



## Bereitstellungsformate

---

- [Konfigurationsprofile](#) , auf Seite 1
- [Konfigurationsprofil-Formate](#), auf Seite 1
- [Open-Format-Profil \(XML\) – Komprimierung und Verschlüsselung](#), auf Seite 5
- [Anwendung eines Profils auf dem Telefon](#), auf Seite 12
- [Parameter-Typen für die Bereitstellung](#), auf Seite 13
- [Datentypen](#), auf Seite 19
- [Profil-Updates und Firmware-Upgrades](#), auf Seite 23

## Konfigurationsprofile

Das Telefon kann im XML-Format konfiguriert werden.

In den Beispielen in diesem Dokument werden Konfigurationsprofile mit einer Syntax im XML-Format (XML) verwendet.

Detaillierte Informationen zu Ihrem Telefon finden Sie im Administratorhandbuch für Ihr Gerät. In allen Handbüchern werden die Parameter beschrieben, die über den Verwaltungswebserver konfiguriert werden können.

## Konfigurationsprofil-Formate

Im Konfigurationsprofil werden die Parameterwerte für das Telefon definiert.

Für das XML-Format des Konfigurationsprofils werden Standard-XML-Entwicklungstools verwendet, um die Parameter und Werte zu kompilieren.



---

**Hinweis** Es wird nur der UTF-8-Zeichensatz unterstützt. Wenn Sie das Profil in einem Editor bearbeiten, ändern Sie das Verschlüsselungsformat nicht, ansonsten wird die Datei vom Telefon nicht erkannt.

---

Jedes Telefon verfügt über einen anderen Funktionssatz und somit auch über einen anderen Parametersatz.

### Profil im XML-Format (XML)

Das Open-Format-Profil ist eine Textdatei mit einer XML-ähnlichen Syntax in einer Hierarchie von Elementen mit Elementattributen und Werten. Mit diesem Format können Sie Standardtools verwenden, um die Konfigurationsdatei zu erstellen. Eine Konfigurationsdatei in diesem Format kann während eines Resynchronisierungsvorgangs vom Bereitstellungsserver an das Telefon gesendet werden. Die Datei kann ohne Kompilierung als binäres Objekt gesendet werden.

Das Telefon unterstützt Konfigurationsformate, die von Standardtools generiert werden. Diese Funktion erleichtert die Entwicklung von Back-End-Bereitstellungssoftware, mit der Konfigurationsprofile aus vorhandenen Datenbanken generiert werden.

Um vertrauliche Informationen im Konfigurationsprofil zu schützen, übermittelt der Bereitstellungsserver diese Art von Datei über einen durch TLS geschützten Kanal an das Telefon. Optional kann die Datei mithilfe des gzip-Deflate-Algorithmus (RFC1951) komprimiert werden.

Die Datei kann mit einer der folgenden Verschlüsselungsmethoden verschlüsselt werden:

- AES-256-CBC-Verschlüsselung
- RFC-8188-basierte HTTP-Inhaltsverschlüsselung mit AES-128-GCM-Schlüssel

### Beispiel: Open-Format-Profil

```
<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>
```

Das Element-Tag <flat-profile> umfasst alle Parameterelemente, die vom Telefon erkannt werden.

## Komponenten der Konfigurationsdatei

Eine Konfigurationsdatei kann die folgenden Komponenten enthalten:

- Element-Tags
- Attribute
- Parameter
- Formatierungsfunktionen
- XML-Kommentare

### Eigenschaften der Element-Tags

- Das XML-Bereitstellungsformat und die Webbenutzeroberfläche ermöglichen die Konfiguration der gleichen Einstellungen. Der Name des XML-Tags und die Feldnamen in der Webbenutzeroberfläche ähneln sich, können aber aufgrund der Beschränkungen beim XML-Elementnamen variieren. Beispielsweise werden Unterstriche ( \_ ) anstelle von Anführungszeichen ( „ „ ) verwendet.

- Vom Telefon werden Elemente mit entsprechenden Parameternamen erkannt, die im speziellen <flat-profile>-Element gekapselt sind.
- Elementnamen werden in spitze Klammern gesetzt.
- Die meisten Elementnamen ähneln den Feldnamen auf den Verwaltungswebseiten für das Gerät, wobei die folgenden Modifikationen gelten:

- Elementnamen dürfen keine Leerzeichen oder Sonderzeichen enthalten. Um den Elementnamen aus dem Verwaltungs-Web-Feldnamen abzuleiten, ersetzen Sie alle Leerzeichen oder Sonderzeichen durch einen Unterstrich [ \_ ], ( \_ ) oder / .

**Beispiel:** Das Element <Resync\_On\_Reset> steht für das Feld **Beim Zurücksetzen erneut synchronisieren**.

- Jeder Elementname muss eindeutig sein. Auf den Verwaltungswebseiten können die gleichen Felder auf mehreren Webseiten angezeigt werden, z. B. die Seiten „Leitung“, „Benutzer“ und „Durchwahl“. Hängen Sie [n] an den Elementnamen an, um die Nummer anzugeben, die auf der Registerkarte „Seite“ angezeigt wird.

**Beispiel:** Das Element <Dial\_Plan\_1\_> steht für **Rufnummernplan** für Leitung 1.

- Jedes öffnende Element-Tag muss über ein entsprechendes schließendes Element-Tag verfügen. Zum Beispiel:

```
<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>
```

- Bei Element-Tags wird die Groß-/Kleinschreibung beachtet.
- Leere Element-Tags sind zulässig und werden als Wert ohne Konfiguration interpretiert. Geben Sie das öffnende Element-Tag ohne ein entsprechendes Element-Tag ein, und fügen Sie ein Leerzeichen und einen Vorwärtsschrägstrich vor der schließenden spitzen Klammer (>) ein. In diesem Beispiel ist Profilregel B leer:

```
<Profile_Rule_B />
```

- Ein leeres Element-Tag kann verwendet werden, um zu verhindern, dass die während einer Resynchronisierung durch einen Benutzer eingegebenen Werte überschrieben werden. Im folgenden Beispiel bleiben die Kurzwahleinstellungen für Benutzer unverändert:

```
<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>
```

- Verwenden Sie einen leeren Wert, um den entsprechenden Parameter auf eine leere Zeichenfolge festzulegen. Geben Sie ein öffnendes und ein schließendes Element ohne Wert ein. Im folgenden Beispiel wird der Parameter GPP\_A auf eine leere Zeichenfolge festgelegt.

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

- Nicht erkannte Elementnamen werden ignoriert.

## Parametereigenschaften

Diese Eigenschaften gelten für die Parameter:

- Alle Parameter, die von keinem Profil festgelegt sind, bleiben auf dem Telefon unverändert.
- Nicht erkannte Parameter werden ignoriert.
- Wenn das Open-Format-Profil mehrere Vorkommen des gleichen Parameter-Tags enthält, überschreibt das letzte dieser Vorkommen alle früheren Vorkommen. Um ein versehentliches Überschreiben der Konfigurationswerte für einen Parameter zu vermeiden, wird empfohlen, dass in jedem Profil immer nur eine Instanz eines Parameters festgelegt wird.
- Das zuletzt verarbeitete Profil hat Vorrang. Wenn in mehreren Profilen der gleiche Konfigurationsparameter angegeben ist, hat der Wert des letzten Profils Vorrang.

## Formate der Zeichenfolge

Die folgenden Eigenschaften gelten für die Formatierung von Zeichenfolgen:

- Kommentare sind über die standardmäßige XML-Syntax zulässig.
 

```
<!-- My comment is typed here -->
```
- Vor- und nachstehende Leerzeichen sind für bessere Lesbarkeit zulässig, werden jedoch aus dem Parameterwert entfernt.
- Neue Zeilen in einem Wert werden in Leerzeichen konvertiert.
- Ein XML-Header in Form von <? ?> ist zulässig, wird jedoch vom Telefon ignoriert.
- Verwenden Sie zum Eingeben von Sonderzeichen grundlegende XML-Escape-Zeichen, wie in der folgenden Tabelle dargestellt.

Sonderzeichen	XML-Escape-Sequenz
& (Und-Zeichen)	&amp;
< (kleiner als)	&lt;

Sonderzeichen	XML-Escape-Sequenz
> (größer als)	&gt;
' (Apostroph)	&apos;
" (doppelte Anführungszeichen)	&quot;

Im folgenden Beispiel werden die Escape-Zeichen eingegeben, um die Symbole für Größer als und Kleiner als darzustellen, die in einer Rufnummernplan-Regel erforderlich sind. In diesem Beispiel wird ein Rufnummernplan für eine Informationshotline definiert, bei dem der Parameter `<Dial_Plan_1_>` (**Administratoranmeldung > Erweitert > Sprache > Nebenstelle (n)**) gleich (S0 <:18005551212>) festgelegt ist.

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

- Numerische Escape-Zeichen, in denen Dezimal- und Hexadezimalwerte verwendet werden (z. B. `&#40;` und `&#x2e;`) sind umgesetzt.
- Die Telefon-Firmware unterstützt nur ASCII-Zeichen.

## Open-Format-Profil (XML) – Komprimierung und Verschlüsselung

Das Open-Format-Konfigurationsprofil kann komprimiert werden, um die Netzwerkbelastung auf dem Bereitstellungsserver zu reduzieren. Das Profil kann auch zum Schutz von vertraulichen Informationen verschlüsselt werden. Die Komprimierung ist nicht erforderlich, muss jedoch vor der Verschlüsselung erfolgen.

### Open-Format-Profil – Komprimierung

Die unterstützte Komprimierungsmethode ist der gzip-Deflate-Algorithmus (RFC1951). Das gzip-Utility und die Komprimierungsbibliothek, die den gleichen Algorithmus (zlib) implementiert, stehen im Internet zur Verfügung.

Um die Komprimierung ermitteln zu können, erwartet das Telefon, dass die komprimierte Datei einen gzip-kompatiblen Header enthält. Durch Aufruf des gzip-Utility im ursprünglichen Open-Format-Profil wird der Header generiert. Die heruntergeladene Header-Datei wird vom Telefon überprüft, um das Dateiformat zu bestimmen.

Wenn beispielsweise `profile.xml` ein gültiges Profil ist, wird die Datei `profile.xml.gz` ebenfalls akzeptiert. Dieser Profiltyp kann über einen der folgenden Befehle generiert werden:

- `>gzip profile.xml`  
Ersetzt die Originaldatei durch die komprimierte Datei.
- `>cat profile.xml | gzip > profile.xml.gz`  
Belässt die Originaldatei und erstellt eine neue komprimierte Datei.

Ein Tutorial zur Komprimierung steht im Abschnitt [Offenes Profil mit Gzip komprimieren](#) zur Verfügung.

## Open-Format-Profil – Verschlüsselung

Die symmetrische Verschlüsselung kann verwendet werden, um ein Open-Format-Konfigurationsprofil zu verschlüsseln, unabhängig davon, ob die Datei komprimiert ist. Die Komprimierung muss, soweit sie angewendet wird, vor der Verschlüsselung durchgeführt werden.

Der Bereitstellungsserver verwendet HTTPS, um die anfängliche Bereitstellung des Telefons nach der Einrichtung abzuwickeln. Die Offline-Vorverschlüsselung von Konfigurationsprofilen ermöglicht die anschließende Verwendung von HTTP für die Resynchronisierung von Profilen. Dadurch wird die Belastung des HTTPS-Servers in großen Bereitstellungen reduziert.

Das Telefon unterstützt zwei Methoden zur Verschlüsselung für Konfigurationsdateien:

- AES-256-CBC-Verschlüsselung
- RFC-8188-basierte HTTP-Inhaltsverschlüsselung mit AES-128-GCM-Schlüssel

Der Schlüssel oder das Input Keying Material (IKM) muss zuvor für das Gerät bereitgestellt worden sein. Bootstrapping des Geheimschlüssels kann über HTTPS sicher erfolgen.

Der Konfigurationsdateiname erfordert kein bestimmtes Format, aber ein Dateiname, der mit der Erweiterung `.cfg` endet, gibt normalerweise ein Konfigurationsprofil an.

### AES-256-CBC-Verschlüsselung

Das Telefon unterstützt die AES-256-CBC-Verschlüsselung für Konfigurationsdateien.

Das OpenSSL-Verschlüsselungstool kann von verschiedenen Internetseiten heruntergeladen und für die Verschlüsselung verwendet werden. Zur Unterstützung der 256-Bit-AES-Verschlüsselung ist möglicherweise eine erneute Kompilierung des Tools zur Aktivierung des AES-Codes erforderlich. Die Firmware wurde mit Version `openssl-1.1.1d` getestet.

[Ein Profil mit OpenSSL verschlüsseln](#) bietet ein Tutorial zur Verschlüsselung.

Bei einer verschlüsselten Datei erwartet das Profil, dass die Datei dasselbe Format aufweist wie bei der Generierung mit dem folgenden Befehl:

```
# 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
```

Ein kleingeschriebenes `-k` steht vor dem Geheimschlüssel; dies kann eine beliebige Nur-Text-Phrase sein und wird verwendet, um einen 64-Bit-Zufallssalt zu generieren. Mit dem durch das `-k`-Argument angegebenen Geheimnis leitet das Verschlüsselungstool einen zufälligen 128-Bit-Anfangsvektor und den tatsächlichen 256-Bit-Verschlüsselungscode ab.

Wenn diese Form der Verschlüsselung in einem Konfigurationsprofil verwendet wird, muss das Telefon den geheimen Schlüsselwert erhalten, um die Datei entschlüsseln zu können. Dieser Wert wird als Qualifizierer in der URL für das Profil angegeben. Die Syntax lautet unter Verwendung einer expliziten URL wie folgt:

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

Dieser Wert wird mithilfe eines der Profile\_Rule-Parameter programmiert.

## Makroerweiterung

Mehrere Bereitstellungsparameter werden intern vor der Auswertung einer Makroerweiterung unterzogen. Dieser Vorbewertungsschritt bietet mehr Flexibilität bei der Steuerung der Resynchronisierung des Telefons und der Upgrade-Aktivitäten.

Die folgenden Parametergruppen werden vor der Auswertung einer Makroerweiterung unterzogen:

- Resync\_Trigger\_\*
- Profile\_Rule\*
- Log\_xxx\_Msg
- Upgrade\_Rule

Unter bestimmten Umständen werden einige allgemeine Parameter (GPP\_\*) auch einer Makroerweiterung unterzogen. Dies wird explizit in [Optionale Argumente für die Resynchronisierung, auf Seite 11](#) angegeben.

Während der Makroerweiterung ersetzen die Inhalte der benannten Variablen die Ausdrücke der Form \$NAME und \$(NAME). Diese Variablen umfassen allgemeine Parameter, mehrere Produktbezeichner, bestimmte Ereignistimer und Bereitstellungsstatus-Werte. Eine vollständige Liste finden Sie im [Makroerweiterungsvariablen](#).

Im folgenden Beispiel wird der Ausdruck \$(MAU) verwendet, um die MAC-Adresse 000E08012345 einzufügen.

Der Administrator gibt Folgendes ein: **\$(MAU) config.cfg**

Die resultierende Makroerweiterung für ein Gerät mit der MAC-Adresse 000E08012345 lautet:  
000E08012345config.cfg

Wenn ein Makroname nicht erkannt wird, wird er nicht erweitert. Der Name STRANGE wird beispielsweise nicht als gültiger Makroname erkannt, während MAU als gültiger Makroname erkannt wird.

Der Administrator gibt Folgendes ein: **\$(STRANGE)\$MAU.cfg**

Die resultierende Makroerweiterung für ein Gerät mit der MAC-Adresse 000E08012345 lautet:  
\$(STRANGE)000E08012345.cfg

Eine Makroerweiterung wird nicht rekursiv angewendet. \$\$MAU" wird beispielsweise in \$MAU" erweitert (\$\$ wird erweitert) und nicht in die MAC-Adresse.

Der Inhalt der speziellen Parameter GPP\_SA bis GPP\_SD wird den Makroausdrücken \$SA bis \$SD zugeordnet. Für diese Parameter wird die Makroerweiterung nur als Argument der Optionen **--key**, **--uid** und **--pwd** in einer Resynchronisierungs-URL durchgeführt.

## Bedingungsdrücke

Bedingungsdrücke können Resynchronisierungsereignisse auslösen und alternative URLs für die Resynchronisierung und Upgrade-Vorgänge auswählen.

Bedingungsdrücke bestehen aus einer Liste von Vergleichen, getrennt durch den Operator **and**. Alle Vergleiche müssen erfüllt werden, damit für die Bedingung „True“ ausgegeben wird.

Jeder Vergleich kann sich auf eine der folgenden drei Arten von Buchstabensymbolen beziehen:

- Ganzzahlige Werte
- Software- oder Hardware-Versionsnummern
- Zeichenfolgen in doppelten Anführungszeichen

### Versionsnummern

Die Software-Version für Cisco IP-Telefone mit Multiplattform-Firmware verwendet dieses Format (wobei *BN* die Buildnummer ist):

- Für Firmware-Version 11.3(1)SR1 und früher: `sipjjjj.11-0-1MPP-376`  
Wobei *yyyy* das Telefonmodell oder die Telefon-Serie angibt; *11* ist die Hauptversion; *0* ist die Nebenversion; *1MPP* ist die Mikroversion; und *376* ist die Buildnummer.
- Für Firmware-Version 11.3(2) und früher: `sipjjjj.11-3-2MPP0001-609`  
Wobei *yyyy* das Telefonmodell oder die Telefon-Serie angibt; *11* ist die Hauptversion; *3* ist die Nebenversion; *2MPP0001* ist die Mikroversion; und *609* ist die Buildnummer.

In der Vergleichszeichenfolge muss dasselbe Format verwendet werden. Andernfalls führt dies zu einem Format-Analysefehler.

Beim Vergleich der Software-Version werden die Hauptversion, Nebenversion und Mikroversion nacheinander abgeglichen; dabei haben die am weitesten links stehenden Ziffern Vorrang vor den anderen Ziffern. Wenn die Versionsnummern identisch sind, wird die Buildnummer verglichen.

### Beispiele für eine gültige Versionsnummer

- Für Firmware-Version 11.3(1)SR1 und früher:

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

- Für Firmware-Version 11.3(2) und später:

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

### Vergleich

- Für Firmware-Version 11.3(1)SR1 und früher:

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

- Für Firmware-Version 11.3(2) und später:

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

Zeichenfolgen in Anführungszeichen können auf Gleichheit oder Ungleichheit verglichen werden. Ganzzahlen und Versionsnummern können auch arithmetisch verglichen werden. Die Vergleichsoperatoren können als Symbole oder als Akronyme angegeben werden. Akronyme eignen sich für die Bedingung in einem Open-Format-Profil.



Operator	Alternative Syntax	Beschreibung	Gilt für Ganzzahl- und Versions-Operanden	Gilt für Operanden von Zeichenfolgen in Anführungszeichen
=	eq	ist gleich	Ja	Ja
!=	ne	ist ungleich	Ja	Ja
<	lt	kleiner als	Ja	Nein
<=	le	kleiner oder gleich	Ja	Nein
>	gt	größer als	Ja	Nein
>=	ge	größer oder gleich	Ja	Nein
UND		und	Ja	Ja

Es ist wichtig, Makrovariablen in doppelte Anführungszeichen zu setzen, wenn ein Buchstabensymbol einer Zeichenfolge erwartet wird. Wenn eine Zahl oder Versionsnummer erwartet wird, sollten Sie nicht so vorgehen.

Bei Verwendung in Zusammenhang mit den Parametern `Profile_Rule*` und `Upgrade_Rule` müssen Bedingungsausdrücke in der Syntax „(expr)?“ wie in diesem Beispiel für eine Upgrade-Regel integriert werden. Denken Sie daran, *BN* durch die Buildnummer Ihrer Firmware-Version zu ersetzen.

- Für Firmware-Version 11.3(1)SR1 und früher

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

- Für Firmware-Version 11.3(2) und später

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

Verwenden Sie die vorstehende Syntax nicht mit Klammern zur Konfiguration der `Resync_Trigger_*`-Parameter.

## URL-Syntax

Verwenden Sie die Standard-URL-Syntax, um anzugeben, wie Konfigurationsdateien und Firmware jeweils in den Parametern `Profile_Rule*` und `Upgrade_Rule` abgerufen werden. Die Syntax lautet wie folgt:

```
[ scheme:// ] [ server [:port]] filepath
```

Wobei **scheme** einer der folgenden Werte ist:

- tftp
- http
- https

Wenn **scheme** nicht angegeben wird, wird TFTP angenommen. Der Server kann ein DNS-anerkannter Host-Name oder eine numerische IP-Adresse sein. Der Port ist die Ziel-UDP- oder -TCP-Portnummer. Der Dateipfad muss mit dem Stammverzeichnis (/) beginnen. Es muss sich um einen absoluten Pfad handeln.

Wenn **server** nicht angegeben wird, wird der über DHCP (Option 66) angegebene TFTP-Server verwendet.



**Hinweis** Für Upgrade-Regeln muss der Server angegeben werden.

Wenn **port** nicht angegeben wird, wird der Standard-Port für das angegebene Schema verwendet. TFTP verwendet UDP-Port 69, HTTP verwendet TCP-Port 80, HTTPS verwendet TCP-Port 443.

Es muss ein Dateipfad vorhanden sein. Dieser muss nicht unbedingt zu einer statischen Datei verweisen, sondern kann dynamischen Inhalt angeben, der über CGI abgerufen wird.

Die Makroerweiterung gilt innerhalb von URLs. Im Folgenden erhalten Sie Beispiele für gültige URLs:

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

Beim Verwenden der DHCP-Option 66 wird die leere Syntax nicht von der Upgrade-Regel unterstützt. Dies gilt nur für Profile Rule\*.

## RFC-8188-basierte HTTP-Inhaltsverschlüsselung

Das Telefon unterstützt die RFC 8188-basierte HTTP-Inhaltsverschlüsselung mit AES-128-GCM-Schlüssel für Konfigurationsdateien. Mit dieser Verschlüsselungsmethode kann jede Entität die HTTP-Nachrichten-Header lesen. Nur die Entitäten, die das Input Keying Material (IKM) kennen, können die Nutzlast lesen. Wenn das Telefon mit IKM bereitgestellt wird, können das Telefon und der Bereitstellungsserver Konfigurationsdateien sicher austauschen und Netzwerkelementen von Drittanbietern gleichzeitig ermöglichen, die Nachrichten-Header zu Analyse- und Überwachungszwecken zu verwenden.

Der XML-Konfigurationsparameter **IKM\_HTTP\_Encrypt\_Content** enthält das IKM auf dem Telefon. Aus Sicherheitsgründen ist dieser Parameter nicht auf der Webseite der Telefon-Verwaltung zugänglich. Er ist ebenfalls nicht in der Konfigurationsdatei des Telefons sichtbar, auf die Sie über die IP-Adresse des Telefons oder über die Konfigurationsberichte des Telefons zugreifen können, die an den Bereitstellungsserver gesendet werden.

Wenn Sie die RFC 8188-basierte Verschlüsselung verwenden, stellen Sie Folgendes sicher:

- Stellen Sie das Telefon mit dem IKM bereit, indem Sie das IKM mit dem XML-Parameter **IKM\_HTTP\_Encrypt\_Content** in der Konfigurationsdatei angeben, die vom Bereitstellungsserver an das Telefon gesendet wird.
- Wenn diese Verschlüsselung auf die vom Bereitstellungsserver an das Telefon gesendeten Konfigurationsdateien angewendet wird, stellen Sie sicher, dass der HTTP-Header der *Inhalts-Codierung* in der Konfigurationsdatei „aes128gcm“ aufweist.

Ohne diesen Header erhält die AES-256-CBC-Methode Vorrang. Ungeachtet des IKM wendet das Telefon die AES-256-CBC-Entschlüsselung an, wenn ein AES-256-Schlüssel in einer Profilregel vorhanden ist.

- Wenn das Telefon diese Verschlüsselung auf die Konfigurationsberichte anwenden soll, die es an den Bereitstellungsserver sendet, stellen Sie sicher, dass kein AES-256-CBC-Schlüssel in der Berichtsregel angegeben ist.

## Optionale Argumente für die Resynchronisierung

Die optionalen Argumente **key**, **uid** und **pwd** können vor den URLs stehen, die in den Profile\_Rule\*-Parametern eingegeben werden, und müssen insgesamt von eckigen Klammern umschlossen sein.

### Taste

Die Option **--key** weist das Telefon darauf hin, dass die Konfigurationsdatei, die es vom Bereitstellungsserver empfängt, mit der AES-256-CBC-Verschlüsselung verschlüsselt ist, es sei denn, der Header *Content-Encoding* in der Datei gibt die Verschlüsselung „aes128gcm“ an. Der Schlüssel selbst wird als Zeichenfolge angegeben, die auf den Begriff **--key** folgt. Der Schlüssel kann optional in Anführungszeichen (") eingeschlossen werden. Das Telefon verwendet den Schlüssel, um die Konfigurationsdatei zu entschlüsseln.

### Beispiele für die Verwendung

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

Für die optionalen Argumente in Klammern wird eine Makroerweiterung durchgeführt. Die speziellen Parameter GPP\_SA bis GPP\_SD werden nur per Makroerweiterung in die Makrovariablen \$SA bis \$SD umgewandelt, wenn sie als key-Optionsargumente verwendet werden. Siehe folgende Beispiele:

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

In Open-Format-Profilen muss das Argument für **--key** dem Argument für die **-k**-Option entsprechen, die **openssl** zugewiesen ist.

### uid und pwd

Die Optionen **uid** und **pwd** können verwendet werden, um die Benutzer-ID und das Kennwort anzugeben, die als Antwort auf HTTP-Standard- und Digest-Authentifizierung gesendet werden, wenn die angegebene URL angefordert wird. Für die optionalen Argumente in Klammern wird eine Makroerweiterung durchgeführt. Die speziellen Parameter GPP\_SA bis GPP\_SD werden nur per Makroerweiterung in die Makrovariablen \$SA bis \$SD umgewandelt, wenn sie als key-Optionsargumente verwendet werden. Siehe folgende Beispiele:

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

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

würde dann erweitert in:

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

# Anwendung eines Profils auf dem Telefon

Nachdem Sie ein XML-Konfigurationsskript erstellt haben, muss es zur Anwendung an das Telefon übertragen werden. Um die Konfiguration zu übernehmen, können Sie die Konfigurationsdatei von einem TFTP-, HTTP- oder HTTPS-Server entweder mithilfe eines Webbrowsers oder mit dem cURL-Befehlszeilen-Utility auf das Telefon herunterladen.

## Die Konfigurationsdatei auf das Telefon von einem TFTP-Server aus herunterladen

Führen Sie die folgenden Schritte aus, um die Konfigurationsdatei für eine TFTP-Serveranwendung auf Ihren PC herunterzuladen.

### Prozedur

---

- Schritt 1** Verbinden Sie Ihren PC mit dem Telefon-LAN.
- Schritt 2** Führen Sie eine TFTP-Serveranwendung auf dem PC aus, und stellen Sie sicher, dass die Konfigurationsdatei im TFTP-Stammverzeichnis verfügbar ist.
- Schritt 3** Geben Sie in einem Webbrowser die LAN-IP-Adresse des Telefons, die IP-Adresse des Computers, den Dateinamen und die Anmeldeinformationen ein. Verwenden Sie das folgende Format:

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

Beispiel:

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

---

## Die Konfigurationsdatei auf das Telefon mit cURL herunterladen

Führen Sie die folgenden Schritte aus, um die Konfiguration auf das Telefon unter Verwendung von cURL herunterzuladen. Dieses Befehlszeilentool wird verwendet, um Daten mit einer URL-Syntax zu übertragen. Hier können Sie cURL herunterladen:

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



---

**Hinweis** Wir empfehlen, dass Sie cURL nicht verwenden, um die Konfiguration auf dem Telefon zu veröffentlichen, da der Benutzername und das Kennwort während der Verwendung von cURL nicht sicher sind.

---

### Prozedur

---

- Schritt 1** Verbinden Sie Ihren PC mit dem LAN-Port des Telefons.
- Schritt 2** Laden Sie die Konfigurationsdatei auf das Telefon herunter, indem Sie den folgenden cURL-Befehl eingeben:

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

## Parameter-Typen für die Bereitstellung

In diesem Abschnitt werden die Bereitstellungsparameter grob nach Funktion sortiert erläutert:

Die folgenden Typen von Bereitstellungsparametern stehen zur Auswahl:

- Allgemeine Dienste
- Wirkung
- Kaufanreize
- Konfigurierbare Zeitpläne
- Profilregeln
- Upgrade-Regel

## Allgemeine Parameter

Die allgemeinen GPP\_\*-Parameter (**Administratoranmeldung** > **Erweitert** > **Sprache** > **Bereitstellung**) werden als freie Zeichenfolgen verwendet, die registriert werden, wenn das Telefon für die Interaktion mit einer bestimmten Bereitstellungsserverlösung konfiguriert wird. Die GPP\_\*-Parameter sind standardmäßig leer. Die Parameter können mit verschiedenen Werten konfiguriert werden:

- Verschlüsselungscodes
- URLs
- Statusinformationen für die mehrstufige Bereitstellung.
- Vorlagen für POST-Anforderungen.
- Zuordnungen von Parameter-Namensaliasen
- Teilweise Zeichenfolgenwerte, die in vollständigen Parameterwerten zusammengefasst werden.

Die GPP\_\*-Parameter stehen für eine Makroerweiterung in anderen Bereitstellungsparametern zur Verfügung. Daher sind Makronamen mit einem Großbuchstaben (A bis P) ausreichend, um den Inhalt der Parameter GPP\_A bis GPP\_P zu ermitteln. Außerdem werden mit den Makronamen mit zwei Großbuchstaben SA bis SD die Parameter GPP\_SA bis GPP\_SD als Sonderfall identifiziert, wenn sie als Argumente der folgenden URL Optionen verwendet werden:

### **key, uid und pwd**

Sie können diese Parameter als Variablen in Bereitstellungs- und Upgrade-Regeln verwenden. Zur Referenzierung wird dem Variablennamen das Zeichen „\$“ vorangestellt, z. B. \$GPP\_A.

## Allgemeine Parameter verwenden

Wenn GPP\_A beispielsweise die Zeichenfolge ABC und GPP\_B die Zeichenfolge 123 enthält, wird für den Ausdruck \$A\$B die Makroerweiterung in ABC123 durchgeführt.

### Vorbereitungen

Greifen Sie auf die Webseite zur Telefonverwaltung zu. Siehe [Auf Weboberfläche des Telefons zugreifen](#).

### Prozedur

- 
- Schritt 1** Wählen Sie **Voice > Bereitstellung** aus.
  - Schritt 2** Blättern Sie zum Abschnitt **Allgemeine Parameter**.
  - Schritt 3** Geben Sie gültige Werte in die Felder GPP A bis GPP P ein.
  - Schritt 4** Klicken Sie auf **Submit All Changes**.
- 

## Parameter aktivieren

Die Parameter Provision\_Enable und Upgrade\_Enable steuern alle Profilresynchronisierungs- und Firmware-Upgrade-Vorgänge. Resynchronisierungen und Upgrades werden unabhängig voneinander gesteuert. Außerdem werden mit diesen Parametern Resynchronisierungs- und Upgrade-URL-Befehle gesteuert, die über den Verwaltungswebserver erteilt werden. Diese beiden Parameter sind standardmäßig auf **Ja** festgelegt.

Der Parameter Resync\_From\_SIP steuert Anforderungen für Resynchronisierungsvorgänge. Ein SIP NOTIFY-Ereignis wird vom Serviceanbieter-Proxyserver an das Telefon gesendet. Wenn aktiviert, kann der Proxy eine Resynchronisierung anfordern. Hierzu sendet der Proxy eine SIP NOTIFY-Nachricht an das Gerät, die das Ereignis zum Resynchronisieren enthält.

Das Gerät gibt auf die Anforderung eine 401-Antwort (Autorisierung für verwendete Anmeldeinformationen abgelehnt) zurück. Das Gerät erwartet eine authentifizierte nachfolgende Anforderung, bevor es die Resynchronisierungsanforderung des Proxy akzeptiert. Mit den Headern Event: reboot\_now und Event: restart\_now werden kalte bzw. warme Neustarts durchgeführt, die ebenfalls geprüft werden.

Die beiden verbleibenden Enable-Parameter lauten Resync\_On\_Reset und Resync\_After\_Upgrade\_Attempt. Diese Parameter bestimmen, ob das Gerät nach dem Neustart der Software und nach jedem Upgrade-Versuch einen Resynchronisierungsvorgang ausführt.

Wenn Resync\_On\_Reset aktiviert ist, führt das Gerät nach dem Startvorgang eine zufällige Verzögerung ein, bevor es zurückgesetzt wird. Die Verzögerung ist eine zufällige Zeitangabe bis zu dem Wert, der für Resync\_Random\_Delay (in Sekunden) angegeben ist. In einem Pool von Telefonen, die gleichzeitig eingeschaltet werden, werden durch diese Verzögerung die Startzeiten der Resynchronisierungsanforderungen der einzelnen Geräte besser verteilt. Diese Funktion kann bei einer großen lokalen Bereitstellung nützlich sein, wenn ein Stromausfall auftritt.

## Kaufanreize

Das Telefon ermöglicht Ihnen die Resynchronisierung in bestimmten Zeitintervallen oder zu einem speziellen Zeitpunkt.

## In bestimmten Zeitintervallen resynchronisieren

Das Telefon ist darauf ausgelegt, regelmäßig eine Resynchronisierung mit dem Bereitstellungsserver durchzuführen. Das Intervall für die Resynchronisierung wird im Parameter `Resync_Periodic` (in Sekunden) konfiguriert. Wenn dieser Wert leer ist, führt das Gerät keine regelmäßigen Resynchronisierungen aus.

Eine erneute Synchronisierung wird normalerweise ausgeführt, wenn die Sprachleitungen inaktiv sind. Wenn eine Sprachleitung aktiv und eine Resynchronisierung fällig ist, verzögert das Telefon die Resynchronisierung, bis die Leitung wieder inaktiv ist. Eine Resynchronisierung kann eine Änderung der Konfigurationsparameter verursachen.

Eine Resynchronisierung kann fehlschlagen, weil das Telefon kein Profil vom Server abrufen kann, die heruntergeladene Datei beschädigt ist oder ein interner Fehler aufgetreten ist. Das Gerät versucht, die Resynchronisierung nach einer Zeitspanne, die in `Resync_Error_Retry_Delay` (in Sekunden) angegeben ist, zu wiederholen. Wenn `Resync_Error_Retry_Delay` auf 0 festgelegt ist, führt das Gerät keine neue Resynchronisierung aus, nachdem eine Resynchronisierung fehlgeschlagen ist.

Wenn ein Upgrade fehlschlägt, wird nach den in `Upgrade_Error_Retry_Delay` angegebenen Sekunden ein erneuter Versuch ausgeführt.

Es stehen zwei konfigurierbare Parameter für die bedingte Auslösung einer Resynchronisierung zur Verfügung: `Resync_Trigger_1` und `Resync_Trigger_2`. Jeder Parameter kann mit einem Bedingungsausdruck programmiert werden, der eine Makroerweiterung durchläuft. Wenn das Intervall für die Resynchronisierung abläuft (Zeit für die nächste Resynchronisierung), verhindern die Auslöser, sofern sie festgelegt sind, die Resynchronisierung, es sei denn, mindestens ein Auslöser wird mit „True“ bewertet.

Die folgende Beispielbedingung löst eine Resynchronisierung aus. Im Beispiel sind seit dem letzten Telefon-Upgrade-Versuch mehr als fünf Minuten (300 Sekunden) und seit dem letzten Resynchronisierungsversuch mindestens 10 Minuten (600 Sekunden) vergangen.

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

## Resynchronisierung zu einem speziellen Zeitpunkt

Der Parameter `Resync_At` ermöglicht es dem Telefon, zu einem bestimmten Zeitpunkt eine Resynchronisierung durchzuführen. Dieser Parameter verwendet das 24-Stunden-Format (hhmm), um die Zeit festzulegen.

Der Parameter `Resync_At_Random_Delay` ermöglicht es dem Telefon, mit einer nicht spezifizierten Verzögerung zu resynchronisieren. Dieser Parameter verwendet ein positives Ganzzahl-Format, um die Zeit festzulegen.

Es sollte vermieden werden, den Server mit Resynchronisierungsanforderungen von mehreren Telefonen zu belasten, deren Resynchronisierung auf dieselbe Zeit festgelegt ist. Aus diesem Grund löst das Telefon die Resynchronisierung bis zu 10 Minuten nach dem angegebenen Zeitpunkt aus.

Wenn Sie die Resynchronisierungszeit beispielsweise auf 1000 (10:00 Uhr) festlegen, löst das Telefon die Resynchronisierung irgendwann zwischen 10:00 Uhr und 10:10 Uhr aus.

Diese Funktion ist standardmäßig deaktiviert. Wenn der Parameter `Resync_At` bereitgestellt wurde, wird der Parameter `Resync_Periodic` ignoriert.

## Konfigurierbare Zeitpläne

Mithilfe der folgenden Bereitstellungsparameter können Sie Zeitpläne für regelmäßige Resynchronisierungen konfigurieren und die Wiederholungsintervalle für Resynchronisierungs- und Upgrade-Fehler angeben:

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

Jeder Parameter akzeptiert einen einzelnen Verzögerungswert (in Sekunden). Die neue erweiterte Syntax ermöglicht eine durch Komma getrennte Liste von aufeinanderfolgenden Verzögerungselementen. Das letzte Element in der Sequenz wird implizit unendlich wiederholt.

Optional können Sie ein Pluszeichen (+) verwenden, um einen anderen numerischen Wert anzugeben, der eine zusätzliche zufällige Verzögerung festlegt.

### Beispiel 1

In diesem Beispiel erfolgt die Resynchronisierung des Telefons in regelmäßigen Abständen alle zwei Stunden. Wenn ein Resynchronisierungsfehler auftritt, erfolgen auf dem Gerät Wiederholungsversuche in den folgenden Intervallen: 30 Minuten, 1 Stunde, 2 Stunden, 4 Stunden. Das Gerät führt weitere Versuche in 4-Stunden-Intervallen durch, bis die Resynchronisierung erfolgreich ist.

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

### Beispiel 2

In diesem Beispiel erfolgt die Resynchronisierung des Geräts in regelmäßigen Abständen jede Stunde (mit einer zusätzlichen zufälligen Verzögerung von bis zu 10 Minuten). Bei einem Resynchronisierungsfehler erfolgen auf dem Gerät Wiederholungsversuche in den folgenden Intervallen: 30 Minuten (plus bis zu 5 Minuten), 1 Stunde (plus bis zu 10 Minuten), 2 Stunden (plus bis zu 15 Minuten). Das Gerät führt weitere Versuche in 2-Stunden-Intervallen durch (plus bis zu 15 Minuten), bis die Resynchronisierung erfolgreich ist.

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

### Beispiel 3

Wenn in diesem Beispiel ein Remote-Upgrade-Versuch fehlschlägt, wiederholt das Gerät das Upgrade nach 30 Minuten, dann wieder nach einer weiteren Stunde und dann nach zwei Stunden. Wenn das Upgrade weiterhin fehlschlägt, versucht es das Gerät alle vier bis fünf Stunden erneut, bis das Upgrade erfolgreich ist.

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

## Profilregeln

Das Telefon bietet mehrere Remote-Konfigurationsprofilparameter (Profile\_Rule\*). Somit können mit jedem Resynchronisierungsvorgang mehrere Dateien abgerufen werden, die von verschiedenen Servern verwaltet werden.

Im einfachsten Fall erfolgt die Resynchronisierung des Geräts regelmäßig mit einem einzelnen Profil auf einem zentralen Server, der alle relevanten internen Parameter aktualisiert. Alternativ kann das Profil auf



verschiedene Dateien aufgeteilt werden. Eine Datei gilt für alle Telefone in einer Bereitstellung. Eine weitere, eindeutige Datei wird für jedes Konto bereitgestellt. Verschlüsselungscodes und Zertifikatinformationen können von einem weiteren Profil bereitgestellt werden, das auf einem separaten Server gespeichert ist.

Wenn eine Resynchronisierung fällig ist, wertet das Telefon die vier Profile\_Rule\*-Parameter nacheinander aus:

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

Jede Auswertung kann dazu führen, dass ein Profil von einem Remote-Bereitstellungsserver abgerufen wird und einige der internen Parameter möglicherweise aktualisiert werden. Wenn eine Auswertung fehlschlägt, wird die Resynchronisierungssequenz unterbrochen und entsprechend den Angaben für den Parameter Resync\_Error\_Retry\_Delay Parameter (in Sekunden) erneut durchgeführt. Wenn alle Auswertungen erfolgreich sind, wartet das Gerät, bis die im Parameter Resync\_Periodic angegebene Zeit erreicht ist, und führt dann eine weitere Resynchronisierung durch.

Der Inhalt der einzelnen Profile\_Rule\*-Parameter besteht aus einer Reihe von Alternativen. Die Alternativen werden durch einen senkrechten Strich | getrennt. Jede Alternative besteht aus einem Bedingungsausdruck, einem Zuweisungsausdruck, einer Profil-URL und allen zugeordneten URL-Optionen. All diese Komponenten sind innerhalb jeder Alternative optional. Im Folgenden sind die zulässigen Kombinationen und die Reihenfolge, in der sie ggf. erscheinen müssen, aufgeführt:

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

Innerhalb jedes Profile\_Rule\*-Parameters müssen alle Alternativen mit Ausnahme der letzten einen Bedingungsausdruck enthalten. Dieser Ausdruck wird ausgewertet und wie folgt verarbeitet:

1. Bedingungen werden von links nach rechts ausgewertet, bis eine gefunden wird, deren Auswertung „True“ ergibt (oder bis eine Alternative ohne Bedingungsausdruck gefunden wird).
2. Alle zugehörigen Zuweisungsausdrücke werden ggf. ausgewertet.
3. Wenn eine URL als Teil dieser Alternative angegeben ist, wird versucht, das Profil herunterzuladen, das sich unter der angegebenen URL befindet. Das System versucht, die internen Parameter entsprechend zu aktualisieren.

Wenn alle Alternativen über Bedingungsausdrücke verfügen und keine Auswertung „True“ ergibt (oder wenn die gesamte Profilregel leer ist), wird der gesamte Profile\_Rule\*-Parameter übersprungen. Der nächste Profilregelparameter in der Sequenz wird ausgewertet.

### Beispiel 1

In diesem Beispiel erfolgt eine unbedingte Resynchronisierung mit dem Profil unter der angegebenen URL, und es wird eine HTTP GET-Anforderung an den Remote-Bereitstellungsserver gesendet:

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

**Beispiel 2**

In diesem Beispiel erfolgt die Resynchronisierung des Geräts mit zwei unterschiedlichen URLs, abhängig vom Registrierungsstatus der Leitung 1. Im Falle einer verlorenen Registrierung führt das Gerät eine HTTP POST-Anforderung an ein CGI-Skript durch. Das Gerät sendet den Inhalt des makroerweiterten GPP\_A-Parameters, der zusätzliche Informationen zum Gerätestatus enthalten kann:

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

**Beispiel 3**

In diesem Beispiel erfolgt die Resynchronisierung des Geräts mit demselben Server. Das Gerät bietet zusätzliche Informationen, wenn kein Zertifikat auf der Einheit installiert ist (für ältere Einheiten vor 2.0):

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

**Beispiel 4**

In diesem Beispiel ist Leitung 1 deaktiviert, bis GPP\_A über die erste URL auf einen Wert gleich „Provisioned“ gesetzt wird. Anschließend erfolgt die Resynchronisierung mit der zweiten URL:

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

**Beispiel 5**

In diesem Beispiel wird angenommen, dass das Profil, das vom Server zurückgegeben wird, XML-Element-Tags enthält. Diese Tags müssen mithilfe der Alias-Zuordnung, die in GPP\_B gespeichert ist, erneut den entsprechenden Parameternamen zugeordnet werden:

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

Eine Resynchronisierung wird normalerweise als fehlgeschlagen betrachtet, wenn ein angefordertes Profil vom Server nicht empfangen wird. Der Parameter Resync\_Fails\_On\_FNF kann dieses Standardverhalten überschreiben. Wenn Resync\_Fails\_On\_FNF auf „No“ festgelegt ist, akzeptiert das Gerät eine Datei-nicht-gefunden-Antwort vom Server als erfolgreiche Resynchronisierung. Der Standardwert für Resync\_Fails\_On\_FNF lautet „Yes“.

## Upgrade-Regel

Mithilfe der Upgrade-Regel wird das Gerät angewiesen, eine neue Software zu aktivieren, und ggf. informiert, wo diese Software abgerufen werden kann. Wenn die Software bereits auf dem Gerät vorhanden ist, versucht es nicht, sie abzurufen. Die Gültigkeit des Software-Speicherorts ist demnach nicht von Bedeutung, wenn sich die gewünschte Software auf der inaktiven Partition befindet.

Mit dem Parameter Upgrade\_Rule wird eine Firmware angegeben, die, wenn sie sich von der aktuellen Firmware unterscheidet, heruntergeladen und angewendet wird, sofern dies nicht durch einen Bedingungsausdruck verhindert wird oder Upgrade\_Enable auf **No** festgelegt wurde.

Das Telefon umfasst einen konfigurierbaren Remote-Upgrade-Parameter: Upgrade\_Rule. Dieser Parameter akzeptiert eine ähnliche Syntax wie die Profilregelparameter. URL-Optionen werden für Upgrades nicht unterstützt, aber Bedingungsausdrücke und Zuweisungsausdrücke können verwendet werden. Wenn Bedingungsausdrücke verwendet werden, können für den Parameter mehrere Alternativen, getrennt durch das |-Zeichen, angegeben werden. Die Syntax für die einzelnen Alternativen lautet wie folgt:

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

Wie bei den Profile\_Rule\*-Parametern werden mit dem Parameter Upgrade\_Rule alle Alternativen ausgewertet, bis ein Bedingungsausdruck erfüllt ist oder eine Alternative keinen Bedingungsausdruck aufweist. Der zugehörige Zuweisungsausdruck wird ausgewertet, sofern er angegeben wurde. Anschließend wird versucht, ein Upgrade über die angegebene URL durchzuführen.

Wenn Upgrade\_Rule eine URL ohne einen Bedingungsausdruck enthält, wird das Gerät auf das Firmware-Image aktualisiert, das mit der URL angegeben wird. Nach der Makroerweiterung und Auswertung der Regel versucht das Gerät so lange nicht erneut, ein Upgrade durchzuführen, bis die Regel oder die effektive Kombination von scheme + server + port + filepath geändert wurde.

Um zu versuchen, ein Firmware-Upgrade durchzuführen, wird zu Beginn des Vorgangs die Audiofunktion auf dem Gerät deaktiviert, und am Ende des Vorgangs wird das Gerät neu gestartet. Das Gerät führt nur dann automatisch ein Upgrade durch, das durch den Inhalt von Upgrade\_Rule gesteuert wird, wenn alle Sprachleitungen derzeit inaktiv sind.

Beispiel:

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

In diesem Beispiel wird mit Upgrade\_Rule die Firmware auf das Image aktualisiert, das unter der angegebenen URL gespeichert ist.

Hier ein weiteres Beispiel:

```
("${F}" 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
```

In diesem Beispiel wird das Gerät angewiesen, basierend auf den Inhalten des allgemeinen Parameters GPP\_F eines von zwei Images zu laden.

Das Gerät kann ein Downgrade-Limit bezüglich der Firmware-Versionsnummer erzwingen; dies kann eine nützliche Anpassungsoption sein. Wenn eine gültige Firmware-Versionsnummer im Parameter Downgrade\_Rev\_Limit konfiguriert ist, weist das Gerät Upgrade-Versuche für Firmware-Versionen vor dem angegebenen Grenzwert zurück.

## Datentypen

Folgende Datentypen werden mit Konfigurationsprofilparametern verwendet:

- {a,b,c,...} – Wahlmöglichkeit zwischen a, b, c...
- Bool – Boolescher Wert „Ja“ oder „Nein“
- CadScript – Miniskript, mit dem die Rhythmusparameter eines Signals angegeben werden. Bis zu 127 Zeichen

Syntax: S<sub>1</sub>[;S<sub>2</sub>], wobei Folgendes gilt:

- $S_i = D_i(\text{on}_{i,1}/\text{off}_{i,1}[\text{on}_{i,2}/\text{off}_{i,2}[\text{on}_{i,3}/\text{off}_{i,3}[\text{on}_{i,4}/\text{off}_{i,4}[\text{on}_{i,5}/\text{off}_{i,5}[\text{on}_{i,6}/\text{off}_{i,6}]]]]])$ . Dies wird als Abschnitt (section) bezeichnet.
- $\text{on}_{i,j}$  und  $\text{off}_{i,j}$  stehen für die Dauer der Aktivität/Inaktivität in Sekunden eines *Segments*.  $i = 1$  oder  $2$  und  $j = 1$  bis  $6$ .
- $D_i$  ist die Gesamtdauer des Abschnitts in Sekunden.

Sie können die Zeitintervalle mit jeweils bis zu drei Dezimalstellen angeben, sodass sie bis auf die Millisekunde genau sind. Der Platzhalter „\*“ steht für Endlosdauer. Die Segmente innerhalb der einzelnen Abschnitte werden der Reihe nach wiedergegeben und wiederholt, bis die Gesamtdauer erreicht ist.

Beispiel 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
```

Beispiel 2 – Eindeutiger Rufplan (kurz, kurz, kurz, lang):

```
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 – Skriptersyntax, die zum Angeben der Rufnummernpläne für Leitung 1 und Leitung 2 verwendet wird.
- Float<n> – Gleitkommawert mit bis zu n Dezimalstellen.
- FQDN – Vollständiger Domänenname. Kann bis zu 63 Zeichen enthalten. Im Folgenden finden Sie einige Beispiele:
  - sip.Cisco.com:5060 oder 109.12.14.12:12345
  - sip.Cisco.com oder 109.12.14.12
- FreqScript – Miniskript, mit dem die Frequenz- und Pegelparameter eines Tons angegeben werden. Enthält bis zu 127 Zeichen.
 

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

  - $F_1$ –  $F_6$  für die Frequenz in Hz stehen (nur Ganzzahlen ohne Vorzeichen).
  - $L_1$ –  $L_6$  sind entsprechende Pegel in dBm (mit bis zu einer Dezimalstelle).

Leerzeichen vor und nach dem Komma sind erlaubt, werden jedoch nicht empfohlen.

Beispiel 1 – Ton für wartenden Anruf:

```
440@-10

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

Beispiel 2 – Wählton:

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

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

- IP – Gültige IPv4-Adresse in der Form x.x.x.x, wobei x zwischen 0 und 255 liegt. Beispiel: 10.1.2.100.
- UserID – Benutzer-ID, wie sie in einer URL angezeigt wird; bis zu 63 Zeichen.
- Phone – Eine Zeichenfolge für die Telefonnummer, z. B. 14081234567, \*69, \*72, 345678; oder eine allgemeine URL, wie z. B. 1234@10.10.10.100:5068 oder jsmith@Cisco.com. Die Zeichenfolge kann bis zu 39 Zeichen enthalten.
- PhTmpl – Telefonnummernvorlage. Jede Vorlage kann eines oder mehrere Muster enthalten, die durch ein Komma (,) voneinander getrennt sind. Leerzeichen zu Beginn jedes Musters werden ignoriert. „?“ und „\*“ sind Platzhalter. Verwenden Sie zur tatsächlichen Darstellung dieser Zeichen %xx. Beispiel: %2a steht für \*. Die Vorlage kann bis zu 39 Zeichen enthalten. Beispiele: „1408\*, 1510\*“, „1408123????“, 555?1.“.
- Port – TCP-/UDP-Portnummer (0-65535). Kann im Dezimal- oder Hexadezimalformat angegeben werden.
- ProvisioningRuleSyntax – Skriptersyntax, mit der Regeln für die Konfigurationsresynchronisierung und Firmware-Upgrades definiert werden.
- PwrLevel – Leistungspegel in dBm mit einer Dezimalstelle, z. B. -13,5 oder 1,5 (dBm).
- RscTmpl – Vorlage des SIP-Antwort-Statuscodes, z. B. „404, 5\*“, „61?“, „407, 408, 487, 481“. Kann bis zu 39 Zeichen enthalten.
- Sig<n> – n-Bit-Wert mit Vorzeichen. Kann im Dezimal- oder Hexadezimalformat angegeben werden. Vor negativen Werten muss ein „-“-Zeichen stehen. Ein „+“-Zeichen vor positiven Werten ist optional.
- Star Codes – Aktivierungscode für einen zusätzlichen Dienst, z. B. \*69. Der Code kann bis zu 7 Zeichen enthalten.
- Str<n> – Allgemeine Zeichenfolge mit bis zu n nicht reservierten Zeichen.
- Time<n> – Zeitdauer in Sekunden mit bis zu n Dezimalstellen. Zusätzlich angegebene Dezimalstellen werden ignoriert.
- ToneScript – Miniskript, mit dem die Frequenz-, Pegel- und Rhythmusparameter eines Anrufstatus-Tons angegeben werden. Das Skript darf maximal 127 Zeichen enthalten.

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

Der Abschnitt  $Z_1$  ähnelt dem Abschnitt  $S_1$  in einem CadScript, allerdings folgt auf jedes Ein-/Aus-Segment ein Frequenzkomponenten-Parameter:  $Z_1 = D_1(\text{on}_{i,1}/\text{off}_{i,1}/f_{i,1} [., \text{on}_{i,2}/\text{off}_{i,2}/f_{i,2} [., \text{on}_{i,3}/\text{off}_{i,3}/f_{i,3} [., \text{on}_{i,4}/\text{off}_{i,4}/f_{i,4} [., \text{on}_{i,5}/\text{off}_{i,5}/f_{i,5} [., \text{on}_{i,6}/\text{off}_{i,6}/f_{i,6}]]]]])$ . Dabei ist:

- $f_{i,j} = n_1[+n_2]+n_3[+n_4[+n_5[+n_6]]]]]$ .
- $1 < n_k < 6$  gibt die Frequenzkomponenten im FreqScript an, die in diesem Segment verwendet werden.

Wenn mehr als eine Frequenzkomponente in einem Segment verwendet wird, werden die Komponenten zusammengezählt.

Beispiel 1 – Wählton:

```
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
```

Beispiel 2 – Unterbrochener Rufton:

```
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
```

- $\text{Uns}\langle n \rangle$  – n-Bit-Wert ohne Vorzeichen, wobei  $n = 8, 16$  oder  $32$ . Kann im Dezimal- oder Hexadezimalformat angegeben werden, z. B.  $12$  oder  $0x18$ , solange der Wert in die n-Bits passt.



**Hinweis** Beachten Sie Folgendes:

- <Par Name> steht für den Namen eines Konfigurationsparameters. In einem Profil wird das entsprechende Tag gebildet, indem das Leerzeichen durch einen Unterstrich „\_“ ersetzt wird, z. B. **Par\_Name**.
- Ein leeres Standardwert-Feld impliziert eine leere Zeichenfolge <“”>.
- Das Telefon verwendet weiterhin die zuletzt konfigurierten Werte für Tags, die in einem bestimmten Profil nicht vorhanden sind.
- Vorlagen werden in der angegebenen Reihenfolge verglichen. Die erste, *nicht die beste*, Übereinstimmung wird ausgewählt. Der Parametername muss genau übereinstimmen.
- Wenn mehr als eine Definition für einen Parameter in einem Profil angegeben ist, wird die letzte entsprechende Definition in der Datei auf dem Telefon verwendet.
- Durch eine Parameterspezifikation mit einem leeren Parameterwert wird erzwungen, dass der Parameter auf den Standardwert zurückgesetzt wird. Um stattdessen eine leere Zeichenfolge anzugeben, verwenden Sie die leere Zeichenfolge „\_“ als Parameterwert.

## Profil-Updates und Firmware-Upgrades

Das Telefon unterstützt die sichere Remotebereitstellung (Konfiguration) und Firmware-Upgrades. Ein nicht konfiguriertes Telefon kann ein speziell für dieses Gerät entwickeltes, verschlüsseltes Profil empfangen. Aufgrund eines sicheren erstmaligen Bereitstellungsmechanismus, bei dem die SSL-Funktionalität verwendet wird, benötigt das Telefon keinen expliziten Schlüssel.

Ein Benutzereingriff ist nicht erforderlich, um ein Profil-Update oder ein Firmware-Upgrade zu starten oder durchzuführen, oder wenn zwischenzeitliche Upgrades erforderlich sind, um eine ältere Version auf einen aktuellen Upgrade-Status zu aktualisieren. Es wird nur dann versucht, eine Profilsynchronisierung durchzuführen, wenn das Telefon inaktiv ist, da eine Resynchronisierung einen Neustart der Software auslösen und einen Anruf beenden kann.

Der Bereitstellungsprozess wird durch allgemeine Parameter verwaltet. Jedes Telefon kann so konfiguriert werden, dass es regelmäßig einen NPS kontaktiert. Kommunikation mit dem NPS erfordert kein sicheres Protokoll, da das aktualisierte Profil mit einem Shared-Secret-Schlüssel verschlüsselt wird. Der NPS kann ein standardmäßiger TFTP-, HTTP- oder HTTPS-Server mit Client-Zertifikaten sein.

Der Administrator kann Telefone über die Webbenutzeroberfläche des Telefons aktualisieren, neu starten oder resynchronisieren. Diese Aufgaben können auch mithilfe einer SIP NOTIFY-Benachrichtigung ausgeführt werden.

Konfigurationsprofile werden mithilfe von gängigen Open-Source-Tools generiert, die sich in Bereitstellungssysteme von Serviceanbietern integrieren lassen.

## Allow Profile Updates (Profilaktualisierungen zulassen)

Profil-Updates können in bestimmten Intervallen ermöglicht werden. Aktualisierte Profile werden von einem Server über TFTP, HTTP oder HTTPS an das Telefon gesendet.

Sie können die Parameter auch in der Konfigurationsdatei des Telefons mit XML-Code (cfg.xml) konfigurieren.

**Vorbereitungen**

Greifen Sie auf die Webseite zur Telefonverwaltung zu. Siehe [Auf Weboberfläche des Telefons zugreifen](#).

**Prozedur****Schritt 1**

Wählen Sie **Sprache** > **Bereitstellung** aus.

**Schritt 2**

Im Abschnitt **Konfigurationsprofil** wählen Sie **Ja** aus dem Parameter **Bereitstellung aktivieren** aus.

Sie können diesen Parameter in der XML-Konfigurationsdatei (cfg.xml) des Telefons konfigurieren, indem Sie eine Zeichenfolge in folgendem Format eingeben:

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

Standard: Ja

**Schritt 3**

Legen Sie die Parameter wie in der Tabelle [Profil Resync-Parameter](#) beschrieben fest.

**Schritt 4**

Klicken Sie auf **Submit All Changes**.

## Zulassen und Konfigurieren von Firmware-Upgrades

Firmware-Updates können in bestimmten Intervallen ermöglicht werden. Aktualisierte Firmware wird von einem Server über TFTP oder HTTP an das Telefon gesendet. Die Sicherheit ist bei Firmware-Upgrades ein zu vernachlässigendes Problem, da Firmware keine persönlichen Daten enthält.

Sie können die Parameter auch in der Konfigurationsdatei des Telefons mit XML-Code (cfg.xml) konfigurieren.

**Vorbereitungen**

Greifen Sie auf die Webseite zur Telefonverwaltung zu. Siehe [Auf Weboberfläche des Telefons zugreifen](#).

**Prozedur****Schritt 1**

Wählen Sie **Voice** > **Bereitstellung** aus.

**Schritt 2**

Im Abschnitt **Firmware-Upgrade** wählen Sie **Ja** aus dem Parameter **Upgrade aktivieren** aus.

Sie können diesen Parameter in der XML-Konfigurationsdatei (cfg.xml) des Telefons konfigurieren, indem Sie eine Zeichenfolge in folgendem Format eingeben:

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

Optionen: Ja und Nein

Standard: Ja

**Schritt 3**

Legen Sie den Parameter **Upgrade bei Wiederholungsverzögerung bei Fehler** in Sekunden fest.

Das Upgrade-Intervall (in Sekunden), das nach einem fehlgeschlagenen Upgrade angewendet wird. Das Gerät verfügt über einen Timer für fehlgeschlagene Firmware-Upgrades. Dieser wird nach einem fehlgeschlagenen Firmware-Upgrade-Versuch aktiviert. Der Wert dieses Parameters dient zur Initialisierung des Timers. Der nächste Firmware-Upgrade-Versuch erfolgt, wenn der Timer bei 0 angelangt ist.



Sie können diesen Parameter in der XML-Konfigurationsdatei (cfg.xml) des Telefons konfigurieren, indem Sie eine Zeichenfolge in folgendem Format eingeben:

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

Standard: 3600

:

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

#### Schritt 4

Legen Sie den Parameter **Upgrade-Regel** fest, indem Sie ein Firmware-Upgrade-Skript eingeben, das die Upgrade-Bedingungen und zugehörige Firmware-URLs definiert. Das Skript verwendet die gleiche Syntax wie die Profilregel. Geben Sie ein Skript ein und verwenden die folgendes Format, um die Upgrade-Regel einzugeben:

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

Zum Beispiel:

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

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

Sie können diesen Parameter in der XML-Konfigurationsdatei (cfg.xml) des Telefons konfigurieren, indem Sie eine Zeichenfolge in folgendem Format eingeben:

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

#### Schritt 5

Klicken Sie auf **Submit All Changes**.

## Upgrade der Firmware mit TFTP, HTTP oder HTTPS

Das Telefon unterstützt Firmware-Upgrades über TFTP, HTTP oder HTTPS.



**Hinweis** Das Zurücksetzen auf frühere Versionen ist möglicherweise nicht für alle Geräte verfügbar. Weitere Informationen finden Sie in den Versionshinweisen für Ihr Telefon und Ihre Firmware-Version.

### Vorbereitungen

Die Firmware-Datei muss auf einen verfügbaren Server heruntergeladen werden.

### Prozedur

#### Schritt 1

Benennen Sie das Image wie folgt um:

```
cp-x8xx-sip.aa-b-cMPP.cop in cp-x8xx-sip.aa-b-cMPP.tar.gz
```

Dabei gilt:

**x8xx** ist die Telefonserie, z. B. 7811 oder 7832.

**aa-b-c** ist die Versionsnummer, z. B. 10-4-1.

- Schritt 2** Verwenden Sie den Befehl `tar -xzvf`, um den Tarball zu entpacken.
- Schritt 3** Kopieren Sie den Ordner in ein TFTP-, HTTP- oder HTTPS-Downloadverzeichnis.
- Schritt 4** Greifen Sie auf die Webseite zur Telefonverwaltung zu. Siehe [Auf Weboberfläche des Telefons zugreifen](#).
- Schritt 5** Wählen Sie **Sprache** > **Bereitstellung** aus.
- Schritt 6** Suchen Sie den Dateinamen der Software, der mit **.loads** endet, und fügen Sie diesen der gültigen URL hinzu.
- Schritt 7** Klicken Sie auf **Submit All Changes**.
- 

## Upgrade der Firmware mit einem Browserbefehl

Ein in die Adressleiste des Browsers eingegebener Upgrade-Befehl kann verwendet werden, auf die Firmware auf einem Telefon zu aktualisieren. Das Telefon wird nur aktualisiert, wenn es inaktiv ist. Es wird automatisch versucht, das Update durchzuführen, wenn ein Anruf abgeschlossen ist.

### Prozedur

---

Geben Sie den folgenden Befehl ein, um das Telefon mit einer URL in einem Webbrowser zu aktualisieren:

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

---