



## Formats de mise à disposition

---

- [Profils de configuration](#) , à la page 1
- [Formats de profil de configuration](#), à la page 1
- [Chiffrement et compression de profil ouvert \(XML\)](#), à la page 5
- [Application d'un profil au téléphone](#), à la page 11
- [Types de paramètres de mise à disposition](#), à la page 12
- [Types de données](#), à la page 19
- [Mises à jour de profil et mises à niveau du micrologiciel](#), à la page 22

## Profils de configuration

Le téléphone accepte la configuration au format XML.

Les exemples de ce document utilisent les profils de configuration avec une syntaxe de format (XML) XML.

Pour des informations détaillées sur votre téléphone, reportez-vous au guide d'administration de votre appareil spécifique. Chaque guide décrit les paramètres pouvant être configurés par le serveur Web d'administration.

## Formats de profil de configuration

Le profil de configuration définit les valeurs des paramètres du téléphone.

Le format XML de profil de configuration utilise les outils de création standard XML pour compiler les paramètres et les valeurs.



---

**Remarque** Seul le jeu de caractères UTF-8 est pris en charge. Si vous modifiez le profil dans un éditeur, ne modifiez pas le format de codage ; dans le cas contraire, le téléphone ne reconnaît pas le fichier.

---

Chaque téléphone comporte un ensemble de fonctionnalités différentes et, par conséquent, un jeu différent de paramètres.

### **Profil de Format XML (XML)**

Le profil de format ouvert est un fichier texte avec une syntaxe similaire à XML dans une hiérarchie d'éléments, avec des valeurs et des attributs d'élément. Ce format vous permet d'utiliser les outils standard pour créer le

fichier de configuration. Un fichier de configuration dans ce format peut être envoyé à partir du serveur de mise à disposition au téléphone lors de l'opération de resynchronisation. Le fichier peut être envoyé sans compilation comme objet binaire.

Le téléphone peut accepter les formats de configuration que génèrent des outils standard. Cette fonctionnalité facilite le développement d'un logiciel de serveur de mise à disposition de back-end qui génère des profils de configuration à partir de bases de données existantes.

Pour assurer la protection des informations confidentielles du profil de configuration, le serveur de mise à disposition envoie ce type de fichier au téléphone via un canal sécurisé par TLS. Éventuellement, le fichier peut être compressé à l'aide de l'algorithme de compression gzip (RFC1951).

Le fichier peut être chiffré à l'aide de l'une de ces méthodes de chiffrement :

- Chiffrement AES-256-CBC
- Chiffrement du contenu HTTP en fonction de RFC-8188 avec un chiffrement AES-128-GCM

### Exemple : profil de format ouvert

```
<flat-profile>
<Resync_On_Reset> Yes </Resync_On_Reset>
<Resync_Periodic> 7200 </Resync_Periodic>
<Profile_Rule> tftp://prov.telco.com:6900/cisco/config/CP_xxxx_MPP.cfg</Profile_Rule>
</flat-profile>
```

La balise element <flat-profile> englobe tous les éléments de paramètre que le téléphone reconnaît.

## Composants du fichier de configuration

Un fichier de configuration peut comporter les composants suivants :

- Balises d'élément
- Attributs
- Paramètres
- Fonctions de mise en forme
- Commentaires XML

### Propriétés de la balise élément

- Le format XML de mise à disposition et l'interface utilisateur Web permettent la configuration des mêmes paramètres. Le nom de la balise XML et les noms de champ dans l'interface utilisateur Web sont similaires, mais varient en raison des restrictions de nom d'élément XML. Par exemple, des traits de soulignement ( \_ ) au lieu de « ».
- Le téléphone reconnaît les éléments comportant des noms de paramètre corrects qui sont encapsulés dans l'élément spécial <flat-profile>.
- Les noms d'élément sont compris entre crochets.

- La plupart des noms d'éléments sont similaires aux noms de champ des pages Web d'administration du périphérique, avec les modifications suivantes :

- Les noms d'éléments ne peuvent pas comporter d'espaces ou de caractères spéciaux. Pour obtenir le nom de l'élément à partir du nom de champ d'administration web, remplacez chaque espace par un trait de soulignement ou les caractères spéciaux [ ], ( ), ou /.

**Exemple :** L'élément<Resync\_On\_Reset> représente le champ **Resync On Reset**.

- Chaque nom d'élément doit être unique. Dans les pages Web d'administration, les mêmes champs peuvent apparaître sur plusieurs pages Web, telles que les pages de ligne, d'utilisateur et de numéro de poste. Ajouter [n] au nom de l'élément pour indiquer le numéro affiché sur l'onglet de la page.

**Exemple :** L'élément<Dial\_Plan\_1\_> représente le plan de **numérotation** de la ligne 1.

- À chaque balise d'ouverture de l'élément doit correspondre une balise de fermeture. Par exemple :

```
<flat-profile>
<Resync_On_Reset> Yes
  </Resync_On_Reset>
<Resync_Periodic> 7200
  </Resync_Periodic>
<Profile_Rule>tftp://prov.telco.com: 6900/cisco/config/CP_xxxx_MPP.cfg
  </Profile_Rule>
</flat-profile>
```

- Les balises d'éléments respectent la casse.
- Les balises d'élément vide sont autorisées et seront interprétées comme configurant la valeur à vide. Saisissez la balise d'ouverture de l'élément sans la balise d'élément correspondante et insérer un espace et une barre oblique avant le signe de fermeture (>). Dans cet exemple, la règle du profil B est vide :

```
<Profile_Rule_B />
```

- Une balise d'élément vide est utilisable pour empêcher le remplacement de toute valeur fournie par l'utilisateur lors de l'opération de resynchronisation. Dans l'exemple suivant, les paramètres de numérotation abrégée de l'utilisateur restent inchangés :

```
<flat-profile>
<Speed_Dial_2_Name ua="rw"/>
<Speed_Dial_2_Number ua="rw"/>
<Speed_Dial_3_Name ua="rw"/>
<Speed_Dial_3_Number ua="rw"/>
<Speed_Dial_4_Name ua="rw"/>
<Speed_Dial_4_Number ua="rw"/>
<Speed_Dial_5_Name ua="rw"/>
<Speed_Dial_5_Number ua="rw"/>
<Speed_Dial_6_Name ua="rw"/>
<Speed_Dial_6_Number ua="rw"/>
<Speed_Dial_7_Name ua="rw"/>
<Speed_Dial_7_Number ua="rw"/>
<Speed_Dial_8_Name ua="rw"/>
<Speed_Dial_8_Number ua="rw"/>
<Speed_Dial_9_Name ua="rw"/>
<Speed_Dial_9_Number ua="rw"/>
</flat-profile>
```

- Utilisez une valeur vide pour définir le paramètre correspondant à une valeur de chaîne vide. Saisissez un élément d'ouverture et de fermeture sans aucune valeur entre eux. Dans l'exemple suivant, le paramètre GPP\_A est défini à une chaîne vide.

```
<flat-profile>
<GPP_A>
  </GPP_A>
</flat-profile>
```

- Les noms d'élément non reconnus sont ignorés.

## Propriétés des paramètres

Ces propriétés s'appliquent aux paramètres :

- Tous les paramètres qu'aucun profil ne précise demeurent inchangés dans le téléphone.
- Les paramètres non reconnus sont ignorés.
- Si le profil de format ouvert contient plusieurs occurrences de la même balise de paramètre, la dernière de ces occurrences a priorité sur les plus anciennes. Pour éviter d'écraser par inadvertance les valeurs de configuration d'un paramètre, il est recommandé que chaque profil précise au maximum une instance d'un paramètre.
- Le dernier profil traité est prioritaire. Si plusieurs profils spécifient le même paramètre de configuration, la valeur du profil le plus récent est prioritaire.

## Formats de la chaîne

Ces propriétés s'appliquent au formatage des chaînes :

- Les commentaires sont autorisés via la syntaxe XML standard.
 

```
<!-- My comment is typed here -->
```
- Les espaces blancs et de fin sont autorisés pour améliorer la lisibilité, mais sont supprimés de la valeur du paramètre.
- Les nouvelles lignes au sein d'une valeur sont converties en espaces.
- Un en-tête XML de la forme `<? ?>` est autorisé, mais le téléphone ne le prend pas en compte.
- Pour saisir des caractères spéciaux, utilisez des caractères XML d'échappement de base, comme indiqué dans le tableau suivant.

Caractère spécial	Séquence d'échappement XML
& (et commercial)	&amp;
< (inférieur à)	&lt;
> (supérieur à)	&gt;
' (apostrophe)	&apos;
« (double guillemet)	&quot;

Dans l'exemple suivant, les caractères d'échappement sont saisis pour représenter les symboles supérieur à et inférieur à nécessaires dans une règle de plan de numérotation. Cet exemple définit un plan de numérotation de la hotline d'information qui définit le paramètre `<Dial_Plan_1_>` (**Admin Login > advanced > Voice > Ext (n)**) égal à (S0 <:18005551212>).

```
<flat-profile>
  <Dial_Plan_1_>
    (S0 &lt;;:18005551212&gt;)
  </Dial_Plan_1_>
</flat-profile>
```

- Les caractères numériques d'échappement, utilisent des valeurs décimales et hexadécimales (C'est-à-dire `&#40;` et `&#x2e;`), sont traduits.
- Le micrologiciel du téléphone ne prend en charge que des caractères ASCII.

## Chiffrement et compression de profil ouvert (XML)

Le profil de configuration ouvert peut être compressé pour réduire la charge du réseau sur le serveur de configuration. Le profil peut également être chiffré pour assurer la protection des informations confidentielles. La compression n'est pas obligatoire, mais elle doit précéder le chiffrement.

### Compression de profil ouvert

La méthode de compression prise en charge est l'algorithme de compression gzip (RFC1951). L'utilitaire gzip et la bibliothèque de compression qui mettent en œuvre le même algorithme (zlib) sont disponibles à partir de sites Internet.

Pour identifier la compression, le téléphone prévoit que le fichier compressé contienne un en-tête compatible gzip. L'appel de l'utilitaire gzip sur le profil ouvert d'origine génère l'en-tête. Le téléphone contrôle l'en-tête du fichier téléchargé pour déterminer le format de fichier.

Par exemple, si `profile.xml` est un profil valide, le fichier `profile.xml.gz` est également accepté. Une des commandes suivantes peut générer ce type de profil :

- `>gzip profile.xml`

Remplace le fichier d'origine par le fichier compressé.

- `>cat profile.xml | gzip > profile.xml.gz`

Laisse le fichier d'origine en place, génère un nouveau fichier compressé.

Un didacticiel sur la compression est fourni à la section [Compresser un profil ouvert avec Gzip](#).

### Chiffrement de profil ouvert

Le chiffrement par clé symétrique est utilisable pour chiffrer un profil de configuration ouvert, que le fichier soit compressé ou non. La compression, si elle est utilisée, doit être appliquée avant le chiffrement.

Le serveur de configuration utilise le protocole HTTPS pour traiter la mise à disposition initiale du téléphone après le déploiement. Le pré-chiffrement des profils de configuration hors connexion permet l'utilisation de HTTP pour synchroniser les profils. Cette fonctionnalité réduit la charge sur le serveur HTTPS dans les déploiements à grande échelle.

Le téléphone prend en charge deux méthodes de chiffrement pour les fichiers de configuration :

- Chiffrement AES-256-CBC

- Chiffrement du contenu HTTP en fonction de RFC-8188 avec un chiffrement AES-128-GCM

La clé ou information de saisie de clé (IKM) doit être mise à disposition par avance dans l'unité à une heure antérieure. Le démarrage de la clé secrète peut être réalisé en toute sécurité à l'aide de HTTPS.

Le nom de fichier final ne nécessite pas un format spécifique, mais un nom de fichier se terminant par l'extension `.cfg` indique normalement un profil de configuration.

## Chiffrement AES-256-CBC

Le téléphone prend en charge le chiffrement AES-256-CBC pour les fichiers de configuration.

L'outil de chiffrement OpenSSL, disponible en téléchargement à partir de différents sites Internet, peut effectuer le chiffrement. La prise en charge pour le chiffrement AES 256 bits peut nécessiter la recompilation de l'outil pour activer le code AES. Le micrologiciel a été testé par rapport à la version openssl-1.1.1d.

[Chiffrer un profil avec OpenSSL](#) propose un didacticiel sur le chiffrement.

Pour un fichier chiffré, le profil prévoit que le fichier aura le format généré par la commande suivante :

```
# example encryption key = SecretPhrase1234
openssl enc -e -aes-256-cbc -k SecretPhrase1234 -in profile.xml -out profile.cfg

# analogous invocation for a compressed xml file
openssl enc -e -aes-256-cbc -k SecretPhrase1234 -in profile.xml.gz -out profile.cfg
```

Un `-k` minuscule précède la clé secrète, qui peut être n'importe quelle phrase en texte non chiffré, et qui est utilisée pour générer une racine 64 bits aléatoire. Avec le mot de passe spécifié par l'argument `-k`, l'outil de chiffrement dérive un vecteur initial aléatoire 128 bits et la clé de chiffrement 256 bits réelle.

Lorsque cette forme de chiffrement est utilisée sur un profil de configuration, le téléphone doit connaître la valeur de la clé secrète pour déchiffrer le fichier. Cette valeur est spécifiée comme un identificateur dans l'URL du profil. La syntaxe est la suivante, à l'aide d'une URL explicite :

```
[--key "SecretPhrase1234"] http://prov.telco.com/path/profile.cfg
```

Cette valeur est programmée en appliquant l'un des paramètres `Profile_Rule`.

### Extension macro

Plusieurs paramètres de mise à disposition font l'objet d'une expansion de macro interne avant d'être évalués. Cette étape de pré-évaluation donne une plus grande souplesse de contrôle des activités de resynchronisation et de mise à niveau du téléphone.

Ces groupes de paramètres font l'objet d'expansion de macro avant l'évaluation :

- `Resync_Trigger_*`
- `Profile_Rule*`
- `Log_xxx_Msg`
- `Upgrade_Rule`

Dans certaines conditions, certains paramètres généraux (GPP\_\*) sont soumis à une expansion de macro, comme explicitement indiqué en [Arguments de resynchronisation facultatifs](#), à la page 10.

Au cours de l'expansion de macro, le contenu des variables nommées remplace des expressions de la forme \$NAME et \$(NAME). Ces variables incluent des paramètres généraux, plusieurs identificateurs de produit, certains minuteurs d'événement et les valeurs d'état de mise à disposition. Pour en obtenir la liste complète, consultez [Variables d'expansion de macro](#).

Dans l'exemple suivant, l'expression \$(MAU) est utilisée pour insérer l'adresse MAC 000E08012345.

L'administrateur saisit : **\$ (MAU) config.cfg**

L'expansion de macro résultante pour un périphérique ayant l'adresse MAC 000E08012345 est :  
000E08012345config.cfg

Si un nom de macro n'est pas reconnu, il demeure non étendu. Par exemple, le nom STRANGE n'est pas reconnu comme nom de macro valide, tandis que MAU est considéré comme un nom de macro valide.

L'administrateur saisit : **\$STRANGE\$MAU.cfg**

L'expansion de macro résultante pour un périphérique ayant l'adresse MAC 000E08012345 est :  
\$STRANGE000E08012345.cfg

L'expansion de macro n'est pas appliquée de manière récursive. Par exemple, \$\$MAU" est développé en \$MAU" (le \$\$ est développé) et ne se traduit pas par l'adresse MAC.

Le contenu des paramètres à usages spéciaux, GPP\_SA à GPP\_SD, est mis en correspondance avec les expressions macro \$SA à \$SD. Ces paramètres subissent uniquement une expansion de macro en tant qu'argument des options de la **--key**, **--uid** et **--pwd** dans une URL de resynchronisation.

## Expressions conditionnelles

Les expressions conditionnelles peuvent déclencher des événements de resynchronisation à partir de l'URL de remplacement dans le cas des opérations de resynchronisation et de mise à niveau.

Les expressions conditionnelles se composent d'une liste de comparaisons, séparées par l'opérateur **et**. Toutes les comparaisons doivent être satisfaites pour que la condition soit vraie.

Chaque comparaison peut se rapporter à un des trois types d'opérandes suivants :

- Valeurs entières
- Numéros de version de logiciel ou de matériel
- Chaînes entre guillemets doubles

### Numéros de version

La version logicielle des téléphones IP Cisco avec micrologiciel multiplateformes utilise ce format (où *BN* est le numéro de version) :

- Pour la version 11.3(1)SR1 du micrologiciel et les versions antérieures : sipyyyy.*11-0-IMPP-376*.  
Où *aaaa* indique le modèle de téléphone ou la série de téléphones ; *11* est la version principale ; *0* correspond à la version mineure ; *IMPP* est la micro version, et *376* est le numéro de version.
- Pour la version 11.3(2) du firmware et les versions ultérieures : sipyyyy.*11-3-2MPP0001-609*  
Où *aaaa* indique le modèle de téléphone ou la série de téléphones ; *11* est la version principale ; *3* correspond à la version mineure ; *2MPP0001* est la micro version, et *609* est le numéro de version.

La chaîne de comparaison doit utiliser le même format. Dans le cas contraire, une erreur d'analyse de format se produit.

Lors de la comparaison de la version du logiciel, la version majeure, la version mineure et la micro version sont comparées en séquence, et les chiffres les plus à gauche ont la priorité sur ces dernières. Lorsque les numéros de version sont identiques, le numéro de build est comparé.

### Exemples de numéros de version valides

- Pour le micrologiciel version 11.3 (1) SR1 et versions antérieures :

```
sip78xx.11-0-1MPP-312
```

- Pour la version du micrologiciel 11.3 (2) et versions ultérieures :

```
sip78xx.11-3-2MPP0001-609
```

### Comparaison

- Pour le micrologiciel version 11.3 (1) SR1 et versions antérieures :

```
sipyyyy.11-3-1MPP-110 > sipyyyy.11-2-3MPP-256
```

- Pour la version du micrologiciel 11.3 (2) et versions ultérieures :

```
sipyyyy.11-3-2MPP0002-256 > sipyyyy.11-3-2MPP0001-609
```

Les chaînes entre guillemets peuvent être comparées pour leur égalité ou leur inégalité. Les numéros de version et les nombres entiers peuvent également être comparés. Les opérateurs de comparaison peuvent être exprimés sous forme de symboles ou d'acronymes. Les acronymes sont utiles pour l'expression de la condition dans un profil de format ouvert.

Opérateur	Autre syntaxe	Description	Applicable aux opérandes entiers et numéros de version	Applicable aux opérandes de chaînes entre guillemets
=	eq	égal à	Oui	Oui
!=	ne	non égal à	Oui	Oui
<	lt	inférieur à	Oui	Non
<=	ie	inférieur ou égal à	Oui	Non
>	gt	supérieur à	Oui	Non
>=	se	supérieur ou égal à	Oui	Non
ET		et	Oui	Oui

Il est important d'encadrer les variables de macro de guillemets là où une chaîne littérale est attendue. Ne le faites pas lorsqu'un numéro ou un numéro de version sont attendus.



Lorsqu'elles sont utilisées dans le cadre des paramètres `Profile_Rule*` et `Upgrade_Rule`, les expressions conditionnelles doivent être placées dans la syntaxe « (expression) ? », comme dans cet exemple de règle de mise à niveau. N'oubliez pas de remplacer la valeur *BN* avec le numéro de version du build du micrologiciel à mettre à niveau.

- Pour le micrologiciel version 11.3 (1) SR1 et versions antérieures

```
($SWVER ne sip78xx.11-0-0MPP-256)? http://ps.tell.com/sw/sip78xx.11-0-0MPP-BN.loads
```

- Pour la version du micrologiciel 11.3 (2) et versions ultérieures:

```
($SWVER ne sip78xx.11-3-2MPP0001-609)?  
http://ps.tell.com/sw/sip78xx.11-3-2MPP0001-BN.loads
```

N'utilisez pas la syntaxe précédente avec des parenthèses pour configurer les paramètres `Resync_Trigger_*`.

## Syntaxe des URL

Utilisez la syntaxe des URL standard pour indiquer comment récupérer les micrologiciels et les fichiers de configuration dans les paramètres `Profile_Rule*` et `Upgrade_Rule`, respectivement. La syntaxe est la suivante :

**Chemin [ schéma:// ] [ serveur [:port]]**

Où **schéma** prend l'une des valeurs suivantes :

- tftp
- http
- HTTPS

Si **schéma** est omis, par défaut tftp est utilisé. Le serveur peut être un nom d'hôte DNS reconnu ou une adresse IP numérique. Le port est le numéro de port TCP ou UDP de destination. Le chemin d'accès doit commencer par le répertoire racine (/) ; Il doit être un chemin d'accès absolu.

Si **serveur** est manquant, le serveur tftp spécifié via DHCP (option 66) est utilisé.




---

**Remarque** Pour les règles de mise à niveau, le serveur doit être spécifié.

---

Si **port** est manquant, le port standard pour le schéma spécifié est utilisé. Tftp utilise le port UDP 69, http utilise le port TCP 80, https utilise le port TCP 443.

Un chemin d'accès doit être présent. Il ne doit pas nécessairement faire référence à un fichier statique, mais peut indiquer un contenu dynamique obtenu au moyen de CGI.

L'expansion de macro s'applique au sein des URL. Voici des exemples d'URL valides :

```
/$MA.cfg  
/cisco/cfg.xml  
192.168.1.130/profiles/init.cfg  
tftp://prov.call.com/cpe/cisco$MA.cfg  
http://neptune.speak.net:8080/prov/$D/$E.cfg  
https://secure.me.com/profile?Linksys
```

Lorsque vous utilisez l'option DHCP 66, la syntaxe vide n'est pas prise en charge par les règles de mise à niveau. Elle ne s'applique qu'à `Profile_Rule*`.

## Chiffrement du contenu HTTP en fonction de RFC-8188

Le téléphone prend en charge le chiffrement du contenu HTTP en fonction de RFC 8188 avec un chiffrement AES-128-GCM des fichiers de configuration. Avec cette méthode de codage, toutes les entités peuvent lire les en-têtes de message HTTP. Cependant, seules les entités qui connaissent l'IKM (Input Keying Material, Informations de saisie d'entrée) peuvent lire la charge utile. Lorsque le téléphone est mis à disposition avec l'IKM, le téléphone et le serveur de mise à disposition peuvent échanger des fichiers de configuration en toute sécurité, tout en autorisant les éléments du réseau de fabricants tiers à utiliser les en-têtes de messages à des fins d'analyse et de surveillance.

Le paramètre de configuration XML **IKM\_HTTP\_Encrypt\_Content** contient l'IKM sur le téléphone. Pour des raisons de sécurité, ce paramètre n'est pas accessible sur la page Web d'administration du téléphone. Il n'est pas non plus visible dans le fichier de configuration du téléphone, auquel vous pouvez accéder à partir de l'adresse IP du téléphone ou depuis les rapports de configuration du téléphone envoyés au serveur de mise à disposition.

Si vous souhaitez utiliser le chiffrement RFC 8188, assurez-vous que vous :

- Configurez le téléphone avec IKM en spécifiant l'IKM avec le paramètre XML **IKM\_HTTP\_Encrypt\_Content** dans le fichier de configuration qui est envoyé du serveur de mise à disposition au téléphone.
- Si ce chiffrement est appliqué aux fichiers de configuration transmis à partir du serveur de mise à disposition au téléphone, vérifiez que l'en-tête HTTP de *codage de contenu* dans le fichier de configuration comporte « aes128gcm ».

En l'absence de cet en-tête, la méthode AES-256-CBC est prioritaire. Le téléphone applique le déchiffrement AES-256-CBC si une clé AES-256-CBC est présente dans une règle de profil, sans tenir compte de IKM.

- Si vous souhaitez que le téléphone applique ce chiffrement à la configuration des rapports qu'il envoie au serveur de mise à disposition, vérifiez qu'aucune clé AES-256-CBC n'est spécifiée dans la règle de rapport.

## Arguments de resynchronisation facultatifs

Les arguments facultatifs, **key**, **uid**, et **pwd**, peuvent précéder les URL saisies dans les paramètres Profile\_Rule\*, collectivement placés entre crochets.

### clé

L'option **--key** indique au téléphone que le fichier de configuration qu'il reçoit du serveur d'approvisionnement est chiffré avec un Chiffrement AES-256-CBC, sauf si l'en-tête *Content-Encoding* du fichier indique un chiffrement « aes128gcm ». La clé elle-même est spécifiée comme une chaîne suivant le terme **--key**. Vous pouvez placer la clé de chiffrement entre guillemets doubles (") de manière optionnelle. Le téléphone utilise la clé pour déchiffrer le fichier de configuration.

### Exemples d'utilisation

```
[--key VerySecretValue]
[--key "my secret phrase"]
[--key a37d2fb9055c1d04883a0745eb0917a4]
```

Les arguments facultatifs entre crochets font l'objet d'une expansion de macro. Les paramètres à usage spécial, GPP\_SA à GPP\_SD, font l'objet d'expansion de macro en variables macro, \$SA à \$SD, uniquement lorsqu'ils sont utilisés comme arguments facultatifs de la clé. Consultez ces exemples :

```
[--key $SC]
[--key "$SD"]
```

Dans les profils de format ouverts, l'argument de **-key** doit être identique à l'argument facultatif de **-k** fourni à **openssl**.

## ID utilisateur et mot de passe

Les options **uid** et **pwd** peuvent être utilisées pour spécifier l'ID utilisateur et le mot de passe qui seront envoyés en réponse aux défis d'authentification HTTP basique et Digest lorsque l'URL spécifiée est demandée. Les arguments facultatifs entre crochets font l'objet d'une expansion de macro. Les paramètres à usage spécial, GPP\_SA à GPP\_SD, font l'objet d'expansion de macro en variables macro, \$SA à \$SD, uniquement lorsqu'ils sont utilisés comme arguments facultatifs de la clé. Consultez ces exemples :

```
GPP_SA = MyUserID
GPP_SB = MySecretPassword
```

```
[--uid $SA --pwd $SB] https://provisioning_server_url/path_to_your_config/your_config.xml
```

Une fois étendus deviennent :

```
[--uid MyUserID --pwdMySecretPassword]
https://provisioning_server_url/path_to_your_config/your_config.xml
```

## Application d'un profil au téléphone

Après avoir créé un script de configuration XML, il doit être transmis au téléphone pour application. Pour appliquer la configuration, vous pouvez soit télécharger le fichier de configuration sur le téléphone à partir d'un serveur TFTP, HTTP ou HTTPS à l'aide d'un navigateur Web, soit utiliser l'utilitaire de ligne de commande cURL.

## Télécharger le fichier de configuration sur le téléphone à partir d'un serveur TFTP

Procédez comme suit pour télécharger le fichier de configuration d'une application du serveur TFTP sur votre PC.

### Procédure

- 
- Étape 1** Connectez votre ordinateur au LAN du téléphone.
  - Étape 2** Exécutez une application de serveur TFTP sur le PC et assurez-vous que le fichier de configuration est disponible dans le répertoire racine TFTP.
  - Étape 3** Dans un navigateur Web, saisissez l'adresse IP du réseau local du téléphone, l'adresse IP de l'ordinateur, le nom de fichier et les informations de connexion. Utilisez ce format :

```
http://<WAN_IP_Address>/admin/resync?tftp://<PC_IP_Address>/<file_name>&xuser=admin&xpassword=<password>
```

Exemple :

```
http://192.168.15.1/admin/resync?tftp://192.168.15.100/my_config.xml&xuser=admin&xpassword=admin
```

## Téléchargez le fichier de configuration sur le téléphone IP avec cURL

Procédez comme suit pour télécharger la configuration sur le téléphone à l'aide de cURL. Cet outil de ligne de commande est utilisé pour transférer des données avec une syntaxe d'URL. Pour télécharger cURL, consultez :

<https://curl.haxx.se/download.html>



### Remarque

Il est recommandé que vous n'utilisiez pas cURL pour valider la configuration jusqu'au téléphone, car le nom d'utilisateur et le mot de passe peuvent être cURL

### Procédure

#### Étape 1

Connectez votre PC au port LAN du téléphone.

#### Étape 2

Téléchargez le fichier de configuration sur le téléphone en saisissant la commande suivante cURL :

```
curl -d @my_config.xml
"http://192.168.15.1/admin/config.xml&xuser=admin&xpassword=admin"
```

## Types de paramètres de mise à disposition

Cette section décrit les paramètres de mise à disposition organisés dans une large mesure par rapport à une fonction :

Ces types de paramètres de mise à disposition existent :

- Objectif général
- Offre
- Déclenchements
- Horaires configurables
- Règles de profil
- Upgrade Rule

## Paramètres généraux

Les paramètres d'usage général GPP\_\* (**Connexion d'administration > Avancé > Voix > Mise à disposition**) sont utilisés comme registres de chaîne libre, lors de la configuration du téléphone pour interagir avec une solution de serveur de mise à disposition donnée. Les paramètres GPP\_\* sont vides par défaut. Ils peuvent être configurés pour obtenir diverses valeurs, notamment les suivantes :

- Clés de chiffrement.
- URL
- Des informations sur l'état d'une mise à disposition en plusieurs étapes.
- Modèles de requête de publication
- Mappages d'alias de noms de paramètres.
- Des valeurs de chaîne partielles, pouvant être combinées en des valeurs de paramètre complètes.

Les paramètres GPP\_\* sont disponibles pour l'expansion de macro au sein d'autres paramètres de mise à disposition. À cette fin, les noms de macros en majuscules avec une seule lettre (A-P) suffisent pour identifier le contenu du GPP\_A au GPP\_P. En outre, les noms de macros en majuscules à deux lettres SA à SD identifient GPP\_SA à GPP\_SD comme un cas particulier lorsqu'ils sont utilisés comme arguments des options URL suivantes :

**key**, **uid**, et **pwd**

Ces paramètres peuvent être utilisés en tant que variables dans les règles de mise à disposition et mise à niveau. Ils sont référencés en préfixant le nom de la variable par un caractère '\$', comme \$GPP\_A.

## Utiliser des paramètres généraux

Par exemple, si GPP\_A contient la chaîne ABC et GPP\_B contient 123, la macro de l'expression \$A\$B est développée en ABC123.

### Avant de commencer

Accéder à la page Web d'administration du téléphone. Reportez-vous à [Accéder à l'interface Web du téléphone](#).

### Procédure

- 
- |                |   |
|----------------|---|
| <b>Étape 1</b> | Sélectionnez <b>Voix &gt; Mise à disposition</b> .                    |
| <b>Étape 2</b> | Faites défiler jusqu'à la section <b>Paramètres à usage général</b> . |
| <b>Étape 3</b> | Saisissez des valeurs valides dans les champs, A GPP à GPP P.         |
| <b>Étape 4</b> | Cliquez sur <b>Envoyer toutes les modifications</b> .                 |
- 

## Activer des paramètres

Les paramètres Provision\_Enable et Upgrade\_Enable contrôlent toutes les resynchronisations de profil et opérations de mise à niveau du micrologiciel. Ces paramètres contrôlent les resynchronisations et les mises à niveau indépendamment. Ces paramètres contrôlent également les commandes de resynchronisation et de

mise à niveau qui sont émises par le serveur Web d'administration. Ces deux paramètres sont définis sur **Oui** par défaut.

Le paramètre `Resync_From_SIP` contrôle les requêtes de resynchronisation. Un événement SIP NOTIFY est envoyé à partir du serveur de proxy du fournisseur de services au téléphone. S'il est activé, le proxy peut demander une resynchronisation. Pour ce faire, le serveur proxy envoie un message SIP NOTIFY contenant l'événement : en-tête de resynchronisation, au périphérique.

Le périphérique répond à la requête avec une réponse 401 (autorisation refusée pour les informations d'identification utilisées). Le périphérique attend une requête ultérieure authentifiée avant d'exécuter la requête de resynchronisation du proxy. Les en-têtes d'événement : `reboot_now` et : `restart_now` effectuent des redémarrages à froid et à chaud, respectivement, qui sont également validés par le serveur.

Les deux derniers paramètres `Enables` sont `Resync_On_Reset` et `Resync_After_Upgrade_Attempt`. Ces paramètres déterminent si le périphérique effectue une opération de resynchronisation après la séquence d'amorçage du logiciel et après chaque tentative de mise à niveau.

Lorsque `Resync_On_Reset` est activé, le périphérique introduit un délai aléatoire qui suit la séquence d'amorçage avant d'effectuer la réinitialisation. Le délai est une durée aléatoire jusqu'à la valeur indiquée par le paramètre `Resync_Random_Delay` (en secondes). Dans un pool de téléphones qui se mettent en marche simultanément, ce délai permet de répartir les heures de début des demandes de resynchronisation de chaque unité. Cette fonctionnalité peut être utile dans les déploiements résidentiels de grande envergure, en cas de panne d'électricité régionale.

## Déclenchements

Le téléphone permet la resynchronisation à des intervalles spécifiques ou à une heure donnée.

### Resynchroniser à des intervalles spécifiques

Le téléphone est conçu pour se resynchroniser régulièrement avec le serveur de mise à disposition. L'intervalle de resynchronisation est configuré dans `Resync_Periodic` (en secondes). Si cette valeur est laissée vide, ou si elle est égale à zéro, le périphérique n'est pas resynchronisé périodiquement.

La resynchronisation a généralement lieu lorsque les lignes téléphoniques sont inactives. Lorsqu'une ligne vocale est active et qu'une resynchronisation doit avoir lieu, le téléphone retarde la resynchronisation jusqu'à ce que la ligne redevienne inactive. Une resynchronisation peut entraîner la modification des valeurs des paramètres de configuration.

Une resynchronisation peut échouer parce que le téléphone ne parvient pas à récupérer un profil à partir du serveur, parce que le fichier téléchargé est endommagé, ou parce qu'une erreur interne s'est produite. Le périphérique tente d'effectuer une resynchronisation à nouveau après une durée spécifiée dans `Resync_Error_Retry_Delay` (en secondes). Si `Resync_Error_Retry_Delay` est défini sur 0, le périphérique ne tente pas d'effectuer à nouveau une resynchronisation après une tentative de resynchronisation infructueuse.

Si une mise à niveau échoue, une nouvelle tentative est effectuée après `Upgrade_Error_Retry_Delay` secondes.

Deux paramètres configurables sont disponibles pour déclencher sous condition une resynchronisation : `Resync_Trigger_1` et `Resync_Trigger_2`. Chaque paramètre peut être programmé avec une expression conditionnelle qui subit une expansion de macro. À l'expiration de l'intervalle de resynchronisation (heure de la prochaine resynchronisation) les déclencheurs, s'ils ont été définis, empêchent la resynchronisation, sauf si un au moins des déclencheurs est égal à Vrai.

L'exemple de condition suivant déclenche une resynchronisation. Dans cet exemple, la dernière tentative de mise à niveau de téléphone s'est écoulée il y a plus de 5 minutes (300 secondes), et moins de 10 minutes (600 secondes) se sont écoulées depuis la dernière tentative de resynchronisation.

```
$UPGTMR gt 300 and $PRVTMR ge 600
```

## Resynchronisation à une heure donnée

Le paramètre Resync\_At permet au téléphone de se resynchroniser à une heure donnée. Ce paramètre utilise le format 24 heures (hhmm) pour spécifier l'heure.

Le paramètre Resync\_At\_Random\_Delay permet au téléphone d'effectuer une resynchronisation après un délai non spécifié dans le temps. Ce paramètre utilise un format entier positif pour indiquer l'heure.

Saturer le serveur avec des requêtes de resynchronisation de plusieurs téléphones qui sont configurés pour effectuer une resynchronisation en même temps doit être évité. Pour ce faire, le téléphone déclenche la resynchronisation au maximum 10 minutes après l'heure spécifiée.

Par exemple, si vous définissez l'heure de resynchronisation à 1000 (10 h 00), le téléphone déclenchera la resynchronisation à tout instant entre 10 h 00 et 10 h 10

Cette fonctionnalité est désactivée par défaut. Lorsque le paramètre Resync\_At est mis à disposition, le paramètre Resync\_At est ignoré.

## Horaires configurables

Vous pouvez configurer des horaires de resynchronisations périodiques, et vous pouvez spécifier des intervalles de nouvelle tentative en cas d'échec de resynchronisation et de mise à niveau à l'aide de ces paramètres de mise à disposition :

- Resync\_Periodic
- Resync\_Error\_Retry\_Delay
- Upgrade\_Error\_Retry\_Delay

Chaque paramètre accepte un seul délai (en secondes). La nouvelle syntaxe étendue permet une liste séparée par des virgules des délais consécutifs. Le dernier élément de la séquence est répété implicitement de manière continue.

Éventuellement, vous pouvez utiliser un signe plus pour indiquer une valeur numérique qui ajoute un délai aléatoire supplémentaire, comme illustré dans cet exemple.

### Exemple 1

Dans cet exemple, le téléphone effectue périodiquement une resynchronisation toutes les 2 heures. En cas de panne de resynchronisation, le périphérique effectue une nouvelle tentative à ces intervalles : 30 minutes, 1 heure, 2 heures, 4 heures. Le périphérique continue d'essayer à des intervalles de 4 heures jusqu'à ce qu'il se resynchronise avec succès.

```
Resync_Periodic=7200  
Resync_Error_Retry_Delay=1800,3600,7200,14400
```

**Exemple 2**

Dans cet exemple, le périphérique se resynchronise périodiquement toutes les heures (plus un délai supplémentaire aléatoire de 10 minutes). Dans le cas d'une panne de resynchronisation, le périphérique effectue une nouvelle tentative à ces intervalles : 30 minutes (plus 5 minutes au maximum), 1 heure (plus 10 minutes au maximum), 2 heures (plus jusqu'à 15 minutes). Le périphérique continue d'essayer à des intervalles de 2 heures (plus 15 minutes au maximum) jusqu'à ce qu'il se resynchronise avec succès.

```
Resync_Periodic=3600+600
Resync_Error_Retry_Delay=1800+300,3600+600,7200+900
```

**Exemple 3**

Dans cet exemple, si une tentative de mise à niveau à distance échoue, le périphérique retente la mise à niveau au bout de 30 minutes, puis à nouveau après une heure, puis dans deux heures. Si la mise à niveau échoue, le périphérique effectue une nouvelle tentative toutes les quatre à cinq heures jusqu'à ce que la mise à niveau réussisse.

```
Upgrade_Error_Retry_Delay = 1800,3600,7200,14400+3600
```

## Règles de profil

Le téléphone fournit plusieurs paramètres de profil de configuration à distance (Profile\_Rule\*). C'est pourquoi chaque opération de resynchronisation peut récupérer plusieurs fichiers, gérés par des serveurs distincts.

Dans le scénario le plus simple, le périphérique se resynchronise périodiquement à un seul profil sur un serveur principal, qui met à jour tous les paramètres internes qui s'appliquent. Le profil peut aussi être partagé entre différents fichiers. Un seul fichier est commun à tous les téléphones d'un déploiement. Un fichier distinct, unique est fourni pour chaque compte. Les clés de chiffrement et les informations de certificat peuvent être fournies par encore un autre profil, stocké sur un serveur distinct.

Chaque fois qu'une resynchronisation doit être effectuée, le téléphone évalue les quatre paramètres Profile\_Rule\* dans l'ordre :

1. Profile\_Rule
2. Profile\_Rule\_B
3. Profile\_Rule\_C
4. Profile\_Rule\_D

Chaque évaluation peut entraîner une récupération du profil à partir d'un serveur de mise à disposition à distance, avec une mise à jour possible d'un certain nombre de paramètres internes. Si une évaluation échoue, la séquence de resynchronisation est interrompue et est retentée à nouveau au moment spécifié par le paramètre Resync\_Error\_Retry\_Delay (en secondes). Si toutes les évaluations réussissent, le périphérique attend la seconde spécifiée par le paramètre Resync\_Periodic et exécute ensuite un autre resynchronisation.

Le contenu de chaque paramètre Profile\_Rule\* se compose d'un ensemble de solutions alternatives. Les solutions alternatives sont séparées par le caractère | (barre verticale). Chaque solution alternative se compose d'une expression conditionnelle, d'une expression d'affectation, d'une URL de profil et de toutes les options d'URL associées. Tous ces composants sont facultatifs au sein de chaque alternative. Voici les combinaisons valides et l'ordre dans lequel elles doivent s'afficher, le cas échéant :



```
[ conditional-expr ] [ assignment-expr ] [[ options ] URL ]
```

Au sein de chaque paramètre Profile\_Rule\*, toutes les solutions alternatives sauf la dernière, doivent fournir une expression conditionnelle. Cette expression est évaluée et est traitée comme suit :

1. Les conditions sont évaluées de gauche à droite, jusqu'à ce qu'il en existe une qui donne la valeur Vrai (ou jusqu'à ce qu'une alternative soit trouvée sans expression conditionnelle).
2. Toutes les expressions d'affectation d'accompagnement sont évaluées, le cas échéant.
3. Si une URL est spécifiée dans le cadre de cette solution alternative, une tentative est effectuée pour télécharger le profil qui se trouve à l'URL spécifiée. Le système tente de mettre à jour les paramètres internes en conséquence.

Si toutes les alternatives comportent des expressions conditionnelles et qu'aucune ne prend la valeur Vrai (ou si l'ensemble de la règle du profil est vide), le paramètre Profile\_Rule\* tout entier est ignoré. Le paramètre de règle de profil suivant de la séquence est évalué.

### Exemple 1

Cet exemple effectue une resynchronisation inconditionnelle au profil de l'URL spécifiée et une requête HTTP GET au serveur de mise à disposition à distance :

```
http://remote.server.com/cisco/$MA.cfg
```

### Exemple 2

Dans cet exemple, le périphérique se resynchronise sur deux URL différentes, en fonctionnalité de l'état d'enregistrement de la ligne 1. En cas de perte d'enregistrement, le périphérique exécute un HTTP POST vers un script CGI. Le périphérique envoie le contenu de la macro étendue GPP\_A, qui peut fournir des informations supplémentaires sur l'état du périphérique :

```
($PRVTMR ge 600)? http://p.tel.com/has-reg.cfg  
| [--post a] http://p.tel.com/lost-reg?
```

### Exemple 3

Dans cet exemple, le périphérique réalise une resynchronisation sur le même serveur. Le périphérique fournit des informations supplémentaires, si un certificat n'est pas installé sur l'unité (pour les unités antérieures à la 2.0 existantes) :

```
("$CCERT" eq "Installed")? https://p.tel.com/config?  
| https://p.tel.com/config?cisco$MAU
```

### Exemple 4

Dans cet exemple, la ligne 1 est désactivée jusqu'à ce que GPP\_A soit défini comme égal à Mis à disposition par la première URL. Ensuite, il effectue une resynchronisation à la deuxième URL :

```
("$A" ne "Provisioned")? (Line_Enable_1_ = "No");! https://p.tel.com/init-prov
```

```
| https://p.tel.com/configs
```

### Exemple 5

Dans cet exemple, on suppose que le profil que le serveur renvoie contient des balises d'éléments XML. Ces balises doivent être mappées de nouveau aux noms de paramètres appropriés par la correspondance des alias stockée dans GPP\_B :

```
[--alias b] https://p.tel.com/account/$PN$MA.xml
```

Une resynchronisation est généralement considérée comme ayant échoué si un profil demandé n'est pas reçu du serveur. Le paramètre Resync\_Fails\_On\_FNF peut remplacer ce comportement par défaut. Lorsque Resync\_Fails\_On\_FNF est défini sur Non, le périphérique considère la réponse file-not-found du serveur comme une resynchronisation réussie. La valeur par défaut pour Resync\_Fails\_On\_FNF est Oui.

## Upgrade Rule

La règle de mise à niveau informe le périphérique qu'il doit s'activer pour recevoir une nouvelle charge et d'où obtenir la charge, le cas échéant. Si la charge est déjà sur le périphérique, il ne tente pas d'obtenir la charge. Ainsi, la validité de l'emplacement de charge importe peu lorsque la charge souhaitée se trouve dans la partition inactive.

L'Upgrade\_Rule indique une version de micrologiciel qui, si elle est différente de la charge actuelle, est téléchargée et appliquée à moins qu'elle ne soit limitée par une expression conditionnelle ou que Upgrade\_Enable soit défini sur **Non**.

Le téléphone fournit un paramètre de mise à niveau configurable à distance, Upgrade\_Rule. Ce paramètre accepte une syntaxe similaire aux paramètres de règle de profil. Les options d'URL ne sont pas prises en charge pour les mises à niveau, mais les expressions conditionnelles et les expressions d'affectation peuvent être utilisées. Si des expressions conditionnelles sont utilisées, le paramètre peut être rempli avec plusieurs alternatives, séparées par le caractère |. La syntaxe de chaque alternative est la suivante :

```
[ conditional-expr ] [ assignment-expr ] URL
```

Comme dans le cas des paramètres de Profile\_Rule\*, le paramètre Upgrade\_Rule évalue chaque alternative jusqu'à ce qu'une expression conditionnelle soit satisfaite ou qu'une alternative ne comporte aucune expression conditionnelle. L'expression d'affectation d'accompagnement est évaluée, le cas échéant. Puis, une tentative de mise à niveau vers l'URL spécifiée est effectuée.

Si Upgrade\_Rule contient une URL sans expression conditionnelle, le périphérique est mis à niveau vers l'image du micrologiciel que spécifie l'URL. Après l'expansion de macro et l'évaluation de la règle, le périphérique n'effectue pas une nouvelle tentative de mise à niveau jusqu'à ce que la règle soit modifiée ou que la combinaison schéma + serveur + port + chemin d'accès soit modifiée.

Pour tenter une mise à niveau du micrologiciel, le périphérique désactive l'audio au début de la procédure et le redémarre à la fin de la procédure. Le périphérique ne démarre automatiquement une mise à niveau qui est définie par le contenu de Upgrade\_Rule que si toutes les lignes téléphoniques sont actuellement inactives.

Par exemple,

```
https://10.73.10.223/firmware/sip78xx.11-3-1MPP-678.loads
```

Dans cet exemple, Upgrade\_Rule met à niveau le micrologiciel vers l'image qui est stockée à l'adresse indiquée.

Voici un autre exemple :

```
("SF" ne "beta-customer")? http://p.tel.com/firmware/sip78xx.11-3-1MPP-678.loads
| http://p.tel.com/firmware/sip78xx.11-3-1MPP-678.loads
```

Cet exemple indique à l'unité de charger l'une des deux images, en fonction du contenu du paramètre général, GPP\_F.

Le périphérique peut imposer une limite antérieure concernant le numéro de révision du micrologiciel, ce qui peut être une option de personnalisation utile. Si un numéro de révision du micrologiciel valide est configuré dans le paramètre Downgrade\_Rev\_Limit, le périphérique rejette les tentatives de mise à niveau pour les versions d'image antérieures à la limite spécifiée.

## Types de données

Ces types de données sont utilisés avec les paramètres de profil de configuration :

- {a, b, c,...} : à choisir parmi a, b, c,...
- Boolean : valeur booléenne « Oui » ou « Non ».
- CadScript : un miniscript qui spécifie les paramètres de cadence d'un signal. Jusqu'à 127 caractères

Syntaxe : S<sub>1</sub>[ ; S<sub>2</sub>], où :

- S<sub>i</sub> = D<sub>i</sub>(activé<sub>i,1</sub>/ désactivé<sub>i,1</sub>[activé<sub>i,2</sub>/ désactivé<sub>i,2</sub>[activé<sub>i,3</sub>/ désactivé<sub>i,3</sub>[activé<sub>i,4</sub>/ désactivé<sub>i,4</sub>[activé<sub>i,5</sub>/ désactivé<sub>i,5</sub>[activé<sub>i,6</sub>/ désactivé<sub>i,6</sub>]]]]) et est appelé une section.
- activé<sub>i,j</sub> et désactivé<sub>i,j</sub> sont les durées en secondes activées/désactivées d'un *segment.i* = 1 ou 2 et j = 1 à 6.
- D<sub>i</sub> est la durée totale de la section en secondes.

Toutes les durées peuvent posséder jusqu'à trois décimales pour fournir une résolution à 1 ms. Le caractère de remplacement « \* » désigne une durée infinie. Les segments d'une section sont émis dans l'ordre et répétés jusqu'à la fin de la durée totale.

Exemple 1 :

```
60(2/4)

Number of Cadence Sections = 1
Cadence Section 1: Section Length = 60 s
Number of Segments = 1
Segment 1: On=2s, Off=4s

Total Ring Length = 60s
```

Exemple 2 : sonnerie distincte (court, court, court, long) :

```
60(.2/.2,.2/.2,.2/.2,1/4)

Number of Cadence Sections = 1
Cadence Section 1: Section Length = 60s
Number of Segments = 4
```

```
Segment 1: On=0.2s, Off=0.2s
Segment 2: On=0.2s, Off=0.2s
Segment 3: On=0.2s, Off=0.2s
Segment 4: On=1.0s, Off=4.0s
```

```
Total Ring Length = 60s
```

- **DialPlanScript** : la syntaxe de script qui est utilisée pour spécifier les plans de numérotation de la ligne 1 et de la ligne 2.
- **Float<n>**- Valeur à virgule flottante comportant jusqu'à n décimales.
- **FQDN** : nom de domaine complet. Ce script peut atteindre 63 caractères. Voici quelques exemples :
  - sip.Cisco.com:5060 or 109.12.14.12:12345
  - sip.Cisco.com or 109.12.14.12
- **FreqScript** : un miniscript qui spécifie les paramètres de fréquence et de niveau d'une tonalité. Comprend jusqu'à 127 caractères.

Syntaxe :  $F_1@L_1[,F_2@L_2[,F_3@L_3[,F_4@L_4[,F_5@L_5[,F_6@L_6]]]]]$ , où :

- $F_1$ –  $F_6$  sont des fréquences en Hz (uniquement un entier non signé).
- $L_1$ –  $L_6$  sont les niveaux correspondant en décibels (dB) (comporte au maximum une décimale).

Les espaces avant et après la virgule sont autorisés, mais non recommandés.

Exemple 1 : tonalité d'attente d'appel :

```
440@-10

Number of Frequencies = 1
Frequency 1 = 440 Hz at -10 dBm
```

Exemple 2 : tonalité :

```
350@-19,440@-19

Number of Frequencies = 2
Frequency 1 = 350 Hz at -19 dBm
Frequency 2 = 440 Hz at -19 dBm
```

- **IP**- Adresse IPv4 valide sous la forme x.x.x.x, où x est compris entre 0 et 255. Exemple : 10.1.2.100.
- **ID utilisateur** : ID utilisateur tel qu'il apparaît dans une URL ; jusqu'à 63 caractères.
- **Téléphone** : une chaîne de numéro de téléphone, telle que 14081234567, \*69, \*72, 345678 ; ou une URL générique, par exemple, 1234@10.10.10.100:5068 ou jsmith@Cisco.com. La chaîne peut contenir jusqu'à 39 caractères.
- **PhTmpl** : un modèle de numéro de téléphone. Chaque modèle peut contenir un ou plusieurs modèles qui sont séparés par une virgule (.). Les espaces au début de chaque modèle sont ignorés. « ? » et « \* » représentent des caractères génériques. Pour une représentation littérale, utilisez %xx. Par exemple, %2a représente \*. Le modèle peut contenir jusqu'à 39 caractères. Exemples : « 1408\*, 1510\* », « 1408123 ???, 555?1. ».

- Port : numéro de Port TCP/UDP (0 à 65535). Elle peut être spécifiée au format décimal ou hexadécimal.
- ProvisioningRuleSyntax : la syntaxe de script utilisée pour définir la resynchronisation de la configuration et les règles de mise à niveau du micrologiciel.
- PwrLevel : niveau d'alimentation exprimé en dBm avec une décimale, par exemple -13.5 ou 1,5 dBm.
- RscTplmt – Modèle de code de statut de réponse SIP, tel que « 404, 5\* », « 61? », « 407, 408, 487, 481 ». Ce script peut atteindre 39 caractères.
- Sig<n> – Valeur signée de n bits. Elle peut être spécifiée au format décimal ou hexadécimal. Un signe « - » doit précéder les valeurs négatives. Un signe + précédant les valeurs positives est facultatif.
- Codes étoile : le code d'activation d'un service supplémentaire, par exemple \*69. Le code peut contenir jusqu'à 7 caractères.
- Str<n> – Chaîne générique comportant jusqu'à n caractères non réservés.
- Durée<n> – Durée en secondes, avec jusqu'à n décimales. Les décimales supplémentaires spécifiées sont ignorées.
- ToneScript : un mini-script qui détermine les paramètres de fréquence, de niveau et de cadence d'une tonalité de progression d'appel. Le script peut contenir jusqu'à 127 caractères.

Syntaxe : FreqScript;Z<sub>1</sub>[:Z<sub>2</sub>].

La section Z<sub>1</sub> est similaire à la section S<sub>1</sub> d'un CadScript, sauf que chaque segment activé/désactivé est suivi d'un paramètre de composants de fréquence : Z<sub>1</sub> = D<sub>1</sub>(activé<sub>i,1</sub>/ désactivé<sub>i,1</sub>/f<sub>i,1</sub>[activé<sub>i,2</sub>/ désactivé<sub>i,2</sub>/f<sub>i,2</sub>[activé<sub>i,3</sub>/ désactivé<sub>i,3</sub>/f<sub>i,3</sub>[, activé<sub>i,4</sub>/ désactivé<sub>i,4</sub>/f<sub>i,4</sub>[activé<sub>i,5</sub>/ désactivé<sub>i,5</sub>/f<sub>i,5</sub>[activé<sub>i,6</sub>/ désactivé<sub>i,6</sub>/f<sub>i,6</sub>]]]])) où :

- $f_{i,j} = n_1[+n_2]+n_3[+n_4[+n_5[+n_6]]]]]$ .
- $1 < n_k < 6$  spécifie les composants de fréquence du FreqScript qui sont utilisés dans ce segment.

Si plus d'un composant de fréquence est utilisé dans un segment, les composants sont additionnés.

Exemple 1 : tonalité :

```
350@-19,440@-19;10(*0/1+2)

Number of Frequencies = 2
Frequency 1 = 350 Hz at -19 dBm
Frequency 2 = 440 Hz at -19 dBm
Number of Cadence Sections = 1
Cadence Section 1: Section Length = 10 s
Number of Segments = 1
Segment 1: On=forever, with Frequencies 1 and 2

Total Tone Length = 10s
```

Exemple 2 : tonalité à répétition :

```
350@-19,440@-19;2(.1/.1/1+2);10(*0/1+2)

Number of Frequencies = 2
Frequency 1 = 350 Hz at -19 dBm
Frequency 2 = 440 Hz at -19 dBm
Number of Cadence Sections = 2
```

```

Cadence Section 1: Section Length = 2s
Number of Segments = 1
Segment 1: On=0.1s, Off=0.1s with Frequencies 1 and 2
Cadence Section 2: Section Length = 10s
Number of Segments = 1
Segment 1: On=forever, with Frequencies 1 and 2

Total Tone Length = 12s

```

- Uns<n>-Valeur non signée de n bits, où n = 8, 16 ou 32. Elle peut être spécifiée au format décimal ou hexagonal, comme 12 ou 0x18, tant que la valeur peut tenir sur n bits.




---

**Remarque**

Gardez à l'esprit :

- <Par Name> représente un nom de paramètre de configuration. Dans un profil, la balise correspondante est formée en remplaçant l'espace par un trait de soulignement « \_ », par exemple **Par\_Name**.
  - Un champ de valeur par défaut vide implique une chaîne vide <"">.
  - Le téléphone continue d'utiliser les dernières valeurs configurées pour les balises qui ne sont pas présentes dans un profil donné.
  - Les modèles sont comparés dans l'ordre indiqué. La première, *et non la plus proche*, correspondance est sélectionnée. Le nom du paramètre doit correspondre exactement.
  - Si plus d'une définition d'un paramètre est indiquée dans un profil, la dernière définition du fichier est celle qui est appliquée dans le téléphone.
  - Une spécification de paramètre avec une valeur de paramètre vide force le paramètre à sa valeur par défaut. Pour spécifier une chaîne vide au lieu de cela, utilisez une chaîne vide «» en tant que valeur du paramètre.
- 

## Mises à jour de profil et mises à niveau du micrologiciel

Le téléphone prend en charge la mise à disposition à distance sécurisée (configuration) et les mises à niveau du micrologiciel. Un téléphone non mis à disposition peut recevoir un profil chiffré destiné à ce périphérique. Le téléphone ne nécessite pas de clé explicite en raison d'un mécanisme de première mise à disposition sécurisé qui utilise la fonctionnalité SSL.

L'intervention de l'utilisateur n'est pas nécessaire pour démarrer ou effectuer une mise à jour de profil, ou une mise à niveau du micrologiciel, ou si des mises à niveau intermédiaires sont nécessaires pour atteindre un état de mise à niveau postérieur à partir d'une version antérieure. Une resynchronisation du profil n'est retentée que lorsque le téléphone IP Cisco est inactif, car une resynchronisation peut déclencher un redémarrage du logiciel et mettre fin à un appel.

Les paramètres généraux gèrent le processus de mise à disposition. Tous les téléphones peuvent être configurés pour contacter régulièrement un serveur de mise à disposition normale (NPS). La communication avec le serveur de mise à disposition normale ne nécessite pas l'utilisation d'un protocole sécurisé, car le profil mis à jour est crypté par une clé secrète partagée. Le serveur NPS peut être un serveur TFTP, HTTP ou HTTPS standard avec des certificats clients.

L'administrateur peut mettre à niveau, redémarrer, ou resynchroniser les téléphones à l'aide de l'interface utilisateur Web du téléphone. L'administrateur peut également effectuer ces tâches à l'aide d'un message de notification SIP.

Les profils de configuration sont générés à l'aide d'outils open source communs qui s'intègrent aux systèmes de mise à disposition des prestataires de services.

## Autoriser les mises à jour de profil

Des mises à jour du profil peuvent être autorisées à intervalles réguliers. Les profils mis à jour sont envoyés à partir d'un serveur sur le téléphone à l'aide de TFTP, HTTP ou HTTPS.

Vous pouvez également configurer les paramètres dans le fichier de configuration du téléphone avec le code XML(cfg.xml).

### Avant de commencer

Accéder à la page Web d'administration du téléphone. Reportez-vous à [Accéder à l'interface Web du téléphone](#).

### Procédure

---

**Étape 1** Sélectionnez **Voix -> Mise à disposition**.

**Étape 2** Dans la section **Profil de configuration**, sélectionnez **Oui** pour le paramètre **Activation de la mise à disposition**.

Vous pouvez également configurer ce paramètre dans le fichier de configuration XML du téléphone (cfg.xml) en entrant une chaîne au format suivant :

```
<Provision_Enable ua="na">Yes</Provision_Enable>
```

Par défaut : Oui

**Étape 3** Définissez les paramètres comme décrit dans le tableau [Paramètres de resynchronisation du profil](#).

**Étape 4** Cliquez sur **Envoyer toutes les modifications**.

---

## Autoriser et configurer les mises à niveau du micrologiciel

Des mises à jour du micrologiciel peuvent être autorisées à intervalles réguliers. Le micrologiciel mis à jour est envoyé à partir d'un serveur sur le téléphone à l'aide de TFTP ou HTTP. La sécurité est moins un problème avec une mise à niveau du micrologiciel, car le micrologiciel ne contient pas d'informations personnelles.

Vous pouvez également configurer les paramètres dans le fichier de configuration du téléphone avec le code XML(cfg.xml).

### Avant de commencer

Accéder à la page Web d'administration du téléphone. Reportez-vous à [Accéder à l'interface Web du téléphone](#).

### Procédure

---

**Étape 1** Sélectionnez **Voix > Mise à disposition**.

**Étape 2** Dans la section **Mise à niveau de micrologiciel**, sélectionnez **Oui** à partir du paramètre **Activation de la mise à niveau**.

Vous pouvez également configurer ce paramètre dans le fichier de configuration XML du téléphone (cfg.xml) en entrant une chaîne au format suivant :

```
<Upgrade_Enable ua="na">Yes</Upgrade_Enable>
```

Options : Oui et Non

Par défaut : Oui

**Étape 3** Configurez le paramètre **Délai d'attente des tentatives en cas d'erreur de mise à niveau** en secondes.

L'intervalle pour réessayer la mise à niveau (en secondes) s'applique lorsque la mise à niveau échoue. Le périphérique dispose d'un temporisateur d'erreur de mise à niveau du micrologiciel qui s'active après l'échec d'une tentative de cette mise à niveau. Le temporisateur est initialisé avec la valeur configurée dans ce paramètre. La prochaine tentative de mise à niveau du micrologiciel sera effectuée lorsque le décompte de ce temporisateur arrivera à zéro.

Vous pouvez également configurer ce paramètre dans le fichier de configuration XML du téléphone (cfg.xml) en entrant une chaîne au format suivant :

```
<Upgrade_Error_Retry_Delay ua="na">3600</Upgrade_Error_Retry_Delay>
```

Valeur par défaut : 3 600

:

```
<tftp|http|https>://<ip address>/image/<load name>
```

**Étape 4** Définissez le paramètre **Règle de mise à niveau** en entrant un script de mise à niveau du micrologiciel définit les conditions de la mise à niveau et les adresses URL associées du micrologiciel. Il utilise la même syntaxe que le paramètre Profile Rule. Saisissez un script et utilisez le format qui suit pour saisir la règle de mise à niveau :

```
<tftp|http|https>://<ipaddress>/image/<load name>
```

Par exemple :

```
tftp://192.168.1.5/image/sip88xx.11-0-0MPP-BN.loads
```

```
tftp://192.168.1.5/image/sip78xx.11-0-1MPP-BN.loads
```

Vous pouvez également configurer ce paramètre dans le fichier de configuration XML du téléphone (cfg.xml) en entrant une chaîne au format suivant :

```
<Upgrade_Rule ua="na">http://10.74.10.205:6970/sip8845_65.0104-MPP-9875dev.loads</Upgrade_Rule>
```

**Étape 5** Cliquez sur **Envoyer toutes les modifications**.

## Mise à niveau du micrologiciel par TFTP, HTTP ou HTTPS

Le téléphone prend en charge la mise à niveau du micrologiciel par TFTP, HTTP, ou HTTPS.



### Remarque

Les régressions à des versions antérieures ne sont pas disponibles pour tous les périphériques. Pour plus d'informations, voir les notes de version de votre version du micrologiciel et du téléphone.



### Avant de commencer

Le fichier de chargement du micrologiciel doit être téléchargé sur un serveur accessible.

### Procédure

---

- Étape 1** Renommer l'image en procédant comme suit :
- `cp-x8xx-sip.aa-b-cMPP.cop` pour `cp-x8xx-sip.aa-b-cMPP.tar.gz`
- où
- x8xx** est la série de téléphones, par exemple 7811 ou 7832.
- aa-b-c** est le numéro de version, par exemple 10-4-1
- Étape 2** Utilisez la commande `tar -xvzf` pour décompresser le composant tar.
- Étape 3** Copiez le dossier dans un répertoire de téléchargement TFTP, HTTP, ou HTTPS
- Étape 4** Accéder à la page Web d'administration du téléphone. Reportez-vous à [Accéder à l'interface Web du téléphone](#).
- Étape 5** Sélectionnez **Voix - > Mise à disposition**.
- Étape 6** Recherchez le nom de fichier de chargement qui se termine par **.loads** et l'ajouter à l'URL valide.
- Étape 7** Cliquez sur **Envoyer toutes les modifications**.
- 

## Mettre à niveau le micrologiciel à l'aide d'une commande de navigateur

Une commande de mise à niveau saisie dans la barre d'adresse de navigateur peut être utilisée pour mettre à niveau le micrologiciel sur un téléphone. Le téléphone ne se met à jour que lorsqu'il est inactif. La mise à jour est retentée automatiquement une fois l'appel terminé.

### Procédure

---

Pour mettre à niveau le téléphone avec une URL dans un navigateur Web, entrez la commande suivante :

```
http://<phone_ip>/admin/upgrade?<schema>://<serv_ip[:port]>/filepath
```

---

