



システム要件

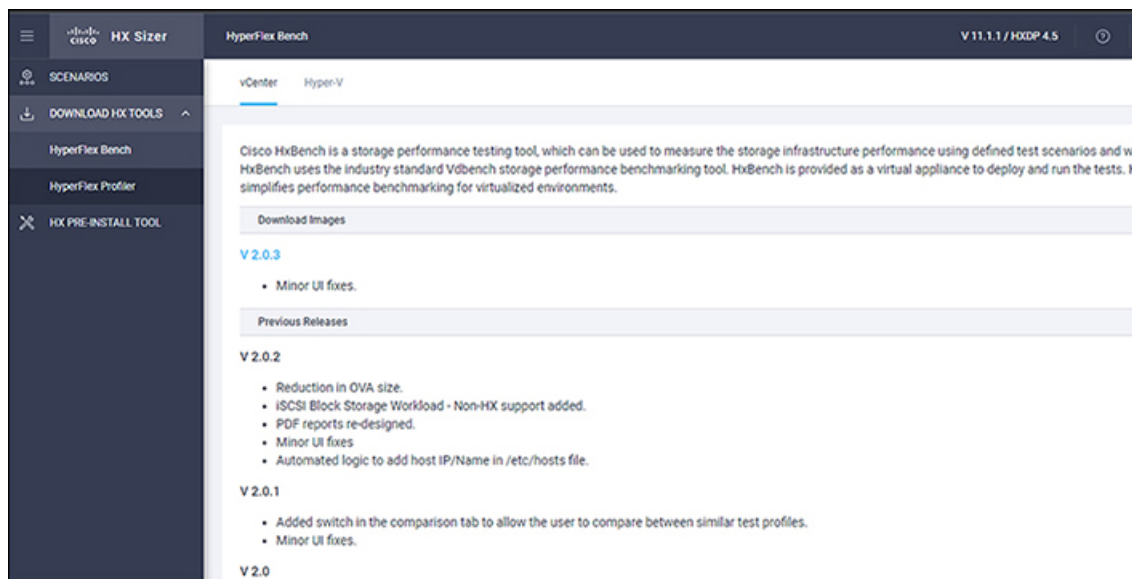
- [前提条件 \(1 ページ\)](#)
- [Kubernetes 環境用の HX プロファイラの設定 \(1 ページ\)](#)
- [プロファイラ サービスの使用 \(6 ページ\)](#)
- [Kubernetes アプリケーション ログの検索 \(8 ページ\)](#)

前提条件

- スクリプトファイルを実行する前に、以下の方法で `sshpass` をインストールしてください。
 - Debian/Ubuntu およびその派生システム：
 - `sudo apt-get install sshpass`
 - RedHat/CentOS ベースのシステム：
 - `yum install sshpass`
 - `dnf install sshpass [Fedora 22 以降のバージョン]`
- ストレージクラスの空き容量要件：40 GB 以上。

Kubernetes 環境用の HX プロファイラの設定

1. <https://hyperflexsizer.cloudapps.cisco.com> から `Cisco-Profiler-2.1-CSI.tar` ファイルをダウンロードします。
 - **HX ツールのダウンロード > HyperFlex プロファイラ > Kubernetes**



2. Cisco-HxProfiler-CSI-4.1.tar.gz を `tar -zxvf Cisco-HxProfiler-CSI-4.1.tar.gz` で解凍します。

```
Cisco-HxBench-2.1-CSI      Cisco-HxProfiler-CSI-4.1-original  Cisco-HxProfiler-CSI_old      temp
Cisco-HxProfiler-CSI     Cisco-HxProfiler-CSI-4.1.tar.gz    Cisco-HxProfiler-CSI-old.tar.gz
Cisco-HxProfiler-CSI-4.1 Cisco-HxProfiler-CSI-Final         Cisco-HxProfiler-CSI.tar.gz
cisco@jvdmade-M:~/prabhu$ cd Cisco-HxProfiler-CSI-4.1
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1$ ls
Deployment_yaml_files Docker_Image Script_files
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1$
```

3. `cd Cisco-HxProfiler-CSI-4.1/Script_files` を実行して、Script_files ディレクトリにアクセスします。

```
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1$ ls
Deployment_yaml_files Docker_Image Script_files
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1$ cd Script_files/
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$ ls -lrt
total 12
-rw-rw-r-- 1 cisco cisco 10042 Nov 28 13:42 profiler_deployment.sh
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$
```

4. `bash profiler_deployment.sh` スクリプトを実行して、スクリプトを実行します。

```
-rw-rw-r-- 1 cisco cisco 10042 Nov 28 13:42 profiler_deployment.sh
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$ ls
profiler_deployment.sh
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$ bash profiler_deployment.sh
```

5. プロンプトが表示されたら、ストレージクラス名を入力し、Enter ボタンを押します。

```
-rw-rw-r-- 1 cisco cisco 10042 Nov 28 13:42 profiler_deployment.sh
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$ ls
profiler_deployment.sh
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$ bash profiler_deployment.sh
-----
Enter storage class name: csi-hxcsi-default
```



(注) 間違ったストレージクラス名を入力すると、エラー応答が表示されます。3回、間違ったレイジ名を入力すると、展開プロセスが終了し、設定プロンプトに戻ります。その場合、1に戻り、プロセスを再起動してください。

6. ワーカー ノードにプロファイラ イメージをデプロイします：「すべてのワーカー ノードにプロファイラ イメージをデプロイしますか(推奨) Y/N： (Do you want to deploy profiler image in all worker nodes (Recommended) Y/N:)」という質問に答えます。
 - プロファイラをすべてのワーカー ノードにデプロイする場合は、**Y** と入力して **Enter** ボタンを押し、手順 8a に戻ります。
 - すべてのワーカー ノードにイメージをデプロイしない場合は、**N** を入力して **Enter** ボタンを押し、ステップ 9 にスキップします。

```
cisco@jvdmade-M:~/profiler_upgraded_rel$ cd Cisco-HxProfiler-CSI-4.1
cisco@jvdmade-M:~/profiler_upgraded_rel/Cisco-HxProfiler-CSI-4.1$ ls
Deployment_yaml_files Docker_Image Script_files
cisco@jvdmade-M:~/profiler_upgraded_rel/Cisco-HxProfiler-CSI-4.1$ cd Script_files/
cisco@jvdmade-M:~/profiler_upgraded_rel/Cisco-HxProfiler-CSI-4.1/Script_files$ ls
profiler_deployment.sh
cisco@jvdmade-M:~/profiler_upgraded_rel/Cisco-HxProfiler-CSI-4.1/Script_files$ bash profiler_deployment.sh
-----
Enter storage class name: csi-hxcsi-default
Do you want to deploy profiler image in all worker nodes(recommended) -Y/N-: y
```

1. プロンプトが表示されたら、ワーカー ノードのユーザー名とパスワードを入力し、**Enter** ボタンを押します。「イメージが正常にデプロイされました (Image deployed successfully)」というメッセージが表示されることを確認してください。

```
profiler_deployment.sh
cisco@jvdmade-M:~/prabhu/Cisco-HxProfiler-CSI-4.1/Script_files$ bash profiler_deployment.sh
-----
Enter storage class name: csi-hxcsi-default
Do you want to deploy profiler image in all worker nodes(recommended) -Y/N-: Y
Enter 10.2.14.251 username : cisco
Enter 10.2.14.251 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.251 closed.
Image deployed Successfully
Enter 10.2.14.252 username : █
```

2. 手順 8a を繰り返して、すべてのワーカー ノードにイメージをデプロイします。

```

Enter storage class name: csi-hxcsi-default
Do you want to deploy profiler image in all worker nodes(recommended) -Y/N-: y
Enter 10.2.14.251 username : cisco
Enter 10.2.14.251 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.251 closed.
Image deployed Successfully
Enter 10.2.14.252 username : cisco
Enter 10.2.14.252 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.252 closed.
Image deployed Successfully
Enter 10.2.14.253 username : cisco
Enter 10.2.14.253 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.253 closed.
Image deployed Successfully
The namespace used for deployment is profiler do you want to change -Y/N-: y

```

7. すべてのワーカー ノードにイメージをデプロイしない場合は、**N**を入力して **Enter** ボタンを押します。特定のノードにイメージをデプロイする場合は、ワーカーノードの IP、ユーザー名、およびパスワードを入力します。

```

-----
Enter storage class name: csi-hxcsi-default
Do you want to deploy profiler image in all worker nodes(recommended) -Y/N-: n
Enter workernode ip : 10.2.14.252
Enter 10.2.14.252 username : cisco
Enter 10.2.14.252 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.252 closed.
Image deployed Successfully
Do you want to deploy image in any worker node -Y/N-: n

```

8. **デプロイメントの名前空間**：デプロイメントに使用されるデフォルトの名前空間は profiler (プロファイラ) です。

提供された名前空間で、Elasticsearch および postgres ポッドが名前空間にデプロイされているかどうかを確認します。そうである場合は：

- 名前空間全体を削除して新しい名前空間をデプロイし、手順 11 に進みます。
1. 削除して新しいものを作成する場合は、**Y**を入力して **Enter** ボタンを押します。デフォルトのプロファイラ名前空間が削除され、profiler という名前の新しい名前空間が作成されます。**Y** と入力して **Enter** ボタンを押し、アクションを確認します。

```

-----
Enter storage class name: csi-hxcsi-default
Do you want to deploy profiler image in all worker nodes(recommended) -Y/N-: n
Enter workernode ip : 10.2.14.251
Enter 10.2.14.251 username : cisco
Enter 10.2.14.251 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.251 closed.
Image deployed Successfully
Do you want to deploy image in any worker node -Y/N-: n
The namespace used for deployment is profiler do you want to change -Y/N-: n
Pods Elasticsearch and Postgres are running do you want to delete ns profiler-deploy1 and create them
The namespace profiler-deploy1 will be deleted and created freshly do you want to continue -Y/N-: y

```

2. 他の名前空間にデプロイする場合は、**y** と入力してから目的の名前空間を入力します。

```

-----
Enter storage class name: csi-hxcsi-default
Do you want to deploy profiler image in all worker nodes(recommended) -Y/N-: n
Enter workernode ip : 10.2.14.251
Enter 10.2.14.251 username : cisco
Enter 10.2.14.251 password :
Copying docker Images
Loading docker Images
Connection to 10.2.14.251 closed.
Image deployed Successfully
Do you want to deploy image in any worker node -Y/N-: n
The namespace used for deployment is profiler do you want to change -Y/N-: y
namespace must consist of lower case alphanumeric characters or '-', and must start and end with an alphanumeric
Enter the namespace: profiler-deploy1
Pods Elasticsearch and Postgres are running do you want to delete ns profiler-deploy1 and create them -Y/N-: y
The namespace profiler-deploy1 will be deleted and created freshly do you want to continue -Y/N-: y

```

9. 新しい名前空間名を入力し、**Enter** ボタンを押します。



- (注) 名前空間名の要件：使用できるのは小文字の英数字と「-」（ダッシュ）です。名前の先頭と末尾は英数字にする必要があります。

すべてのノードに Elasticsearch と Postgres yaml ファイルのデプロイが作成されます。

```

The namespace used for deployment is profiler do you want to change -Y/N-: y
namespace must consist of lower case alphanumeric characters or '-', and must start and end with an alphanumeric character
Enter the namespace: new-profiler-test
namespace/new-profiler-test created
configmap/elasticsearch created
persistentvolumeclaim/elasticsearch-pvc created
deployment.apps/elasticsearch created
service/elasticsearch created
persistentvolumeclaim/postgres-pvc created
configmap/postgres-config created
deployment.apps/postgres created
service/postgres created
serviceaccount/mymonitoring created
Error from server (AlreadyExists): error when creating "../Deployment_yaml_files/create_cluster_role.yaml": clusterroles.rbac.authorization.k8s.io "mymonitoring-clusterrole" already exists
clusterrolebinding.rbac.authorization.k8s.io/mymonitoring-clusterrole-binding-new-profiler-test created

```

10. 現在の UNIX パスワードのプロンプトで、appadmin GUI アカウントのパスワード（デフォルトのパスワードは **password**）を入力し、**Enter** ボタンを押します。

```

clusterrolebinding.rbac.authorization.k8s.io/mymonitoring-clusterrole-binding-new-profiler-test created
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
Success

```

11. 新しい UNIX パスワードのプロンプトで、新しいユーザー定義パスワードを入力し、**Enter** ボタンを押します。

```
(current) UNIX password:
Enter new UNIX password:
Retype new UNIX password:
Success
```

12. 新しい UNIX パスワードを再入力するプロンプトで、新しいパスワードを入力して Enter ボタンを押します。これでプロファイラのデプロイメントが作成されます。

```
Success
secret/secret-monitoring-auth created
persistentvolumeclaim/profiler-pvc created
deployment.apps/profiler created
service/profiler created
```

13. 結果として、ポート番号を持つ最終的な `profiler-ip` が得られます。IP アドレスをコピーしてブラウザに貼り付け、UI を取得します。

```
-----profiler-ip-----
10.2.14.251:32042
```

プロファイラ サービスの使用

HX Workload Profiler でサービスを開始し、停止するには、`profiler_service.sh` コマンドを使用します。

次の表に、プロファイラ サービスを使用するための大まかな手順を示します。

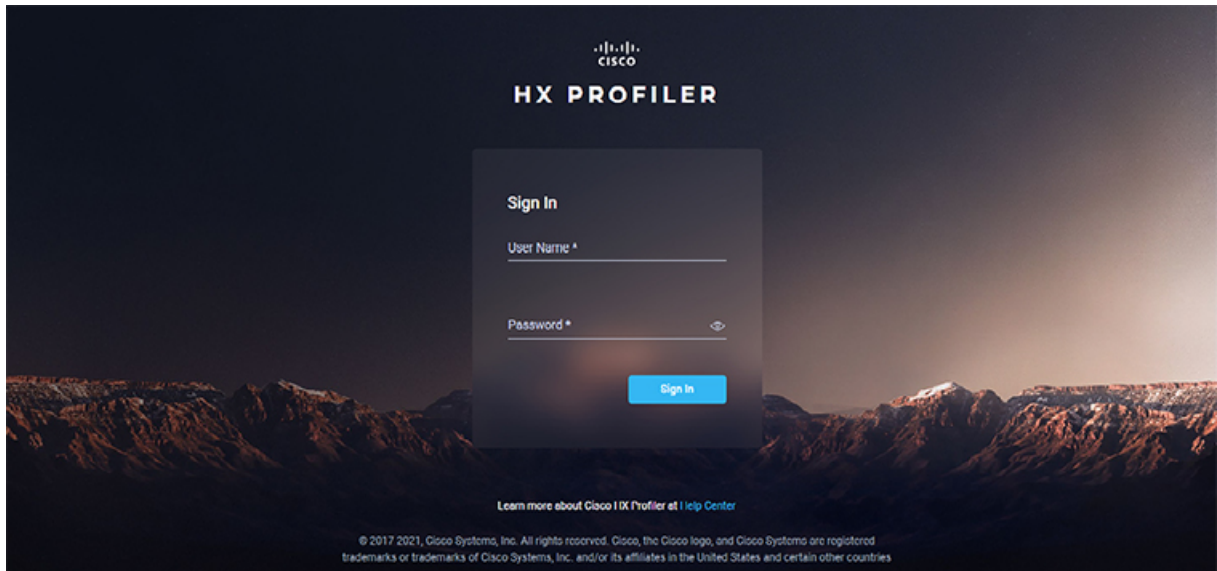
タスク	参照先
プロファイラ サービスの開始	プロファイラ サービスの開始 (8 ページ)
プロファイラ サービスの停止	プロファイラ サービスの停止 (8 ページ)

プロファイラへのログイン

HX ワークロードプロファイラのユーザー インターフェイス (UI) は、認証のためにシステムのクレデンシャルを使用します。

ステップ 1 UI にアクセスするには、ブラウザ ウィンドウを起動し、`http://< IP:PORT_NUMBER >` または `http://< IP:PORT_NUMBER >/profiler/index.html` または `http://< IP:PORT_NUMBER >/profiler/index.html` を入力します。ここで、`IP:PORT_NUMBER` は、展開スクリプトの出力です。

HX プロファイラ UI が表示されます。

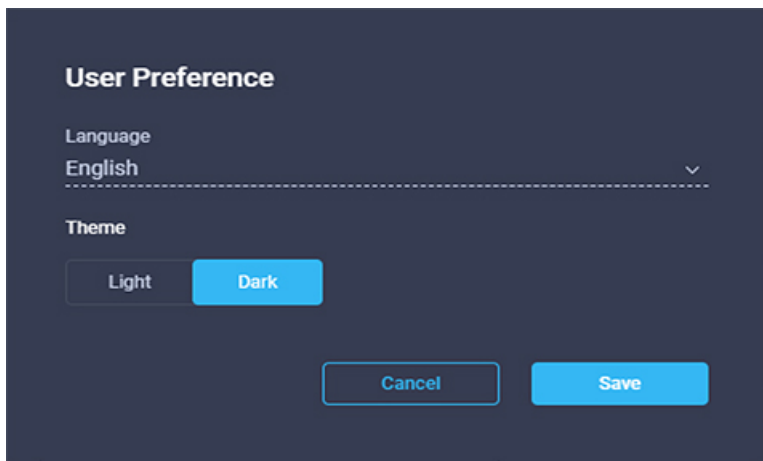


ステップ 2 指示された場合、次のクレデンシャルで UI にログインします。

ユーザー名: **monitoring**

パスワード: <インストール ワークフロー中に設定した新しいパスワード>

ステップ 3 UI の右上隅にある [ユーザー設定 (User Preference)] オプションでは、[言語 (Language)] または [テーマ (Theme)] を構成できます。



[キャンセル (Cancel)] または [保存 (Save)] をクリックして続行します。

ステップ 4 完了したら、ページの右上にある [ログアウト (Logout)] をクリックして、ユーザーセッションを終了できます。

プロファイラ サービスの開始

プロファイラ サービスを開始するには、プロファイラ ポッド内で以下のコマンドを実行します。

次のコマンドを実行します : `sh podscript.sh`。

プロファイラ サービスの停止

プロファイラ サービスを停止するには、プロファイラ ポッド内で次の手順を実行します。

ステップ1 `cd profiler_scripts/` コマンドを実行します。

ステップ2 `sudo sh profiler_service.sh stop` コマンドを実行します。

Kubernetes アプリケーション ログの検索

HX Workload プロファイラ ログは次の場所で確認できます。

表 1: アプリケーション ログ

Log	パス
サーバ	/home/monitoring/monitor/server.log
コントローラ	/home/monitoring/controller/logs/*
モニタ (Monitor)	/home/monitoring/monitor/monitor/monitor.log