

証明書ベース認証を使用したISE SFTPの設定

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[設定](#)

[1. CentOSサーバの設定](#)

[2. ISEリポジトリの設定](#)

[3. ISEサーバでキーペアを生成する](#)

[3.1. ISE GUI](#)

[3.2. ISE CLI](#)

[4.統合](#)

[確認](#)

[関連情報](#)

概要

このドキュメントでは、CentOSディストリビューションを使用するLinuxサーバを、Identity Services Engine(ISE)に対する公開キーインフラストラクチャ(PKI)認証を使用したSecure File Transfer Protocol(SFTP)サーバとして設定する方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- 一般的なISEの知識
- ISEリポジトリの設定
- Linuxの基本的な一般知識

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- ISE 2.2
- ISE 2.4
- ISE 2.6
- ISE 2.7
- ISE 3.0
- CentOS Linuxリリース8.2.2004 (コア)

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。ネットワークが稼働中の場合は、コマンドの潜在的な影響について理解しておいてください。

背景説明

ファイル転送のセキュリティを強化するために、ISEはSFTPを介してPKI証明書を使用して認証し、リポジトリファイルへのより安全なアクセス方法を確保できます。

設定

1. CentOSサーバの設定

1.1ルートディレクトリをルートユーザとして作成します。

```
mkdir -p /cisco/engineer
```

1.2.ユーザグループの作成

```
groupadd tac
```

1.3.このコマンドは、ユーザーをメインディレクトリ（ファイル）に追加し、ユーザーがグループエンジニアに属することを指定します。

```
useradd -d /cisco/engineer -s /sbin/nologin engineer  
usermod -aG tac engineer
```

注：コマンドの/sbin/nologin部分は、ユーザがセキュアシェル(SSH)を介してログインできないことを示しています。

1.4.ファイルをアップロードするディレクトリの作成に進みます。

```
mkdir -p /cisco/engineer/repo
```

1.4.1ディレクトリファイルの権限を設定する。

```
chown -R engineer:tac /cisco/engineer/repo  
find /cisco/engineer/repo -type d -exec chmod 2775 {} \+  
find /cisco/engineer/repo -type f -exec chmod 664 {} \+
```

1.5. CentOSサーバが証明書のチェックを実行するディレクトリとファイルを作成します。

ディレクトリ:

```
mkdir /cisco/engineer/.ssh  
chown engineer:engineer /cisco/engineer/.ssh  
chmod 700 /cisco/engineer/.ssh
```

File :

```
touch /cisco/engineer/.ssh/authorized_keys
chown engineer:engineer /cisco/engineer/.ssh/authorized_keys
chmod 600 /cisco/engineer/.ssh/authorized_keys
```

1.6. sshd_configシステムファイルにログイン権限を作成します。

このファイルを編集するには、このコマンドでvim Linuxツールを使用します。

```
vim /etc/ssh/sshd_config
```

1.6.1指定した行を次に追加します。

```
#Subsystem sftp /usr/libexec/openssh/sftp-server
Subsystem sftp internal-sftp
Match Group tac
ChrootDirectory %h
X11Forwarding no
AllowTCPForwarding no
ForceCommand internal-sftp
```

1.7. sshd_configシステムファイルの同期を確認するために、コマンドを実行します。

```
sshd -t
```

注：出力は、ファイルの構文が正しいことを意味します。

1.8. SSHサービスの再起動に進みます。

```
systemctl restart sshd
```

注：一部のLinuxサーバにはselinuxが適用されており、このパラメータを確認するには、getenforceコマンドを使用できます。強制モードの場合はpermissiveに変更することをお勧めします。

1.9. (オプション) semanage.confファイルを編集して、強制をpermissiveに設定します。

```
vim /etc/selinux/semanage.conf
```

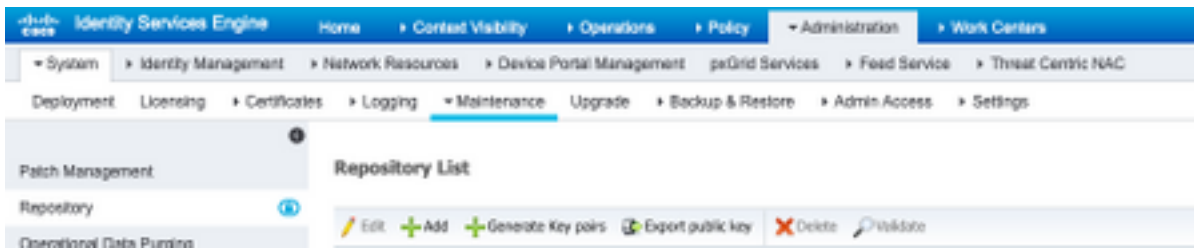
コマンドsetenforce0を追加します。

```
setenforce0
```

2. ISEリポジトリの設定

2.1. ISEグラフィックユーザインターフェイス(GUI)を使用してリポジトリを追加します。

[管理(Administration)] > [システムメンテナンス(System Maintenance)] > [リポジトリ(Repository)]
> [追加(Add)]に移動します



2.2. リポジトリの適切な設定を入力します。

[Repository List](#) > [Add Repository](#)

Repository Configuration

* Repository Name

* Protocol

Location

* Server Name

* Path

Credentials

* Enable PKI authentication

* User Name

* Password

注：エンジニアのルートディレクトリではなくrepoディレクトリにアクセスする必要がある場合は、ターゲットパスは/repo/である必要があります。

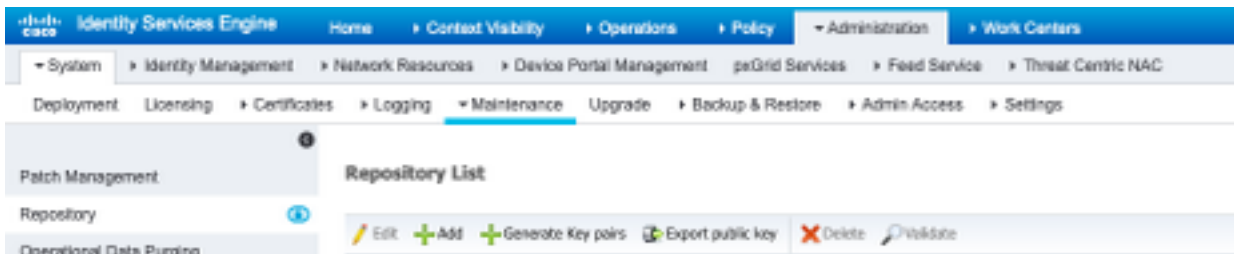


3. ISEサーバでキーペアを生成する

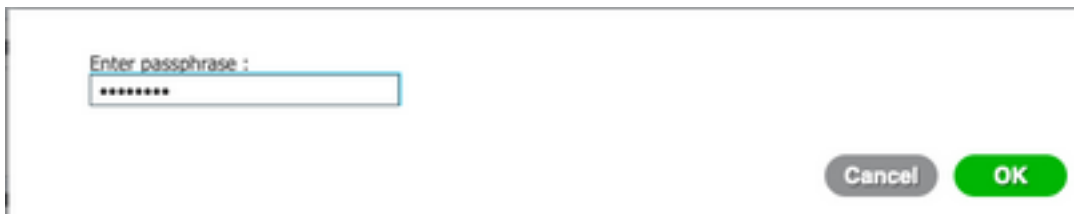
3.1. ISE GUI

図に示すように、[管理] > [システムメンテナンス] > [リポジトリ] > [キーペアの生成]に移動します。

注：リポジトリへの完全な双方向アクセスを実現するには、ISE GUIおよびコマンドラインインターフェイス(CLI)からキーペアを生成する必要があります。



3.1.1. パスフレーズを入力します。これは、キーペアを保護するために必要です。

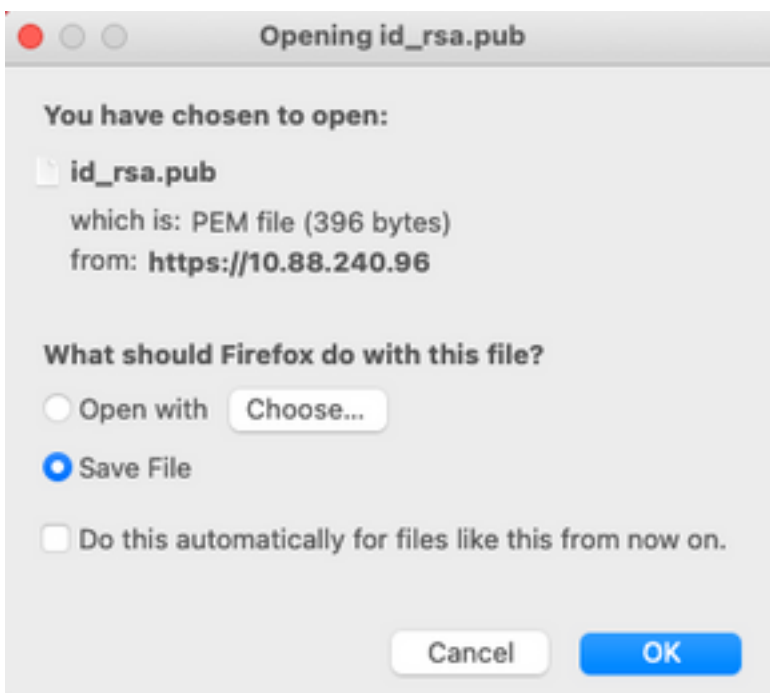


注：公開キーをエクスポートする前に、まずキーペアを生成します。

3.1.2. 公開キーのエクスポートに進みます。

[管理(Administration)] > [システムメンテナンス(System Maintenance)] > [リポジトリ(Repository)] > [公開キーのエクスポート(Export public key)]に移動します。

[Export public key]を選択します。id_rsa.pubという名前のファイルが生成されます(これ以降の参照のために保存してください)。



3.2. ISE CLI

3.2.1. リポジトリの設定を終了するノードのCLIに移動します。

注：この時点から、PKI認証を使用してSFTPリポジトリへのアクセスを許可する各ノードで、次の手順が必要になります。

3.2.2.このコマンドを実行して、LinuxサーバのIPをhost_keyシステムファイルに追加します。

```
crypto host key add host <Linux server IP>
ise24https/admin# crypto host_key add host 10.88.240.102
host key fingerprint added
# Host 10.88.240.102 found: line 2
10.88.240.102 RSA_SHA256:sFA1b+NujB8NxIx4zhS/7Fj1hyHRkJLKyLhJCLteSpE
```

3.2.3.パブリックCLIキーを生成する。

```
crypto key generate rsa passphrase <passphrase>
ise24https/admin# crypto key generate rsa passphrase admin123
```

3.2.4.次のコマンドを使用して、ISEのCLIから公開キーファイルをエクスポートします。

```
crypto key export <name of the file> repository <repository name>
```

注：公開キーファイルをエクスポートできる、以前にアクセス可能なリポジトリが必要です。

```
ise24https/admin# crypto key export public repository FTP
```

4.統合

4.1. CentOSサーバにログインします。

authorized_keyファイルを以前に構成したフォルダに移動します。


4.2.認証キーファイルの編集

vimコマンドを実行して、ファイルを変更します。

```
vim /cisco/engineer/.ssh/authorized_keys
```

4.3.「キーペアの生成」セクションのステップ4および6で生成されたコンテンツをコピーして貼り付けます。

ISE GUIから生成された公開キー：



```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQJcgg87051c@wTP16Grmf8r3Mnx+ogorSuTmPToC+0zjt16iAbTIjs/
PZreawf9urQXg0xEnSHa1kF0FPAJrKqoLBlRGusZelyNxVL06t1Vfx8IEIEhQTd9dy9uRQ3XIDUigC3q5jFPs0pG4rHsHmg0GbZJL
BNFvUgRjw0015x8IylyeLdt16oL7RFoTU3Y51hvfGXSI5ZhxGKsXjm2hA0+rkbffPfy37LT7w8HpAEaEVgLXL4o3mFUrdKCc04
ptPQ7B12vvIH0hcZqG+Gnpw3U+SHxGwks1fc393vCA4smzFnuNZ4/Q1jLppP4s2hqrAVedr+r90z+8XdsxV root@ise24https
```

ISE CLIから生成された公開キー：

```

public
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCAH+SANAYb47+NXFYuz06s0+gSykTRrGfdMryIiitCwBs0bGs5yc9S8VKpLyyocsIvco4/
vF/pSHoTE1R3wrZTL1vCIUrGnnqdQv4+0YnIbJ/f8EgZnXQ+fLK8oyLeVxPgd8cewL3HMV8giQHLizAdXtQB886tkno40cmT/
HAYXQ/a9YRZ1l29D6pjK5WyuTkbUxwVn9hx/
5E5zp34pFr9opq+UaTNX0yYuuJ328FGEFdKuFBSUjAokP0nJTLN8GdLAQ6x4kkkcXwXkT8F1saPZwyJuqY8FNWtyiFIVY5Ct5G0zm
D0Cj6vMaV8L7GZdDI4NZHn7llpptqJFYAb65QB admin@ise24htts

```

Linuxサーバ上のAuthorized_keyファイル：

```

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCAH+SANAYb47+NXFYuz06s0+gSykTRrGfdMryIiitCwBs0bGs5yc9S8VKpLyyocsIvco4/vF/pSHoTE1R3wrZTL1vCIUrGnnqdQv4+0YnIbJ/f8EgZnXQ+fLK8oyLeVxPgd8cewL3HMV8giQHLizAdXtQB886tkno40cmT/HAYXQ/a9YRZ1l29D6pjK5WyuTkbUxwVn9hx/5E5zp34pFr9opq+UaTNX0yYuuJ328FGEFdKuFBSUjAokP0nJTLN8GdLAQ6x4kkkcXwXkT8F1saPZwyJuqY8FNWtyiFIVY5Ct5G0zmD0Cj6vMaV8L7GZdDI4NZHn7llpptqJFYAb65QB admin@ise24htts

```

...

```

:wq!

```

4.4. ファイルにキーを貼り付けた後、Escキーを押し、wqを実行します！コマンドを発行して、ファイルを保存します。

確認

1. Linuxサーバから、このコマンドをrootとして実行します。

```
tail -f /var/log/secure
```

図に示すように、出力を表示する必要があります。

```

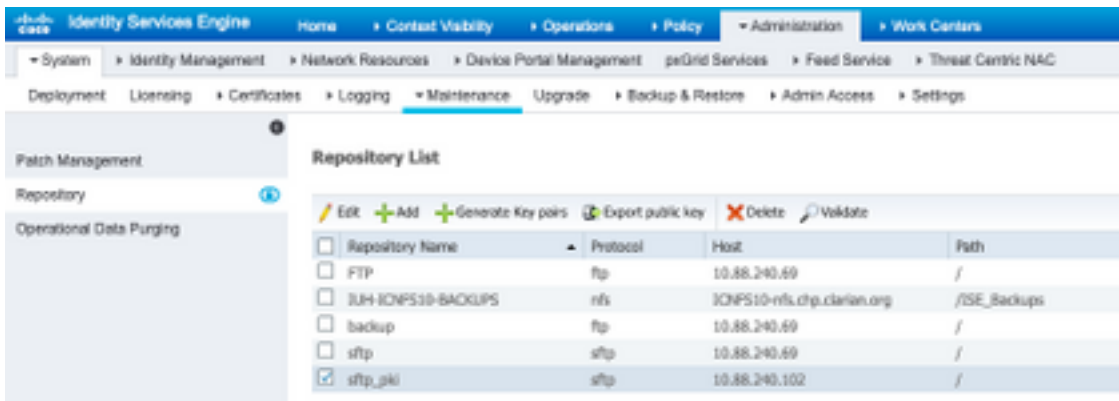
[[root@localhost ~]# tail -f /var/log/secure
Apr 12 21:37:53 localhost sshd[668112]: Accepted publickey for root from 10.24.140.234 port 61159 ssh2: RSA SHA256:MNHNp2AtvX080bTswgPLK0G8aWfUueGbkEW1EkcaeXU
Apr 12 21:37:53 localhost systemd[668117]: pam_unix(systemd-user:session): session opened for user root by (uid=0)
Apr 12 21:37:53 localhost sshd[668112]: pam_unix(sshd:session): session opened for user root by (uid=0)
Apr 12 21:38:27 localhost sshd[668201]: Accepted publickey for engineer from 10.24.140.234 port 61164 ssh2: RSA SHA256:MNHNp2AtvX080bTswgPLK0G8aWfUueGbkEW1EkcaeXU
Apr 12 21:38:27 localhost systemd[668208]: pam_unix(systemd-user:session): session opened for user engineer by (uid=0)
Apr 12 21:38:27 localhost sshd[668201]: pam_unix(sshd:session): session opened for user engineer by (uid=0)

```

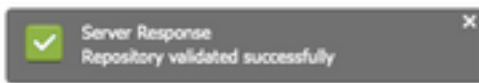
2. ISE検証の場合。

GUIで[Administration] > [System] > [Maintenance] > [Repository]に移動します。

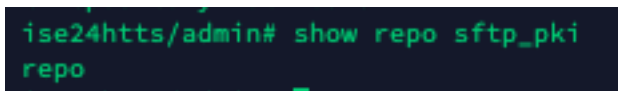
リポジトリ・リストから目的のリポジトリを選択し、「検証」を選択します。



画面の右下隅に[Server Response]というポップアップが表示されます。



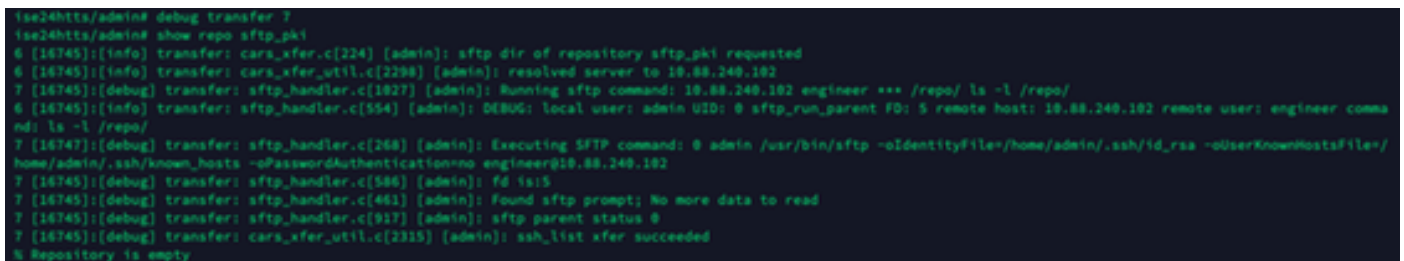
CLIから、`show repo sftp_pki`コマンドを実行して、キーを検証します。



ISEをさらにデバッグするには、CLIで次のコマンドを実行します。

`debug transfer 7`

次の図に示すように、出力を表示する必要があります。



関連情報

https://www.cisco.com/c/en/us/td/docs/security/ise/2-2/admin_guide/b_ise_admin_guide_22/b_ise_admin_guide_22_chapter_01011.html