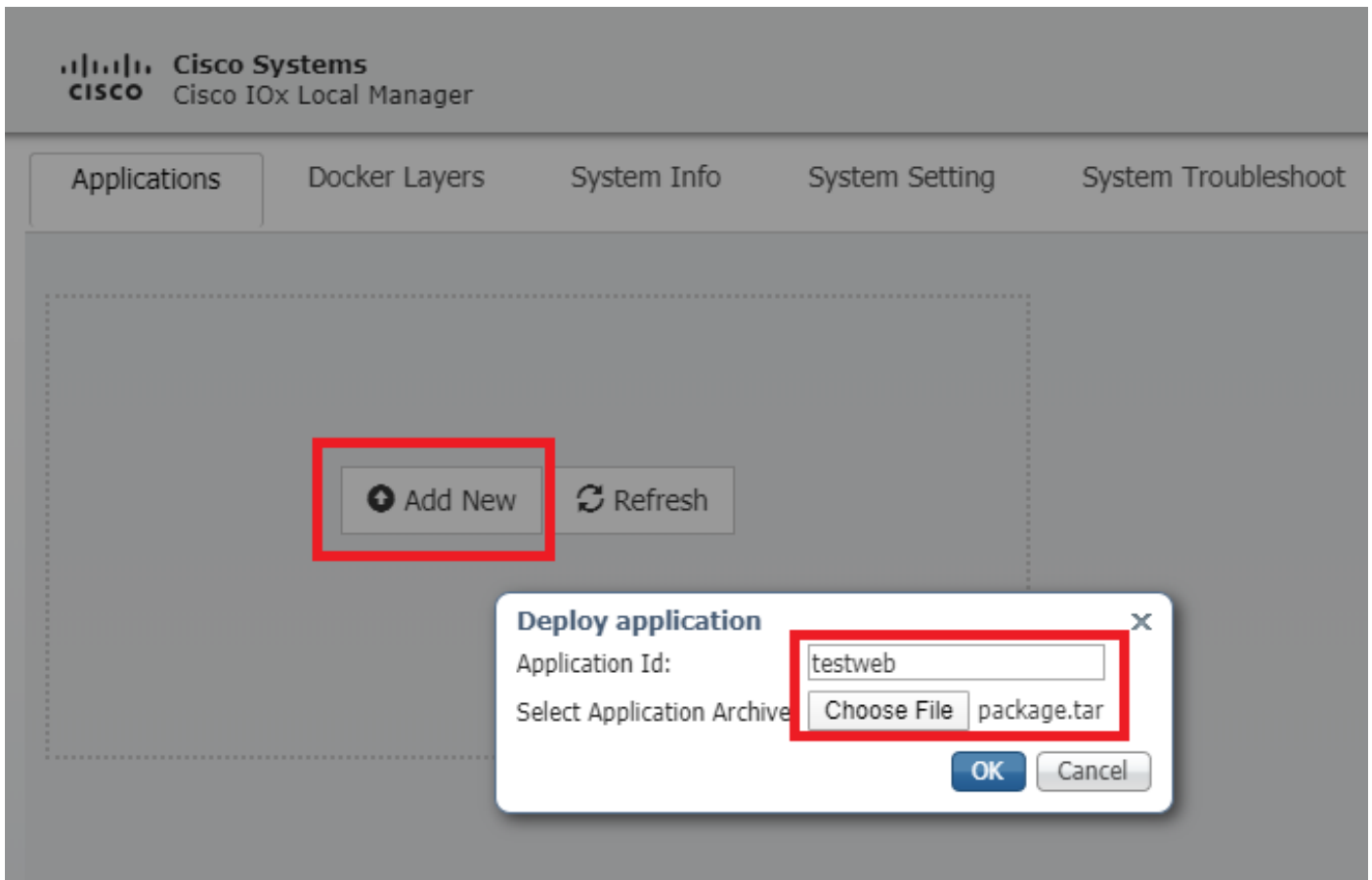
Dans l'écran de connexion de Local Manager, utilisez à nouveau la définition de l'utilisateur priv 15 ci-dessus.



## Déployer

Une fois connecté, démarrez le déploiement de l'application.

Cliquez sur **Add New**, choisissez un nom pour l'application et accédez au package de l'application sur votre ordinateur, comme l'illustre l'image.



Après avoir cliqué sur **OK**, l'application est téléchargée et installée sur le IE3400. Si tout va bien, vous devriez voir cette image sur votre écran, avec votre application IOx dans un état déployé.

## Activer

Une fois l'application déployée, l'étape suivante consiste à l'activer. Dans cette phase, les ressources utilisées par l'application sont spécifiées.

Cliquez sur le bouton **Activer** de l'application IOx déployée, comme illustré dans l'image.

## testweb

DEPLOYED

simple docker webserver for arm64v8

TYPE	VERSION	PROFILE
docker	1.0	c1.tiny

Memory +

1.6%

CPU +

7.1%

✓ Activate

Upgrade

Delete

Dans l'écran suivant, vous pouvez sélectionner la quantité de ressources de calcul à affecter à l'application.

Afin de configurer la mise en réseau, cliquez sur **modifier** pour la configuration réseau, comme indiqué dans l'image.

✓ Activate App

### Resources

#### Resource Profile

Profile: c1.tiny

CPU 100 cpu-units

Memory 32 MB

Disk 10 MB

Avail. CPU (cpu-units) 1400 Avail. Memory (MB) 2048 Avail. Disk (MB) 4000

Advanced Settings

#### Network Configuration

Name	Network Config	Description	Action
eth0	mgmt-bridge300	none	edit

Add App Network Interface

#### Peripheral Configuration

Device Type	Name	Label	Status	Action
-------------	------	-------	--------	--------

Add Peripheral

Dans la configuration du réseau, choisissez de modifier le nom de réseau par défaut (mgmt-bridge300), puis cliquez sur **Paramètres d'interface**, comme indiqué dans l'image.



## ▼ Network Configuration

Name	Network Config	Description	Action
eth0	mgmt-bridge300	none	<a href="#">edit</a>

eth0

mgmt-bridge300 L2br network ▼

[Interface Setting](#)

Description (optional):

✓ OK

✗ Cancel

Dans la fenêtre contextuelle, choisissez d'utiliser une adresse IP dynamique, qui permet à l'application IOx de récupérer une adresse IP du serveur DHCP dans le VLAN que vous avez configuré sur l'interface AppGi1/1 ou de définir une adresse IP statique.

Il est important, comme indiqué à l'étape de mise en réseau, que vous spécifiez le VLAN à faire correspondre à l'accès/la liaison d'AppGi1/1 dans cette étape.

Actuellement, une interface d'application IOx doit avoir un ID de VLAN configuré même si l'interface AppGigabitEthernet1/1 est configurée en mode d'accès, car le trafic reçu par virteth2/L2br est toujours balisé.

**Interface Setting** ✕

**IPv4 Setting**

Static       Dynamic       Disable

IP/Mask	<input type="text" value="192.168.0.222"/> / <input type="text" value="24"/>
DNS	<input type="text"/>
Default Gateway IP	<input type="text" value="192.168.0.1"/>

**Vlan ID**

Vlan ID	<input type="text" value="1"/>
---------	--------------------------------

Enregistrez maintenant les paramètres réseau et activez l'application.

Si tout va bien, l'application doit se retrouver dans l'état Activé.

### Démarrer

Maintenant que l'application est activée, tout ce qui reste pour faire avancer les choses, est de démarrer l'application.

Cliquez sur le bouton **Démarrer** de l'application dans Local Manager, comme illustré dans l'image.

## testweb

ACTIVATED

simple docker webserver for arm64v8

TYPE  
docker

VERSION  
1.0

PROFILE  
c1.tiny

Memory \*

1.6%

CPU \*

7.1%

▶ Start

⊘ Deactivate

⚙ Manage

Après cette action, l'application doit être en état d'exécution et vous devez être en mesure d'atteindre l'application à l'aide du réseau configuré.

Si vous choisissez d'utiliser DHCP, accédez à **Manage > App-Info > interface-name > eth0**, afin de trouver l'adresse IP de l'application.

Pour ce document, cette application est un serveur Web simple, exécuté sur le port 9000. Pour le tester, accédez à l'adresse IP configurée (ou obtenue via DHCP) :

← → ↻ 🏠 ⓘ Not secure | 192.168.0.222:9000

# IOX python webserver on arm64v8

## Utiliser IOxclient

IOxclient est l'homologue CLI de Local Manager et est utilisé pour gérer et déployer des applications IOx sur des plates-formes IOx à l'aide de CLI. IOxclient parle avec le même service Web exécuté sur le périphérique IOx lui-même, dans ce cas l'IE3400.

Vous pouvez télécharger ioxclient à l'aide du lien suivant :

<https://developer.cisco.com/docs/iox/#!iox-resource-downloads>

## Préparer ioxclient

Une fois extrait, nous pouvons exécuter ioxclient et créer un profil qui indiquera à ioxclient comment atteindre l'IE3400 :

```
[jensd@cen8 ~]$ ioxclient profiles create
```

```
Config file not found : /home/jensd/.ioxclientcfg.yaml
Creating one time configuration..
Your / your organization's name : Cisco
Your / your organization's URL :
Your IOx platform's IP address[127.0.0.1] : 192.168.0.30
Your IOx platform's port number[8443] : 443
Authorized user name[root] : admin
Password for admin :
Local repository path on IOx platform[/software/downloads]:
URL Scheme (http/https) [https]:
API Prefix[/iox/api/v2/hosting/]:
Your IOx platform's SSH Port[2222]:
Your RSA key, for signing packages, in PEM format[]:
Your x.509 certificate in PEM format[]:
Activating Profile default
Saving current configuration
```

Les informations à entrer dépendent de la configuration réseau et de l'utilisateur priv15 spécifiés comme expliqué au début de cet article.

N'oubliez pas que le numéro de port IOx pour IE3400 est 443 par défaut et non le port 8443 suggéré.

## Déployer

La première étape consiste à déployer IE3400 comme application installée sur le périphérique de périphérie.

Ceci peut être fait comme indiqué ici via ioxclient :

```
[jensd@cen8 ~]$ ioxclient app install testweb package.tar
Currently active profile : default
Command Name: application-install
Using the package descriptor file in the project dir
Validating descriptor file package.yaml with package schema definitions
...
Sending request to install the app
Installation Successful. App is available at :
http://192.168.0.30/iox/api/v2/hosting/apps/testweb
Successfully deployed
```

Vous pouvez vérifier l'état de l'application à l'aide de la commande de liste d'applications :

```
[jensd@cen8 ~]$ ioxclient app list
Currently active profile : default
Command Name: application-list
List of installed App :
 1. testweb ---> DEPLOYED
```

## Activer

Maintenant que l'application est déployée, vous pouvez l'activer. Au cours de cette phase, vous spécifiez les ressources à utiliser par l'application.

Pour choisir les ressources de calcul que l'application peut utiliser ou configurer la mise en réseau, vous devez fournir des informations au format JSON.

Pour cet article, ce JSON est utilisé pour l'activation :

```
[jensd@cen8 ~]$ cat activation.json
{
  "resources": {
    "network": [{
      "interface-name": "eth0",
      "network-info": {"vlan-id": "1"},
      "network-type": "vlan",
      "ipv4": {
        "dns": "",
        "gateway": "192.168.0.1",
        "ip": "192.168.0.223",
        "mode": "static",
        "prefix": "24"
      }
    }]
  }
}
```

Les éléments ci-dessus définissent l'ID de VLAN et l'adresse IP corrects à utiliser pour l'application. Si vous souhaitez utiliser DHCP, vous pouvez supprimer le bloc ipv4.

Il est important, comme indiqué dans l'étape de mise en réseau, que vous spécifiez le VLAN à faire correspondre à l'accès/la liaison d'AppGi1/1 dans cette étape.

Actuellement, une interface d'application IOx doit avoir un ID de VLAN configuré même si l'interface AppGigabitEthernet1/1 est configurée en mode d'accès, car le trafic reçu par virteth2/L2br est toujours balisé.

Afin d'activer l'application IOx via la charge utile JSON, utilisez cette commande :

```
[jensd@cen8 ~]$ ioxclient app activate testweb --payload activation.json
Currently active profile : default
Command Name: application-activate
Payload file : activation.json. Will pass it as application/json in request body..
App testweb is Activated
```

De nouveau, vous pouvez utiliser la commande app list pour vérifier l'état après l'activation :

```
[jensd@cen8 ~]$ ioxclient app list
Currently active profile : default
Command Name: application-list
List of installed App :
 1. testweb ---> ACTIVATED
```

## Démarrer

Maintenant que l'application est activée, tout ce qui reste pour faire avancer les choses, est de démarrer l'application.

La commande ioxclient pour démarrer notre application est la suivante :

```
[jensd@cen8 ~]$ ioxclient app start testweb
Currently active profile : default
Command Name: application-start
App testweb is Started
```

Afin de savoir quelle adresse IP a été acquise auprès de DHCP, si vous avez ignoré la partie ipv4 lors de l'activation, vous pouvez utiliser cette commande :

```
[jensd@cen8 ~]$ ioxclient app info testweb | grep ipv4
  "ipv4": {
    "ipv4": "192.168.0.223",
    ...
```

Pour vérifier si la commande start a réussi, nous pouvons à nouveau utiliser la commande **ioxclient app list** ou simplement vérifier si l'application fait ce qu'elle est censée faire.

Pour cet article, cette application est un serveur Web simple, exécuté sur le port 9000 afin que nous puissions le tester en interrogeant l'adresse IP configurée (ou obtenue via DHCP) :

```
[jensd@cen8 ~]$ curl http://192.168.0.223:9000
<html><body><h1>IOX python webserver on arm64v8</h1></body></html>
```

## Utiliser l'interface de ligne de commande IOS-XE

Périphérique IOS-XE compatible IOx, qui expose la possibilité de déployer l'application et qui est géré directement à partir de l'interface de ligne de commande IOS-XE. Cela ne nécessite pas l'activation du serveur Web sur IOS-XE.

## Déployer

Puisque l'application est déployée à partir de l'interface de ligne de commande IOS-XE, copiez d'abord votre package d'application IOx vers un système de fichiers facilement accessible à partir de cette interface de ligne de commande.

Le plus simple est de copier d'abord le package d'application IOx (**package.tar**) sur flash :

```
ie3400#copy scp: flash:
Address or name of remote host []? 192.168.0.21
Source username [admin]? jensd
Source filename []? /home/jensd/package.tar
Destination filename [package.tar]?
Password:
  Sending file modes: C0644 16547840 package.tar
!!!!!!!!!!!!!!
16547840 bytes copied in 25.244 secs (655516 bytes/sec)
```

Une fois le package disponible, déployez-le pour IOx :

```
ie3400#app-hosting install appid testweb package flash:package.tar
Installing package 'flash:package.tar' for 'testweb'. Use 'show app-hosting list' for progress.
```

Une fois cette opération terminée, vous pouvez vérifier l'état de l'installation.

```
ie3400#sh app-hosting list
```

App id	State
testweb	DEPLOYED

## Activer

Après le déploiement, comme avec les autres méthodes, l'étape suivante consiste à activer l'application. Dans cette phase, spécifiez les ressources utilisées par l'application.

Vous devez au minimum configurer le réseau. Ceci peut être fait comme indiqué ici :

```
ie3400#conf t
Enter configuration commands, one per line. End with CNTL/Z.
ie3400(config)#app-hosting appid testweb
ie3400(config-app-hosting)#app-vnic AppGigabitEthernet trunk
ie3400(config-config-app-hosting-trunk)#vlan 1 guest-interface 0
ie3400(config-config-app-hosting-vlan-access-ip)#guest-ipaddress 192.168.0.224 netmask
255.255.255.0
ie3400(config-config-app-hosting-vlan-access-ip)#end
```

Cette commande indique à l'application de recevoir une adresse IP statique, à l'aide de l'ID de VLAN 1.

Il est important, comme indiqué dans l'étape de mise en réseau, que vous spécifiez le VLAN à faire correspondre à l'accès/la liaison d'AppGi1/1 dans cette étape.

Actuellement, une interface d'application IOx doit avoir un ID de VLAN configuré même si l'interface AppGigabitEthernet1/1 est configurée en mode d'accès, car le trafic reçu par virteth2/L2br est toujours étiqueté :

Après avoir configuré la configuration, poursuivez l'activation :

```
ie3400#app-hosting activate appid testweb
testweb activated successfully
Current state is: ACTIVATED
```

## Démarrer

Après l'activation, la dernière étape consiste à démarrer l'application.

Pour cela, utilisez la commande suivante :

```
ie3400#app-hosting start appid testweb
testweb started successfully
Current state is: RUNNING
```

Si vous n'avez pas défini d'adresse IP dans la phase d'activation et si vous avez décidé d'utiliser DHCP, vous pouvez obtenir l'adresse IP reçue par votre application :

```
ie3400#sh app-hosting detail | i IPv4
IPv4 address      : 192.168.0.224
```

Afin de vérifier si la commande start a réussi, nous pouvons simplement vérifier si l'application fait ce qu'elle est censée faire.

Pour ce document, cette application est un serveur Web simple, qui s'exécute sur le port 9000 afin que nous puissions le tester en interrogeant l'adresse IP configurée (ou obtenue via DHCP).

← → ↻ 🏠 ⓘ Not secure | 192.168.0.224:9000

# **IOX python webserver on arm64v8**