

# Automatiser l'importation et l'exportation de la configuration des alias

## Table des matières

---

[Introduction](#)

[Conditions préalables](#)

[Exigences](#)

[Composants utilisés](#)

[Informations générales](#)

[Exportation et importation d'une table d'alias](#)

[Exportation de la table d'alias avec le script Bash](#)

[Explication](#)

[1.- Configuration des clés et des chemins SSH](#)

[2.- Connexion au proxy et configuration du tunnel SSH](#)

[3.- Pause pendant 5 secondes avant de continuer](#)

[4.- Exporter le fichier aliasconfig à partir du système distant](#)

[5.- Télécharger le fichier dans le répertoire local](#)

[Finalisation du script](#)

[Clés SSH](#)

[Vérification du fichier exporté](#)

[Importation de table d'alias avec un script Bash](#)

[Explication](#)

[1.- Configuration du chemin SSH et de la clé](#)

[2.-Obtenir la date et l'heure actuelles](#)

[3.- Téléchargez le nouveau fichier aliasconfig sur le serveur ESA distant](#)

[4.- Importer le nouveau fichier aliasconfig et valider avec un commentaire](#)

[5.-Imprimez l'alias actuel config et enregistrez-le dans un nouveau fichier local](#)

[Vérification des modifications](#)

[Vérification des nouvelles entrées de table aliasconfig](#)

[Vérifier les modifications validées](#)

[Flexibilité des scripts](#)

[Réflexions finales](#)

[Liens de référence](#)

---

## Introduction

Ce document décrit les étapes pour automatiser les tâches d'importation et d'exportation de la configuration d'alias dans l'appliance de sécurité de la messagerie.

# Conditions préalables

## Exigences

Cisco recommande de connaître les sujets suivants :

- Cisco Secure Email Gateway (SEG / ESA) AsyncOS 16.0.2
- Accès de l'interface de ligne de commande au cloud Appliance
- CLI Linux
- Script Shell

## Composants utilisés

Les informations contenues dans ce document sont basées sur les logiciels suivants :

- Appliance de sécurisation de la messagerie cloud (CESA)
- Bousculade

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

## Informations générales

L'objectif est d'automatiser certaines tâches, mais certains processus nécessitent généralement une intervention manuelle. Cependant, dans le cas de l'importation et de l'exportation de la configuration d'alias actuelle, ces tâches peuvent être entièrement automatisées, ce qui élimine le besoin de saisie manuelle.

## Exportation et importation d'une table d'alias

Pour importer une table d'alias, commencez par vérifier l'accès SSH et SCP afin de vous assurer que vous pouvez vous connecter à la passerelle de messagerie.

Avant de continuer, une table d'alias doit exister au sein de l'appliance :

```
(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)>
```

Lorsque vous utilisez la sous-commande export de la commande aliasconfig pour sauvegarder une table d'alias existante, un fichier (portant un nom que vous spécifiez) est généré et enregistré

dans le répertoire /configuration pour l'écouteur.

## Exportation de la table d'alias avec le script Bash

Dans ce cas, un script bash se connecte à votre appliance CES et procède à l'exportation du fichier d'alias

Le script bash est structuré comme suit :

```
#!/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
```

## Explication

### 1.- Configuration des clés et des chemins SSH

- PROXY\_KEY et SECOND\_KEY : Chemin d'accès complet au fichier de clé privée SSH utilisé pour l'authentification. Dans ce cas, les deux clés sont définies sur le même chemin.
- Exemple : /full/path/folder/.ssh/id\_rsa
- PORT\_LOCAL : Spécifie le port local (2200) pour le tunnel SSH.
- UTILISATEUR\_PROXY : Nom d'utilisateur utilisé pour la connexion au serveur proxy.
- HÔTE\_PROXY : Nom d'hôte du serveur proxy.
- HÔTE\_CIBLE : Nom de domaine complet (FQDN) de l'hôte cible, mis à jour vers esa1.xyz.iphmx.com.
- UTILISATEUR\_DISTANT : Nom d'utilisateur utilisé pour la connexion à l'appliance distante via le tunnel SSH.
- FICHER\_DISTANT : Chemin d'accès au système distant où le fichier exporté est stocké (/configuration/filename.csv).
- RÉP\_LOCAL : Le répertoire local d'enregistrement du fichier est /full/path/folder/Downloads.
- CHEMIN\_FICHER\_LOCAL : Chemin d'accès local complet du fichier téléchargé, mis à jour en \${LOCAL\_DIR}/aliasconfig-file.csv.

### 2.- Connexion au proxy et configuration du tunnel SSH

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

- Objectif:
  - Configure un tunnel SSH vers le serveur proxy pour une communication sécurisée avec l'hôte cible.
- Mises à jour :
  - La clé privée SSH se trouve désormais dans /full/path/folder/.ssh/id\_rsa.
  - Le nom d'hôte cible a été mis à jour vers esa1.xyz.iphmx.com.
- Gestion des erreurs :
  - Si la connexion échoue, un message d'erreur s'affiche et le script se ferme avec un code d'erreur (exit 1).

### 3.- Pause pendant 5 secondes avant de continuer

- Objectif:
  - Introduit un délai pour s'assurer que le tunnel SSH est complètement établi avant de passer à l'étape suivante.

#### 4.- Exporter le fichier aliasconfig à partir du système distant

```
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
```

- Objectif:
  - Se connecte à l'hôte cible via le tunnel SSH et exporte la configuration d'alias vers un fichier nommé aliasconfig-file.csv.
- Mises à jour :
  - Le nom du fichier exporté a été mis à jour en aliasconfig-file.csv.
- Redirection de sortie :
  - 2>/dev/null supprime tous les messages d'erreur de la commande SSH.
- Gestion des erreurs :
  - Si l'exportation échoue, un message d'erreur s'affiche et le script se ferme.

#### 5.- Télécharger le fichier dans le répertoire local

```
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
```

- Objectif:
  - Utilise scp pour copier en toute sécurité le fichier exporté du système distant vers le répertoire local.
- Mises à jour :
  - Le fichier local est enregistré sous le nom aliasconfig-file.csv dans le répertoire /full/path/folder/Downloads.
- Gestion des erreurs :
  - Si le téléchargement du fichier échoue, un message d'erreur s'affiche et le script se ferme.

#### Finalisation du script

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

- Objectif:

- Génère un message de réussite et quitte le script avec un code de réussite (exit 0), indiquant que toutes les opérations ont été effectuées avec succès.

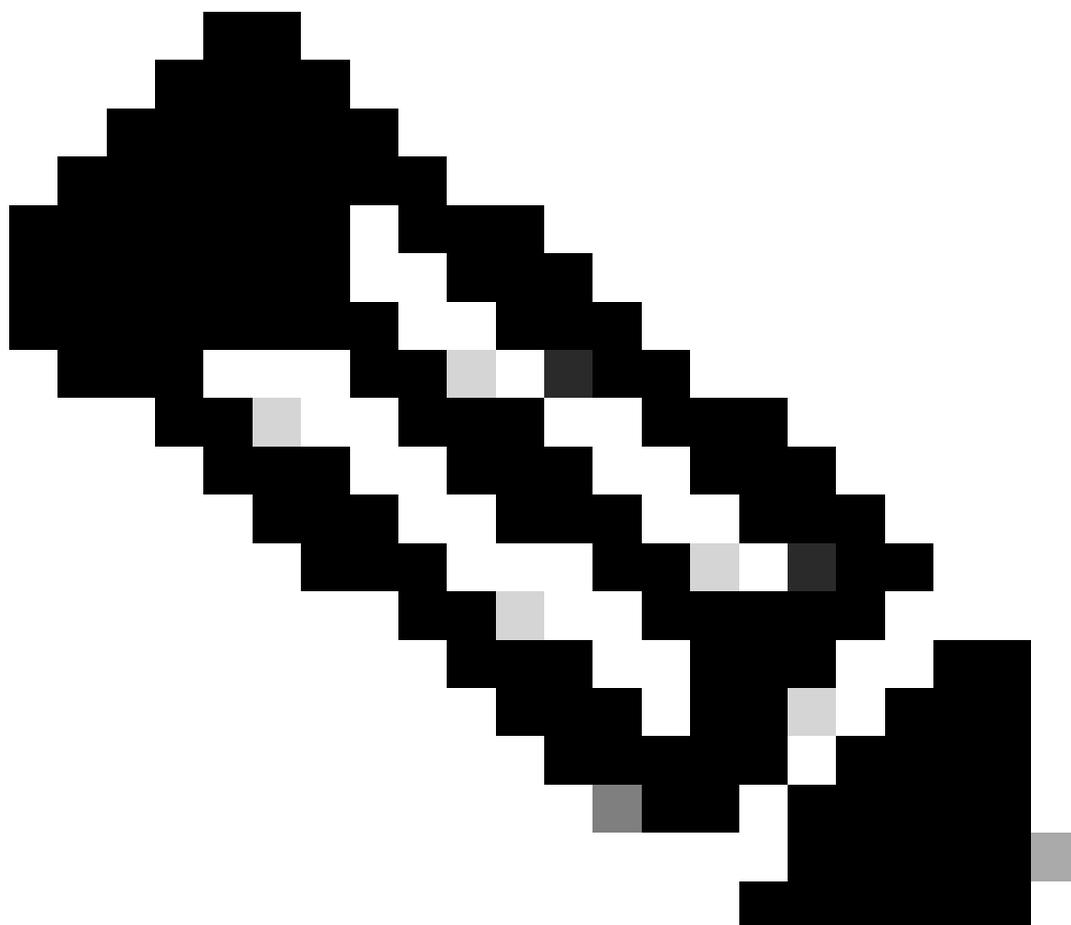
## Clés SSH

Vous pouvez noter dans le script deux variables `PROXY_KEY` et `HOST_KEY`. Ces clés peuvent être identiques ou différentes.

La `CLÉ_PROXY` est utilisée pour se connecter au cloud proxy qui est obligatoire pour accéder à vos serveurs CESA.

`HOST_KEY` est la clé utilisée pour se connecter en tant qu'utilisateur local, ce qui élimine le besoin d'un mot de passe.

---



Remarque : Pour configurer l'accès SSH au proxy CES et configurer une clé SSH pour un utilisateur local sur l'appliance, consultez le guide de configuration.

---

## Vérification du fichier exporté

Une fois le script d'exportation exécuté, vous pouvez vérifier son contenu et vous pouvez constater qu'il contient les mêmes informations que l'original présenté dans la commande CLI `aliasconfig` de l'appliance.

```
$ 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
```

## Importation de table d'alias avec un script Bash

Une fois que vous avez exporté le fichier d'alias actuel, vous pouvez le modifier, ajouter les entrées nécessaires, puis l'importer dans la configuration ESA `alia`.

Le script d'importation bash est structuré comme suit :

```
#!/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

```

## Explication

### 1.- Configuration du chemin SSH et de la clé

- CLÉ\_SSH : Chemin d'accès au fichier de clé privée SSH, utilisé pour l'authentification sécurisée sur le serveur distant.
- PORT\_LOCAL : Port local désigné pour le tunnel SSH.
- UTILISATEUR\_DISTANT : Compte utilisateur pour l'authentification sur le serveur distant.
- FICHER\_LOCAL : Chemin local vers le fichier CSV aliasconfig à importer.
- RÉP\_SORTIE : Dossier local dans lequel une copie de la configuration actuelle est enregistrée après le processus d'importation.

### 2.- Obtenir la date et l'heure actuelles

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

- Objectif:
  - Stocker la date et l'heure actuelles dans un format spécifique pour une utilisation dans les noms de fichiers et les commentaires.
- Mises à jour :
  - Facilite le suivi et l'organisation des journaux et des sauvegardes par horodatage.

### 3.- Téléchargez le nouveau fichier aliasconfig sur le serveur ESA distant

Voici le nouveau contenu du fichier aliasconfig :

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

Suivez les instructions pour télécharger le fichier :

```
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."
```

- Objectif:
  - Transférez le fichier CSV aliasconfig de la machine locale au répertoire /configuration sur le serveur distant en utilisant SCP sur SSH.
- Mises à jour :
  - Si le téléchargement échoue, le script affiche une erreur et s'arrête pour empêcher les importations incomplètes.

#### 4.- Importer le nouveau fichier aliasconfig et valider avec un commentaire

```
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'."
```

- Objectif:
  - Connectez-vous via SSH, importez le fichier CSV téléchargé dans la configuration d'alias et validez les modifications avec un commentaire horodaté pour le suivi.
- Mises à jour :
  - Si l'importation ou la validation échoue, un message d'erreur s'affiche et le script se ferme pour maintenir la cohérence de la configuration.

#### 5.- Imprimez l'aliasconfig actuel et enregistrez-le dans un nouveau fichier local

```
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"
```

- Objectif:
  - Connectez-vous via SSH, imprimez la configuration d'alias actuelle et enregistrez le résultat dans un fichier local horodaté à des fins de sauvegarde et d'audit.
- Mises à jour :
  - Si l'opération échoue, le script affiche une erreur et s'arrête pour éviter de donner des résultats incomplets.

## Vérification des modifications

Une fois le script exécuté, vous pouvez vérifier les modifications dans la table de configuration des alias et vérifier les validations dans les journaux système.

### Vérification des nouvelles entrées de table aliasconfig

De nouvelles modifications ont été appliquées à la table.

```
(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
```

### Vérifier les modifications validées

Cette commande vous permet de suivre et d'examiner les modifications validées pour l'ESA, y compris l'utilisateur qui a effectué la modification et la date à laquelle elle s'est produite.

```
(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
```

## Flexibilité des scripts

Bien que ce script soit actuellement écrit en bash, il peut être facilement adapté ou réécrit dans

d'autres langages de script ou de programmation, tels que Python, PowerShell ou Perl, pour mieux s'aligner avec les préférences ou les exigences de différents administrateurs et environnements. Cette flexibilité permet de conserver la logique et le workflow de base tout en exploitant le langage ou les outils les plus adaptés à vos besoins opérationnels.

## Réflexions finales

Ce script d'importation/exportation offre une solution pratique et efficace pour gérer les configurations d'alias directement sur l'appliance. En automatisant le téléchargement, l'importation et la sauvegarde des fichiers de configuration, les administrateurs peuvent introduire des modifications en toute sécurité et fiabilité, sans intervention manuelle. Non seulement le script rationalise le processus, mais il assure également la traçabilité grâce à des sauvegardes horodatées et des commentaires de validation.

En outre, un tel script permet de maintenir la cohérence et la conformité dans votre environnement, en particulier lorsque plusieurs modifications ou mises à jour en masse sont nécessaires. Les sauvegardes régulières de la configuration actuelle offrent une couche de sécurité supplémentaire, permettant une récupération ou une restauration rapide si nécessaire.

Dans l'ensemble, cette approche permet aux équipes de gérer les mises à jour de configuration avec plus de confiance, de contrôle et d'efficacité. Pour tout ajustement futur, le script peut être facilement adapté pour gérer d'autres types de fichiers de configuration ou pour automatiser davantage les tâches de maintenance supplémentaires, le cas échéant.

## Liens de référence

- [Accès à l'interface de ligne de commande \(CLI\) de votre solution cloud de sécurisation de la messagerie électronique \(CES\)](#)
- [Instructions CLI : PuTTY \[Utilisateurs Windows/PC\]](#)
- [Comment configurer l'authentification par clé publique SSH pour la connexion à l'ESA sans mot de passe](#)

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.