

エイリアス設定のインポートとエクスポートの自動化

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[別名テーブルのエクスポートおよびインポート](#)

[Bashスクリプトを使用した別名テーブルのエクスポート](#)

[説明](#)

[1.- SSHのキーとパスの設定](#)

[2.- プロキシに接続し、SSHトンネルをセットアップする](#)

[3.- 続行する前に5秒間一時停止します](#)

[4.- リモートシステムからのaliasconfigファイルのエクスポート](#)

[5.- ローカルディレクトリへのファイルのダウンロード](#)

[スクリプトの最終処理](#)

[SSHキー](#)

[エクスポートファイルの確認](#)

[Bashスクリプトを使用した別名テーブルのインポート](#)

[説明](#)

[1.- SSHパスとキー設定](#)

[2.- 現在の日時を取得する](#)

[3.- 新しいaliasconfigファイルをリモートESAサーバにアップロードする](#)

[4.- 新しいaliasconfigファイルをインポートし、コメントを付けてコミットする](#)

[5.- 現在のaliasconfigを出力し、新しいローカルファイルに保存する](#)

[変更の確認](#)

[新しいaliasconfigテーブルエントリの確認](#)

[コミット変更の確認](#)

[スクリプトの柔軟性](#)

[最後の考察](#)

[参照リンク](#)

はじめに

このドキュメントでは、Eメールセキュリティアプライアンス(ESA)内のエイリアス設定のインポートおよびエクスポートのタスクを自動化する手順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。

- Cisco Secure Email Gateway(SEG/ESA)AsyncOS 16.0.2
- クラウドアプライアンスへのコマンドラインインターフェイス(CLI)アクセス
- LinuxのCLI
- シェルスクリプト

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアに基づいています。

- クラウドEメールセキュリティアプライアンス(CESA)
- バッシュ

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、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

目的は特定のタスクを自動化することですが、一部のプロセスは通常、手動による介入を必要とします。ただし、現在のエイリアス設定をインポートおよびエクスポートする場合は、これらのタスクを完全に自動化できるため、手動で入力する必要はありません。

別名テーブルのエクスポートおよびインポート

エイリアステーブルをインポートするには、最初にSSHおよびSCPアクセスを確認し、電子メールゲートウェイに接続できることを確認します。

次に進む前に、アプライアンス内に別名テーブルが存在している必要があります：

```
(Machine esa1.xyz.iphmx.com) (SERVICE)> clustermode cluster; aliasconfig print  
  
test: test@example.com, test@example2.com, test@example3.com  
test2: test@domain.com, test@domain2.com, test@domain3.com  
(Cluster Hosted_Cluster) (SERVICE)>
```

aliasconfigコマンドのexportサブコマンドを使用して既存の別名テーブルをバックアップすると、ファイル（指定した名前）が生成され、リスナーの/configurationディレクトリに保存されます。

Bashスクリプトを使用した別名テーブルのエクスポート

この場合、bashスクリプトはCESアプライアンスに接続し、エイリアスファイルのエクスポートを続行します

bashスクリプトは次のように構成されています。

```
#!/bin/bash

# Configuration of SSH keys and paths
PROXY_KEY="/full/path/folder/.ssh/id_rsa"
SECOND_KEY="/full/path/folder/.ssh/id_rsa"
LOCAL_PORT="2200"
PROXY_USER="dh-user"
PROXY_HOST="f4-ssh.iphmx.com"
TARGET_HOST="esa1.xyz.iphmx.com"
REMOTE_USER="local_server_user"
REMOTE_FILE="/configuration/filename.csv"
LOCAL_DIR="/full/path/folder/Downloads"
LOCAL_FILE_PATH="${LOCAL_DIR}/aliasconfig-file.csv"

# 1. Connect to the proxy and set up the SSH tunnel
echo "Establishing connection to the proxy..."
ssh -i "$PROXY_KEY" -l "$PROXY_USER" -N -f "$PROXY_HOST" -L "$LOCAL_PORT:${TARGET_HOST}:22"
if [ $? -ne 0 ]; then
    echo "Error: Failed to establish connection to the proxy."
    exit 1
fi
echo "Proxy connection established."

# Pause for 5 seconds before proceeding
sleep 5

# 2. Export the aliasconfig file from the remote system
echo "Exporting aliasconfig file from the remote system..."
ssh -i "$SECOND_KEY" "$REMOTE_USER"@127.0.0.1 -p "$LOCAL_PORT" 'clustermode cluster; aliasconfig export'
if [ $? -ne 0 ]; then
    echo "Error: Failed to export the aliasconfig file."
    exit 1
fi
echo "Aliasconfig file successfully exported."

# Pause for 5 seconds before proceeding
sleep 5

# 3. Download the file to the local directory
echo "Downloading file to the local directory..."
scp -i "$SECOND_KEY" -P "$LOCAL_PORT" -O "$REMOTE_USER"@127.0.0.1:"$REMOTE_FILE" "$LOCAL_DIR" 2>/dev/null
if [ $? -ne 0 ]; then
    echo "Error: Failed to download the file to the local directory."
    exit 1
fi
echo "File successfully downloaded to: $LOCAL_FILE_PATH"

# Pause for 5 seconds before finalizing
sleep 5
```

```
# Finalizing the script
echo "Process completed successfully."
exit 0
```

説明

1.- SSHのキーとパスの設定

- PROXY_KEYおよびSECOND_KEY：認証に使用されるSSH秘密キーファイルへのフルパス。この場合、両方のキーは同じパスに設定されます。
- 例： /full/path/folder/.ssh/id_rsa
- LOCAL_PORT:SSHトンネルのローカルポート(2200)を指定します。
- PROXY_USER：プロキシサーバへの接続に使用するユーザ名。
- PROXY_HOST：プロキシサーバのホスト名。
- TARGET_HOST：ターゲットホストの完全修飾ドメイン名(FQDN)。esa1.xyz.iphmx.comに更新されます。
- REMOTE_USER:SSHトンネル経由でリモートアプライアンスに接続するために使用されるユーザ名。
- REMOTE_FILE：エクスポートされたファイルが保存されているリモートシステム上のパス (/configuration/filename.csv)。
- LOCAL_DIR：ファイルを保存するローカルディレクトリは/full/path/folder/Downloadsに設定されています。
- LOCAL_FILE_PATH：ダウンロードしたファイルの完全なローカルパス。
\${LOCAL_DIR}/aliasconfig-file.csvに更新されます。

2. – プロキシに接続し、SSHトンネルをセットアップする

```
echo "Establishing connection to the proxy..."
ssh -i "$PROXY_KEY" -l "$PROXY_USER" -N -f "$PROXY_HOST" -L "$LOCAL_PORT:${TARGET_HOST}:22"
```

- 目的:
 - ターゲットホストとのセキュアな通信のために、プロキシサーバへのSSHトンネルをセットアップします。
- アップデート:
 - SSH秘密キーは/full/path/folder/.ssh/id_rsaにあります。
 - ターゲットホスト名がesa1.xyz.iphmx.comに更新されました。
- エラー処理:
 - 接続が失敗すると、エラーメッセージが表示され、スクリプトはエラーコードで終了します(exit 1)。

3. – 続行する前に5秒間一時停止します

- 目的:
 - 次のステップに進む前に、SSHトンネルが完全に確立されるように遅延を導入します。

4. – リモートシステムからのaliasconfigファイルのエクスポート

```
echo "Exporting aliasconfig file from the remote system..."
ssh -i "$SECOND_KEY" "$REMOTE_USER"@127.0.0.1 -p "$LOCAL_PORT" 'clustermode cluster; aliasconfig export
```

- 目的:
 - SSHトンネルを介してターゲットホストに接続し、エイリアス設定をaliasconfig-file.csvという名前のファイルにエクスポートします。
- アップデート:
 - エクスポートファイルのファイル名がaliasconfig-file.csvに更新されました。
- 出力ダイレクト:
 - 2>/dev/nullは、SSHコマンドからのエラーメッセージを表示しません。
- エラー処理:
 - エクスポートが失敗すると、エラーメッセージが表示され、スクリプトが終了します。

5. – ローカルディレクトリへのファイルのダウンロード

```
echo "Downloading file to the local directory..."
scp -i "$SECOND_KEY" -P "$LOCAL_PORT" -O "$REMOTE_USER"@127.0.0.1:"$REMOTE_FILE" "$LOCAL_DIR" 2>/dev/nu
```

- 目的:
 - scpを使用して、エクスポートされたファイルをリモートシステムからローカルディレクトリに安全にコピーします。
- アップデート:
 - ローカルファイルは、aliasconfig-file.csvとして/full/path/folder/Downloadsディレクトリに保存されます。
- エラー処理:
 - ファイルのダウンロードが失敗すると、エラーメッセージが表示され、スクリプトが終了します。

スクリプトの最終処理

```
echo "Process completed successfully."
exit 0
```

- 目的:

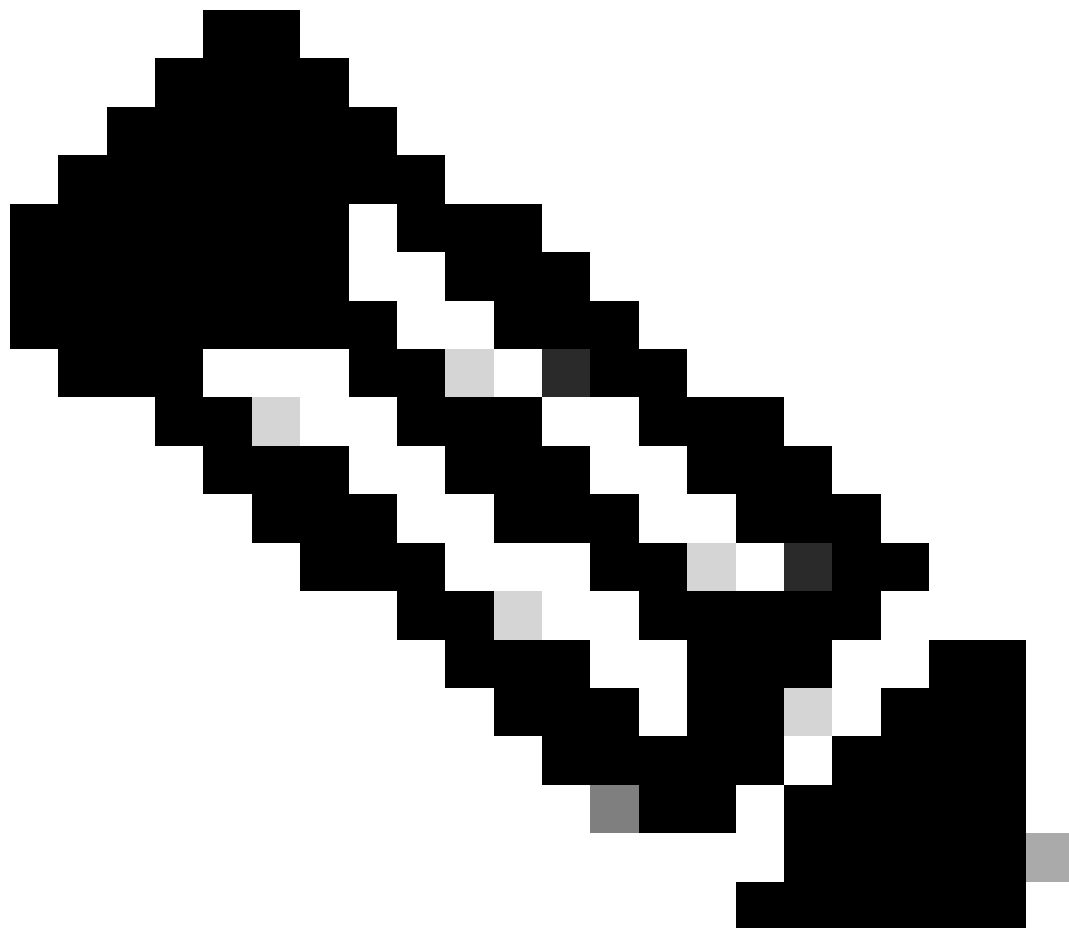
– 成功メッセージを出力し、すべての操作が正常に完了したことを示す成功コード(exit 0)を付けてスクリプトを終了します。

SSHキー

スクリプトでは、PROXY_KEYとHOST_KEYの2つの変数をメモできます。これらのキーは同じでも異なっている場合もあります。

PROXY_KEYは、CESAサーバにジャンプするために必須であるプロキシクラウドへの接続に使用されます。

HOST_KEYはローカルユーザとしてログインするために使用されるキーで、パスワードは不要です。



注:CESプロキシへのSSHアクセスを設定し、アプライアンスのローカルユーザのSSHキーを設定するには、設定ガイドを参照してください。

エクスポートファイルの確認

エクスポートスクリプトが実行されると、スクリプトの内容を確認でき、アプライアンスの aliasconfig CLI コマンドで提示された元のスクリプトと同じ情報がスクリプトに含まれていることを確認できます。

```
$ pwd
/full/path/folder/Downloads
$ ls
filename.csv
$ cat filename.csv
# File exported by the CLI at 20250702T125347
test: test@example.com, test@example2.com, test@example3.com
test2: test@domain.com, test@domain2.com, test@domain3.com
```

Bashスクリプトを使用した別名テーブルのインポート

現在のエイリアスファイルをエクスポートしたら、そのファイルを変更して必要なエントリを追加し、ESAのエイリアス設定にインポートできます。

インポートbashスクリプトは、次のように構成されています。

```
#!/bin/bash

# Configuration of SSH keys and paths
SSH_KEY="/full/path/folder/.ssh/id_rsa"
LOCAL_PORT="2200"
REMOTE_USER="local_server_user"
LOCAL_FILE="/full/path/folder/Downloads/new-filename.csv"
OUTPUT_DIR="/full/path/folder/Downloads"

# Get the current local date in the desired format
CURRENT_DATE=$(date +"%Y-%m-%d_%H-%M-%S")

# 1. Upload the new aliasconfig file
echo "Uploading new aliasconfig file to the remote system..."
scp -i "$SSH_KEY" -P "$LOCAL_PORT" -O "$LOCAL_FILE" "$REMOTE_USER"@127.0.0.1:/configuration 2>/dev/null
if [ $? -ne 0 ]; then
    echo "Error: Failed to upload the aliasconfig file."
    exit 1
fi
echo "Aliasconfig file successfully uploaded."

# Pause for 5 seconds before proceeding
sleep 5

# 2. Import the new aliasconfig file and commit with a comment
COMMIT_COMMENT="Importing new entries to aliasconfig - $CURRENT_DATE"
echo "Importing the new aliasconfig file and committing changes..."
ssh -i "$SSH_KEY" "$REMOTE_USER"@127.0.0.1 -p "$LOCAL_PORT" "clustermode cluster; aliasconfig import new"
if [ $? -ne 0 ]; then
```

```

        echo "Error: Failed to import the aliasconfig file or commit changes."
        exit 1
    fi
    echo "Aliasconfig file successfully imported and committed with comment: '$COMMIT_COMMENT'."

    # Pause for 5 seconds before proceeding
    sleep 5

    # 3. Print the current aliasconfig and save it to a new file
    OUTPUT_FILE="${OUTPUT_DIR}/current-aliasconfig-${CURRENT_DATE}.txt"
    echo "Printing current aliasconfig and saving it to: $OUTPUT_FILE..."
    ssh -i "$SSH_KEY" "$REMOTE_USER"@127.0.0.1 -p "$LOCAL_PORT" 'clustermode cluster; aliasconfig print' >
    if [ $? -ne 0 ]; then
        echo "Error: Failed to print the current aliasconfig."
        exit 1
    fi
    echo "Current aliasconfig successfully saved to: $OUTPUT_FILE"

    # Finalizing the script
    echo "Process completed successfully."
    exit 0

```

説明

1.- SSHパスとキー設定

- SSH_KEY：リモートサーバに対する安全な認証に使用されるSSH秘密キーファイルへのパス。
- LOCAL_PORT:SSHトンネルに指定されたローカルポート。
- REMOTE_USER：リモート・サーバ上の認証用のユーザー・アカウント。
- LOCAL_FILE：インポートされるaliasconfig CSVファイルへのローカルパス。
- OUTPUT_DIR：インポート処理の後に現在の構成のコピーが保存されるローカルフォルダ。

2. – 現在の日付と時刻を取得する

```
CURRENT_DATE=$(date +%Y-%m-%d_%H-%M-%S")
```

- 目的:
 - ファイル名とコメントで使用するために、現在の日付と時刻を特定の形式で保存します。
- アップデート:
 - タイムスタンプによるログとバックアップの追跡と編成を容易にします。

3. – 新しいaliasconfigファイルをリモートESAサーバにアップロードする

aliasconfigファイルの新しい内容は次のとおりです。


```
# File exported by the CLI at 20250709T112719
test: new-date@example.com, new-date@example2.com, new-date@example3.com
test2: new-date@domain.com, new-date@domain2.com, new-date@domain3.com
```

指示に従って、ファイルをアップロードします。

```
echo "Uploading new aliasconfig file to the remote system..."
scp -i "$SSH_KEY" -P "$LOCAL_PORT" -O "$LOCAL_FILE" "$REMOTE_USER"@127.0.0.1:/configuration 2>/dev/null
if [ $? -ne 0 ]; then
    echo "Error: Failed to upload the aliasconfig file."
    exit 1
fi
echo "Aliasconfig file successfully uploaded."
```

- 目的:
 - SCP over SSHを使用して、ローカルマシンからリモートサーバの/configurationディレクトリにaliasconfig CSVファイルを転送します。
- アップデート:
 - アップロードが失敗すると、スクリプトはエラーを表示し、インポートが不完全にならないように停止します。

4. – 新しいaliasconfigファイルをインポートし、コメントを付けてコミットする

```
COMMIT_COMMENT="Importing new entries to aliasconfig - $CURRENT_DATE"
echo "Importing the new aliasconfig file and committing changes..."
ssh -i "$SSH_KEY" "$REMOTE_USER"@127.0.0.1 -p "$LOCAL_PORT" "clustermode cluster; aliasconfig import new"
if [ $? -ne 0 ]; then
    echo "Error: Failed to import the aliasconfig file or commit changes."
    exit 1
fi
echo "Aliasconfig file successfully imported and committed with comment: '$COMMIT_COMMENT'."
```

- 目的:
 - SSH経由で接続し、アップロードされたCSVファイルをエイリアス設定にインポートし、追跡用のタイムスタンプ付きコメントで変更を確定します。
- アップデート:
 - インポートまたはコミットが失敗すると、エラーメッセージが表示され、設定の一貫性を維持するためにスクリプトが終了します。

5. – 現在のaliasconfigを印刷し、新しいローカルファイルに保存する

```
OUTPUT_FILE="${OUTPUT_DIR}/current-aliasconfig-${CURRENT_DATE}.txt"
```

```
echo "Printing current aliasconfig and saving it to: $OUTPUT_FILE..."
ssh -i "$SSH_KEY" "$REMOTE_USER"@127.0.0.1 -p "$LOCAL_PORT" 'clustermode cluster; aliasconfig print' >
if [ $? -ne 0 ]; then
    echo "Error: Failed to print the current aliasconfig."
    exit 1
fi
echo "Current aliasconfig successfully saved to: $OUTPUT_FILE"
```

- 目的:
 - SSH経由で接続し、現在のエイリアス設定を印刷し、バックアップと監査のためにタイムスタンプ付きのローカルファイルに出力を保存します。
- アップデート:
 - 操作が失敗した場合、スクリプトはエラーを表示し、不完全な結果を表示しないように停止します。

変更の確認

スクリプトが実行されると、エイリアス設定テーブルの変更を確認し、システムログでコミットを確認できます。

新しいaliasconfigテーブルエントリの確認

新しい変更がテーブルに適用されました。

```
(Machine esa1.xyz.iphmx.com) (SERVICE)> clustermode cluster; aliasconfig print
```

```
test: new-data@example.com, new-data@example2.com, new-data@example3.com
test2: new-data@domain.com, new-data@domain2.com, new-data@domain3.com
```

コミット変更の確認

このコマンドを使用すると、変更を行ったユーザとその発生日を含め、ESAにコミットされた変更を追跡および確認できます。

```
(Machine esa1.xyz-66.iphmx.com) (SERVICE)> grep "commit" system_logs
Wed Jul 9 11:29:42 2025 Info: PID 95790: User local_server_user commit changes: Importing new entries
```

スクリプトの柔軟性

このスクリプトは現在Bashで記述されていますが、他のスクリプト言語やプログラミング言語 (Python、PowerShell、Perlなど) で簡単に調整または書き換えでき、異なる管理者や環境の好みや要件に合わせることができます。この柔軟性により、お客様の運用要件に最も適した言語や

ツールを活用しながら、コアロジックとワークフローを維持できます。

最後の考察

このインポート/エクスポートスクリプトは、アプライアンス上でエイリアス設定を直接管理するための実用的で効率的なソリューションを提供します。コンフィギュレーションファイルのアップロード、インポート、およびバックアップを自動化することで、管理者は手動による介入なしに安全かつ確実に変更を導入できます。このスクリプトはプロセスを合理化するだけでなく、タイムスタンプ付きのバックアップとコミットコメントを通じてトレーサビリティを確保します。

また、このようなスクリプトを使用すると、特に複数の変更や一括更新が必要な場合に、環境内の一貫性とコンプライアンスを維持するのに役立ちます。現在の構成を定期的にバックアップすることで、セキュリティの層が追加され、必要に応じて迅速なリカバリやロールバックが可能になります。

全体として、このアプローチにより、チームは設定のアップデートをより確実に、コントロールして、効率よく管理できるようになります。今後の調整のために、このスクリプトを他のタイプのコンフィギュレーションファイルを処理するように簡単に調整したり、必要に応じて追加のメンテナンスタスクをさらに自動化したりできます。

参照リンク

- [クラウドEメールセキュリティ\(CES\)ソリューションのコマンドラインインターフェイス \(CLI\)へのアクセス](#)
- [CLIの手順：PuTTY \[Windows/PCユーザ\]](#)
- [パスワードなしで ESA にログインできる SSH 公開キー認証の設定方法](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。