



프로비저닝 형식

- 구성 프로파일, 1 페이지
- 구성 프로파일 형식, 1 페이지
- 공개 프로파일(XML) 압축 및 암호화, 5 페이지
- 프로파일을 전화기에 적용, 12 페이지
- 프로비저닝 매개 변수 유형, 13 페이지
- 데이터 유형, 19 페이지
- 프로파일 업데이트 및 펌웨어 업그레이드, 23 페이지

구성 프로파일

전화기는 XML 형식의 구성을 허용합니다.

이 문서의 예에서는 XML 형식(XML) 구문의 구성 프로파일을 사용합니다.

전화기에 대한 자세한 내용은 해당 장치의 관리 설명서를 참조하십시오. 각 설명서에는 관리 웹 서버를 통해 구성할 수 있는 매개 변수의 설명이 나옵니다.

구성 프로파일 형식

구성 프로파일은 전화기에 대한 매개 변수 값을 정의합니다.

구성 프로파일 XML 형식은 표준 XML 제작 도구를 사용하여 매개 변수와 값을 컴파일합니다.



참고 UTF-8 문자 집합만 지원됩니다. 편집기에서 프로파일을 수정하려면 인코딩 형식을 변경하지 마십시오. 그렇지 않으면 전화기에서 파일을 인식할 수 없습니다.

각 전화기마다 기능이 다르므로 매개 변수 집합도 다릅니다.

XML 형식(XML) 프로파일

공개 형식 프로파일은 요소 특성과 값을 포함하는 요소의 계층으로 XML과 비슷한 구문을 사용하는 텍스트 파일입니다. 이 형식은 표준 도구를 사용하여 구성 파일을 만들 수 있게 해줍니다. 이 형식의 구성 파일은 재동기화 작업을 수행하는 동안 프로비저닝 서버에서 전화기로 전송할 수 있습니다. 이 파일은 이진 개체로 컴파일하지 않고 전송할 수 있습니다.

전화기는 표준 도구가 생성하는 구성 형식을 받을 수 있습니다. 이 기능 덕분에 기존 데이터베이스를 바탕으로 구성 프로파일을 생성하는 백엔드 프로비저닝 서버 소프트웨어를 쉽게 개발할 수 있습니다.

구성 파일의 기밀 정보를 보호하기 위해, 프로비저닝 서버는 TLS로 보호되는 채널을 통해 이 유형의 파일을 전화기로 전달합니다. 선택적으로 gzip deflate 알고리즘(RFC1951)을 사용하여 파일을 압축할 수 있습니다.

파일은 다음 암호화 방법 중 하나로 암호화될 수 있습니다.

- AES-256-CBC 암호화
- AES-128-GCM 암호화를 사용한 RFC-8188 기반 HTTP 콘텐츠 암호화

예: 공개 프로파일 형식

```
<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>
```

<flat-profile> 요소 태그는 전화기가 인식하는 모든 파라미터 요소를 묶습니다.

구성 파일의 구성 요소

구성 파일은 다음과 같은 구성 요소를 포함할 수 있습니다.

- 요소 태그
- 특성
- 파라미터
- 서식 기능
- XML 주석

요소 태그 속성

- XML 프로비저닝 형식과 웹 UI는 동일한 설정의 구성을 허용합니다. 웹 UI의 XML 태그 이름과 필드 이름은 비슷하지만 XML 요소 이름 제한 때문에 다릅니다. 예를 들어, "" 대신 밑줄(_)을 사용합니다.

- 전화기는 특수 <flat-profile> 요소로 캡슐화한 유효한 파라미터명을 가진 요소를 인식합니다.
- 요소 이름은 꺾쇠 괄호 안에 넣습니다.
- 장치에 대한 관리 웹 페이지에서 대부분의 요소 이름은 다음과 같은 수정 사항을 제외하고 필드 이름과 비슷합니다.
 - 요소 이름은 공백 또는 특수 문자를 포함할 수 없습니다. 관리 웹 필드 이름에서 요소 이름을 유추하려면 공백이나 특수 문자 [,], (,), /를 밑줄로 대체하면 됩니다.
 - 예: <Resync_On_Reset> 요소는 재설정 시 재동기화 필드에 해당합니다.
 - 각 요소 이름은 고유해야 합니다. 관리 웹 페이지에서 회선, 사용자 및 내선 번호 페이지와 같은 여러 웹 페이지에 같은 필드가 나타날 수 있습니다. 페이지 탭에 표시된 번호를 지정하려면 요소 이름에 [n]을 추가합니다.
 - 예: <Dial_Plan_1_> 요소는 회선 1에 대한 다이얼 플랜을 나타냅니다.
- 각 여는 요소 태그에는 해당하는 닫는 요소 태그가 있어야 합니다. 예:

```
<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>
```

- 요소 태그는 대소문자를 구분합니다.
- 빈 요소 태그를 사용할 수 있으며 빈 값으로 구성하는 것으로 해석됩니다. 여는 요소 태그를 해당하는 요소 태그 없이 입력하고, 닫는 꺾쇠 괄호(>) 앞에 공백과 슬래시를 삽입합니다. 이 예에서 Profile Rule B는 비어 있습니다.

```
<Profile_Rule_B />
```

- 빈 요소 태그는 사용자가 제공한 값을 재동기화 작업 중에 덮어쓰지 않도록 예방하는 데 사용할 수 있습니다. 다음 예에서 사용자 단축 다이얼 설정은 변경되지 않습니다.

```
<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>
```

- 해당 매개 변수를 빈 문자열로 설정하려면 빈 값을 사용합니다. 여는 요소와 닫는 요소를 중간에 값 없이 입력합니다. 다음 예에서 GPP_A 매개 변수는 빈 문자열로 설정됩니다.

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

- 인식되지 않는 요소 이름은 무시됩니다.

프로퍼티 속성

다음과 같은 속성이 매개 변수에 적용됩니다.

- 프로파일에서 지정되지 않은 매개 변수는 전화기에서 변경되지 않습니다.
- 인식되지 않는 매개 변수는 무시됩니다.
- 공개 형식 프로파일에 동일한 매개 변수 태그가 여러 번 포함되는 경우 마지막 항목이 이전 항목을 재정의합니다. 매개 변수의 구성 값이 실수로 재정의되지 않게 하려면 각 프로파일에서 매개 변수의 인스턴스를 최대 하나씩 지정하는 것이 좋습니다.
- 마지막으로 처리된 프로파일이 우선합니다. 여러 프로파일이 동일한 구성 매개 변수를 지정하는 경우 나중 프로파일의 값이 우선합니다.

문자열 형식

다음과 같은 속성이 문자열의 서식에 적용됩니다.

- 주석은 표준 XML 구문으로 사용할 수 있습니다.


```
<!-- My comment is typed here -->
```
- 선행 및 후행 공백은 가독성을 위해 사용할 수 있지만 매개 변수 값에서 제거됩니다.
- 값 내의 줄바꿈은 공백으로 변환됩니다.
- <? ?> 형식의 XML 헤더는 허용되지만 전화기에서 무시합니다.
- 특수 문자를 입력하려면 다음 표에 나오는 기본 XML 문자 이스케이프를 사용합니다.

특수 문자	XML 이스케이프 시퀀스
앰퍼샌드(&)	&
<(보다 작음)	<
>(보다 큼)	>
'(아포스트로피)	'

특수 문자	XML 이스케이프 시퀀스
"(큰따옴표)	";

다음 예에서는 다이얼 플랜 규칙에 필요한 보다 큼 기호(>)와 보다 작음 기호(<)를 나타내기 위해 문자 이스케이프를 입력했습니다. 다음 예에서는 <Dial_Plan_1_> 파라미터를 (S0 <:18005551212>)와 동일하게 설정하는 정보 핫라인 다이얼 플랜을 정의합니다(관리자 로그인 > 고급 > 음성 > 내선 전화(n)).

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

- 십진수 및 16진수 값을 사용한 숫자 캐릭터 이스케이프(예: (및 .)는 변환됩니다.
- 전화기 펌웨어는 ASCII 문자만 지원합니다.

공개 프로파일(XML) 압축 및 암호화

공개 구성 프로파일을 압축하여 프로비저닝 서버에서 네트워크 부하를 줄일 수 있습니다. 프로파일을 암호화하여 기밀 정보를 보호할 수도 있습니다. 압축은 필수가 아니지만 암호화하려면 먼저 압축해야 합니다.

공개 프로파일 압축

지원되는 압축 방법은 **gzip deflate** 알고리즘(RFC1951)입니다. **gzip** 유틸리티 및 이와 동일한 알고리즘을 구현하는 압축 라이브러리(zlib)는 인터넷 사이트에서 구할 수 있습니다.

전화기는 압축을 식별하기 위해 압축된 파일이 **gzip** 호환 헤더를 포함한다고 예상합니다. 원래 공개 프로파일에서 **gzip** 유틸리티를 호출하면 헤더가 생성됩니다. 전화기는 다운로드한 파일 헤더를 검사하여 파일 형식을 확인합니다.

예를 들어 `profile.xml`이 올바른 프로파일이라면 `profile.xml.gz` 파일도 사용할 수 있습니다. 다음 중 한 명령으로 이 프로파일 유형을 생성할 수 있습니다.

- `>gzip profile.xml`
원래 파일을 압축된 파일로 대체합니다.
- `>cat profile.xml | gzip > profile.xml.gz`
원래 파일을 그대로 두고 새 압축 파일을 생성합니다.

압축에 대한 자습서는 [Gzip으로 공개 프로파일 압축](#) 섹션에 있습니다.

공개 프로파일 암호화

대칭 키 암호화로 파일의 압축 여부에 관계없이 공개 구성 프로파일을 암호화할 수 있습니다. 압축(적용되는 경우)은 암호화 전에 적용되어야 합니다.

프로비저닝 서버는 HTTPS를 사용하여 구축 후 전화기의 초기 프로비저닝을 처리할 수 있습니다. 구성 프로파일을 오프라인으로 사전 암호화하면 프로파일 재동기화에 HTTP를 사용할 수 있습니다. 이 기능으로 대규모 구축에서 HTTPS 서버의 부하를 줄일 수 있습니다.

전화기는 구성 파일에 대해 두 가지 암호화 방법을 지원합니다.

- AES-256-CBC 암호화
- AES-128-GCM 암호화를 사용한 RFC-8188 기반 HTTP 콘텐츠 암호화

키 또는 IKM(Input Keating Material)을 먼저 장치에 프로비저닝해야 합니다. 비밀 키의 부트스트랩은 HTTPS를 사용하여 안전하게 할 수 있습니다.

구성 파일 이름에는 특정 형식이 필요하지 않지만, .cfg 확장명으로 끝나는 파일은 일반적으로 구성 프로파일을 나타냅니다.

AES-256-CBC 암호화

전화기는 구성 파일에 대해 AES-256-CBC 암호화를 지원합니다.

다양한 인터넷 사이트에서 다운로드할 수 있는 OpenSSL 암호화 도구로 암호화를 수행할 수 있습니다. 256비트 AES 암호화를 지원하려면 AES 코드 활성화를 위해 도구의 재검파일이 필요할 수 있습니다. 펌웨어는 버전 openssl-1.1.1d를 대상으로 테스트되었습니다.

[OpenSSL로 프로파일 암호화](#) 에서 암호화에 대한 자습서를 볼 수 있습니다.

프로파일은 암호화된 파일이 다음 명령을 사용하여 생성된 것과 동일한 형식의 파일일 것으로 예상합니다.

```
# 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
```

비밀 키는 어떤 일반 텍스트도 될 수 있으며, 앞의 소문자 -k는 임의의 64비트 솔트를 생성하는 데 사용됩니다. 비밀 키에 -k 인수가 사용되면 암호화 도구가 임의의 128비트 초기 벡터와 실제 256비트 암호화 키를 만듭니다.

이러한 형식의 암호화가 구성 프로파일에 사용되면, 파일의 암호를 해독하기 위해 전화기에 비밀 키 값을 전달해야 합니다. 이 값은 URL 프로파일 내의 한정자로 지정됩니다. 구문은 명시적 URL을 사용하며 다음과 같습니다.

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

이 값은 Profile_Rule 매개 변수 중 하나를 사용하여 프로그래밍됩니다.

매크로 확장

여러 프로비저닝 매개 변수는 평가되기 전에 내부적으로 매크로 확장을 거칩니다. 이 사전 평가 단계는 전화기 재동기화 및 업그레이드 작업에 큰 유연성을 제공합니다.

평가 전에 매크로 확장을 거치는 매개 변수 그룹은 다음과 같습니다.

- Resync_Trigger_*
- Profile_Rule*
- Log_xxx_Msg
- Upgrade_Rule

[선택적 재동기화 인수, 11 페이지](#)에서 설명한 것처럼, 특정한 조건에서는 일부 일반 목적 매개 변수 (GPP_*)도 매크로 확장을 거칩니다.

매크로 확장을 거치는 동안 \$NAME 및 \$(NAME) 형식의 식을 명명된 변수의 내용으로 대체합니다. 이러한 변수에는 일반 목적 매개 변수, 여러 제품 식별자, 특정 이벤트 타이머 및 프로비저닝 상태 값이 포함됩니다. 전체 목록은 [매크로 확장 변수](#)에 있습니다.

다음 예에서는 MAC 주소 000E08012345를 삽입하기 위해 식 \$(MAU)를 사용합니다.

관리자 입력: `$(MAU) config.cfg`

장치의 MAC 주소 000E08012345로 매크로 확장을 거친 결과: 000E08012345config.cfg

매크로 이름이 인식되지 않으면 확장되지 않고 유지됩니다. 예를 들어 STRANGE는 유효한 매크로 이름으로 인식되지 않지만 MAU는 유효한 매크로 이름으로 인식됩니다.

관리자 입력: `$(STRANGE)$MAU.cfg`

장치의 MAC 주소 000E08012345로 매크로 확장을 거친 결과: \$(STRANGE)000E08012345.cfg

매크로 확장은 재귀적으로 적용되지 않습니다. 예를 들어 \$\$MAU"는 \$MAU"로 확장되지만(\$가 확장됨), MAC 주소로 확장되지는 않습니다.

특수 목적 매개 변수 GPP_SA ~ GPP_SD의 내용은 매크로 확장 \$\$SA ~ \$\$SD로 매핑됩니다. 이러한 파라미터는 재동기화 URL의 --key, --uid, --pwd 옵션의 인수로 사용될 때만 매크로 확장됩니다.

조건식

조건식은 재동기화 이벤트를 트리거할 수 있으며 재동기화 및 업그레이드 작업의 대체 URL을 선택할 수 있습니다.

조건식은 **and** 연산자로 구분한 비교의 목록으로 구성됩니다. 모든 비교가 충족되어야 조건이 참이 됩니다.

각 비교는 세 가지 리터럴 유형 중 하나를 사용할 수 있습니다.

- 정수 값
- 소프트웨어 또는 하드웨어 버전 번호

- 큰따옴표로 묶은 문자열

버전 번호

다중 플랫폼 펌웨어가 있는 Cisco IP 전화기의 소프트웨어 버전은 이 형식을 사용합니다(여기서 **BN**은 빌드 번호).

- 펌웨어 릴리스 11.3(1)SR1 및 이전 버전의 경우: sipyyyy.11-0-1MPP-376
여기서 yyyy는 전화기 모델 또는 전화기 시리즈를 나타냅니다. 11은 주 버전입니다. 0은 부 버전입니다. 1MPP는 마이크로 버전이며, 376은 빌드 번호입니다.
- 펌웨어 릴리스 11.3(2) 및 이후 버전의 경우: sipyyyy.11-3-2MPP0001-609
여기서 yyyy는 전화기 모델 또는 전화기 시리즈를 나타냅니다. 11은 주 버전입니다. 3은 부 버전입니다. 2MPP0001는 마이크로 버전이며, 609은 빌드 번호입니다.

비교하는 문자열도 같은 형식을 사용해야 합니다. 그렇지 않으면 형식 구문 분석 오류가 발생합니다. 소프트웨어 버전을 비교할 때 주 버전, 부 버전 및 마이크로 버전은 순서대로 비교되며 왼쪽 숫자가 다른 숫자보다 우선합니다. 버전 번호가 동일하면 빌드 번호가 비교됩니다.

유효한 버전 번호 예

- 펌웨어 릴리스 11.3(1) SR1 이전인 경우:
sip68xx.11-0-1MPP-312
- 펌웨어 릴리스 11.3(2) 이상인 경우:
sip68xx.11-3-2MPP0001-609

비교

- 펌웨어 릴리스 11.3(1) SR1 이전인 경우:
sipyyyy.11-3-1MPP-110 □ sipyyyy.11-2-3MPP-256
- 펌웨어 릴리스 11.3(2) 이상인 경우:
sipyyyy.11-3-2MPP0002-256 □ sipyyyy.11-3-2MPP0001-609

따옴표로 묶은 문자열은 같음 또는 같지 않음을 비교할 수 있습니다. 정수와 버전 번호는 산술적으로도 비교할 수 있습니다. 비교 연산자는 기호 또는 약어로 표기할 수 있습니다. 약어는 공개 형식 프로파일에서 조건을 표시할 때 편리합니다.

연산자	대체 구문	설명	정수 및 버전 피연산자에 적용	따옴표로 묶은 문자열 피연산자에 적용
=	eq	같음	예	예

연산자	대체 구문	설명	정수 및 버전 피연산자에 적용	따옴표로 묶은 문자열 피연산자에 적용
!=	ne	같지 않음	예	예
<	lt	보다 작음	예	아니요
<=	le	보다 작거나 같음	예	아니요
>	gt	보다 큼	예	아니요
>=	ge	보다 크거나 같음	예	아니요
AND		및	예	예

문자열 리터럴이 필요한 곳에는 매크로 변수를 큰따옴표로 묶는 것이 중요합니다. 숫자나 버전 번호가 필요한 곳에는 큰따옴표로 묶지 않습니다.

Profile_Rule* 및 Upgrade_Rule 매개 변수와 함께 사용하는 경우 이 업그레이드 규칙 예와 같이 조건식을 “(expr)?” 구문 안에 넣어야 합니다. BN을 업그레이드할 펌웨어 로드의 빌드 번호로 대체해야 합니다.

- 펌웨어 릴리스 11.3(1) SR1 이전인 경우

```
($SWVER ne sip68xx.11-0-0MPP-256)? http://ps.tell.com/sw/sip68xx.11-0-0MPP-BN.loads
```

- 펌웨어 릴리스 11.3(2) 이상인 경우

```
($SWVER ne sip68xx.11-3-2MPP0001-609)?  
http://ps.tell.com/sw/sip68xx.11-3-2MPP0001-BN.loads
```

괄호를 사용한 앞의 구문을 Resync_Trigger_* 매개 변수를 구성하는 데 사용하지 마십시오.

URL 구문

Profile_Rule* 매개 변수와 Upgrade_Rule 매개 변수에서 설정 파일과 펌웨어 로드를 검색하는 데는 표준 URL 구문을 사용합니다. 구문은 다음과 같습니다.

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

여기서 **scheme**은 다음 중 한 값입니다.

- tftp
- http
- HTTPS

scheme이 생략되면 tftp가 사용됩니다. 서버는 DNS 인식 호스트 이름이거나 숫자 IP 주소일 수 있습니다. **port**는 대상 UDP 또는 TCP 포트 번호입니다. **filepath**는 루트 디렉터리(/)로 시작해야 하며, 절대 경로여야 합니다.

server가 없으면 DHCP(옵션 66)를 통해 지정된 tftp 서버가 사용됩니다.



참고 업그레이드 규칙의 경우 `server`를 반드시 지정해야 합니다.

`port`가 없는 경우 지정된 `scheme`의 표준 포트가 사용됩니다. `Tftp`는 UDP 포트 69, `http`는 TCP 포트 80, `https`는 TCP 포트 443을 사용합니다.

`filepath`는 반드시 있어야 합니다. 이 정적 파일을 참조해야 하는 것은 아니며, CGI를 통해 얻은 동적 콘텐츠를 나타낼 수 있습니다.

매크로 확장이 URL 내에 적용됩니다. 다음은 유효한 URL의 예입니다.

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

```
/$MA.cfg
/cisco/cfg.xml
192.168.1.130/profiles/init.cfg
tftp://prov.call.com/cpe/cisco$MA.cfg
```

DHCP 옵션 66을 사용하는 경우, 빈 구문은 업그레이드 규칙에는 지원되지 않습니다. 프로파일 규칙에만 적용됩니다*.

RFC 8188 기반 HTTP 콘텐츠 암호화

전화기는 구성 파일을 위한 AES-128-GCM 암호화를 사용하여 RFC 8188 기반 HTTP 콘텐츠 암호화를 지원합니다. 이 암호화 방법을 사용하면 엔터티는 HTTP 메시지 헤더 읽을 수 있습니다. 그러나 IKM(Input Keing Material)을 알고 있는 엔터티만 페이로드를 읽을 수 있습니다. 전화기가 IKM으로 제공되면, 전화기와 프로비저닝 서버는 구성 파일을 안전하게 교환하는 동시에 제3자 네트워크 요소가 분석 및 모니터링 목적으로 메시지 헤더를 사용할 수 있습니다.

XML 구성 매개 변수 `IKM_HTTP_Encrypt_Content`는 전화기에 IKM을 유지합니다. 보안상의 이유로 이 매개 변수는 전화 관리 웹 페이지에서 액세스할 수 없습니다. 또한 전화기의 IP 주소 또는 프로비저닝 서버로 전송되는 전화기의 구성 보고서에서 액세스할 수 있는 전화기의 구성 파일에도 표시되지 않습니다.

RFC 8188 기반 암호화를 사용하려는 경우 다음 사항을 확인하십시오.

- 프로비저닝 서버에서 전화기로 전송되는 구성 파일에서 XML 매개 변수 `IKM_HTTP_Encrypt_Content`를 사용하여 IKM을 지정함으로써 IKM으로 전화기를 프로비저닝합니다.
- 이 암호화가 프로비저닝 서버에서 전화기로 전송된 구성 파일에 적용되는 경우 구성 파일의 `Content-Encoding` 헤더에 “aes128gcm”이 있는지 확인하십시오

이 헤더가 없는 경우, AES-256-CBC 방법이 우선합니다. 전화기는 IKM에 상관없이 AES-256-CBC 키가 프로파일 규칙에 있는 경우 AES-256-CBC 암호 해독을 적용합니다.

- 전화기가 프로비저닝 서버로 전송하는 구성 보고서에 이 암호화를 적용하려는 경우 보고서 규칙에 지정된 AES-256-CBC 키가 없는지 확인하십시오.

선택적 재동기화 인수

선택적 인수 **key**, **uid**, **pwd**는 Profile_Rule * 파라미터에 입력된 URL 앞에 전체를 대괄호로 묶고 지정할 수 있습니다.

key

--key 옵션은 파일의 *Content-Encoding* 헤더가 “aes128gcm” 암호화를 나타내지 않는 한, 프로비저닝 서버에서 수신하는 설정 파일이 AES-256-CBC 암호화로 암호화됨을 전화기에 알려줍니다. 키 자체는 조건 **--key** 이후의 문자열로 지정됩니다. 선택적으로 큰따옴표(")로 키를 묶을 수 있습니다. 전화기는 키를 사용하여 구성 파일의 암호를 해독합니다.

사용 예

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

대괄호로 묶은 선택적 인수에는 매크로 확장이 적용됩니다. 특수 목적 매개 변수 GPP_SA ~ GPP_SD는 key 옵션 인수로 사용된 경우에만 매크로 변수 SSA ~ SSD로 매크로 확장됩니다. 다음 예를 참조하십시오.

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

공개 형식 프로파일에서 **--key**에 대한 인수는 **openssl**에 지정하는 **-k** 옵션에 대한 인수와 동일해야 합니다.

uid 및 pwd

uid 및 **pwd** 옵션을 사용하여 지정된 URL이 요청될 때 HTTP 기본 및 다이제스트 인증 문제에 대한 응답으로 전송되는 사용자 ID 및 암호를 지정할 수 있습니다. 대괄호로 묶은 선택적 인수에는 매크로 확장이 적용됩니다. 특수 목적 매개 변수 GPP_SA ~ GPP_SD는 key 옵션 인수로 사용된 경우에만 매크로 변수 SSA ~ SSD로 매크로 확장됩니다. 다음 예를 참조하십시오.

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

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

다음과 같이 확장됩니다.

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

프로파일을 전화기에 적용

XML 구성을 만든 다음에는 이를 전화 애플리케이션으로 전달해야 합니다. 구성을 적용하려면 웹 브라우저를 사용하여 구성 파일을 TFTP, HTTP 또는 HTTPS 서버에서 전화기로 다운로드하거나 cURL 명령줄 유틸리티를 사용하여 다운로드할 수 있습니다.

TFTP 서버에서 전화기로 구성 파일 다운로드

구성 파일을 PC의 TFTP 서버 애플리케이션으로 다운로드하려면 다음 단계를 완료합니다.

프로시저

단계 1 PC를 전화기 LAN에 연결합니다.

단계 2 PC에서 TFTP 서버 애플리케이션을 실행하고 TFTP 루트 디렉터리에 구성 파일이 있는지 확인합니다.

단계 3 웹 브라우저에서 전화기 LAN IP 주소, 컴퓨터의 IP 주소, 파일 이름, 로그인 자격 증명을 입력합니다. 다음 형식을 사용합니다.

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

예:

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

cURL을 사용하여 전화기로 구성 파일 다운로드

cURL을 사용하여 구성을 전화기로 다운로드하려면 다음 과정을 완료합니다. 이 명령줄 도구는 URL 구문을 사용하여 데이터를 전송하는 데 사용됩니다. cURL을 다운로드하려면 다음 주소를 방문합니다.

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



참고 cURL을 사용할 때 사용자 이름과 암호가 캡처될 수 있으므로, 구성을 전화기로 보내는 때는 cURL을 사용하지 않는 것이 좋습니다.

프로시저

단계 1 PC를 전화기의 LAN 포트에 연결합니다.

단계 2 다음 cURL 명령을 입력해 구성 파일을 전화기로 다운로드합니다.

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

프로비저닝 매개 변수 유형

이 섹션에서는 기능에 따라 광범위하게 프로비저닝 매개 변수를 설명합니다.

다음과 같은 프로비저닝 매개 변수 유형이 있습니다.

- 일반 목적
- 활성화
- 트리거
- 구성 가능한 일정
- 프로파일 규칙
- 업그레이드 규칙

일반 목적 매개 변수

일반 목적 매개 변수 **GPP_***(관리자 로그인 > 고급 > 음성 > 프로비저닝)는 자유 문자열로 사용되고 특정 프로비저닝 서버 솔루션과 상호 작용하기 위해 Cisco IP 전화기를 구성하는 경우에 등록합니다. **GPP_*** 매개 변수는 기본적으로 비어 있습니다. 다음을 포함한 다양한 값을 포함하도록 구성할 수 있습니다.

- 암호화 키
- URL
- 다중 단계 프로비저닝 상태 정보.
- 게시 요청 템플릿
- 매개 변수명 별칭 매핑
- 최종적으로 부분 문자열 값은 완전한 매개 변수 값으로 결합됩니다.

GPP_* 매개 변수는 다른 프로비저닝 매개 변수 내에서 매크로 확장으로 이용 가능합니다. 이를 위해서는 **GPP_A**에서 **GPP_P**까지의 내용을 식별하는 데 한 자의 대문자 매크로 이름이면 충분합니다. 또한 다음 URL 옵션의 인수로 사용되는 특별한 경우 **SA**에서 **SD**까지의 두 대문자 매크로 이름으로 **GPP_SA**에서 **GPP_SD**까지 식별합니다.

key, uid, pwd

이러한 매개 변수는 프로비저닝과 업그레이드 규칙에 변수로 사용할 수 있습니다. 이는 **\$GPP_A**와 같이 변수 이름 앞에 '\$' 문자를 붙여 참조됩니다.

일반 목적 매개 변수 사용

예를 들어 GPP_A에 문자열 ABC가 포함되고 GPP_B에 123이 포함된 경우 식 \$A\$B 매크로는 ABC123으로 확장됩니다.

시작하기 전에

전화기 관리 웹페이지 액세스. [전화기 웹 인터페이스 액세스](#) 참조

프로시저

-
- 단계 1 음성 > 프로비저닝을 선택합니다.
 - 단계 2 일반 목적 매개 변수 섹션으로 스크롤합니다.
 - 단계 3 GPP A 필드부터 GPP P 필드까지 유효한 값을 입력합니다.
 - 단계 4 모든 변경 사항 제출을 클릭합니다.
-

매개 변수 활성화

Provision_Enable 및 Upgrade_Enable 매개 변수는 모든 프로파일 재동기화 및 펌웨어 업그레이드 작업을 제어합니다. 이러한 매개 변수는 재동기화 및 업그레이드를 서로 독립적으로 제어합니다. 또한 이러한 매개 변수는 관리 웹 서버를 통해 전달된 재동기화 및 업그레이드 URL 명령을 제어합니다. 이러한 매개 변수는 기본적으로 **Yes**로 설정됩니다.

Resync_From_SIP 매개 변수는 재동기화 작업을 위한 요청을 제어합니다. SIP NOTIFY 이벤트가 서비스 제공자 프록시 서버에서 전화기로 전송됩니다. 활성화된 경우 프록시가 재동기화를 요청할 수 있습니다. 이렇게 하기 위해 프록시는 Event: resyn 헤더가 포함된 SIP NOTIFY 메시지를 장치로 전송합니다.

장치는 401 응답(사용된 자격 증명에 대해 권한 부여 거절)으로 해당 요청에 응답합니다. 장치는 인증됨 후속 요청을 기다린 후 프록시의 재동기화 요청을 이행합니다. Event: reboot_now 및 Event: restart_now 헤더는 각각 콜드 재시작 및 워م 재시작을 수행하며, 모두 응답을 받습니다.

나머지 두 개 항목은 Resync_On_Reset 및 Resync_After_Upgrade_Attempt입니다. 이러한 매개 변수는 장치가 전원 켜기 소프트웨어 재부팅 후 및 각 업그레이드 시도 후 재동기화 작업을 수행할지 여부를 결정합니다.

Resync_On_Reset가 활성화되면 부팅 시퀀스 후 재설정이 수행되기 전에 임의의 지연을 추가합니다. 최대 지연 시간은 Resync_Random_Delay에 지정된 값(초)입니다. 전화기의 풀에서 동시에 전화기를 켜는 경우 각 장치의 재동기화 요청 시작 시간을 분산시킵니다. 이 기능은 대규모 거주 지역 구축에서 지역 정전 발생 시 유용합니다.

트리거

전화기는 특정한 시간 또는 특정한 간격으로 재동기화할 수 있습니다.

특정 간격으로 재동기화

전화기는 주기적으로 프로비저닝 서버로 재동기화하도록 설계되었습니다. 재동기화 간격은 `Resync_Periodic(초)`에서 구성합니다. 이 값이 비어 있으면 장치는 주기적으로 재동기화하지 않습니다.

일반적으로 재동기화는 음성 회선이 유희 상태인 경우에 수행됩니다. 재동기화 시간이 되었을 때 음성 회선이 활성 상태이면, 전화기는 회선이 유희 상태가 될 때까지 재동기화 절차를 지연시킵니다. 재동기화로 인해 구성 매개 변수 값이 변경될 수 있습니다.

전화기가 서버에서 프로파일을 검색할 수 없거나, 다운로드한 파일이 손상되었거나, 내부 오류가 발생하면 재동기화 작업이 실패할 수 있습니다. 장치는 `Resync_Error_Retry_Delay`에 지정된 시간(초)이 지난 후 재동기화를 다시 시도합니다. `Resync_Error_Retry_Delay`가 0으로 설정된 경우 장치는 재동기화 시도가 실패한 후 재동기화를 다시 시도하지 않습니다.

업그레이드가 실패한 경우 `Upgrade_Error_Retry_Delay`에 지정된 시간(초)이 지난 후 다시 시도합니다.

재동기화를 조건부로 트리거하는 데 사용할 수 있는 두 가지 구성 가능한 매개 변수 `Resync_Trigger_1` 및 `Resync_Trigger_2`가 있습니다. 각 매개 변수는 매크로 확장을 거치는 조건식으로 프로그래밍할 수 있습니다. 재동기화 간격이 만료되었을 때(다음 재동기화 시간이 되었을 때) 설정된 트리거가 있는 경우, 하나 이상의 트리거가 참으로 평가되지 않으면 재동기화를 차단합니다.

다음 예에서 조건은 재동기화를 트리거합니다. 이 예에서 마지막 전화기 업그레이드 시도는 5분(300 초) 이상 경과되었고 마지막 재동기화 시도는 10분(600) 이상 경과되었습니다.

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

특정 시간에 재동기화

`Resync_At` 매개 변수를 사용하면 특정 시간에 전화기를 재동기화할 수 있습니다. 이 매개 변수는 24 시간 형식(hhmm)으로 시간을 지정합니다.

`Resync_At_Random_Delay` 매개 변수는 전화기가 지정되지 않은 지연 시간에 재동기화하도록 허용하지 않습니다. 이 매개 변수는 양의 정수 형식으로 시간을 지정합니다.

다수의 전화기를 동시에 재동기화하도록 설정하여 재동기화 요청이 서버로 쇄도하지 않도록 해야 합니다. 이를 위해 전화기는 지정된 시간이 지난 후 최대 10분 후 재동기화를 트리거합니다.

예를 들어 재동기화 시간을 1000(오전 10시)으로 설정한 경우, 전화기는 오전 10:00 ~ 10:10 사이에 재동기화를 트리거합니다.

이 기능은 기본적으로 비활성화됩니다. `Resync_At` 매개 변수가 프로비저닝된 경우 `Resync_Periodic` 매개 변수는 무시됩니다.

구성 가능한 일정

다음과 같은 프로비저닝 매개 변수를 사용하여 주기적 재동기화를 예약할 수 있으며, 재동기화 및 업그레이드 실패에 대한 재시도 간격을 지정할 수 있습니다.

- `Resync_Periodic`
- `Resync_Error_Retry_Delay`

• Upgrade_Error_Retry_Delay

각 매개 변수는 단일 지연 값(초)을 받습니다. 새 확장 구문에서는 연속된 지연 요소를 쉼표로 구분된 목록으로 사용할 수 있습니다. 시퀀스의 마지막 요소는 암시적으로 무한 반복됩니다.

선택적으로 더하기 기호를 사용하여 임의의 추가 지연을 적용하는 숫자 값을 지정할 수 있습니다.

예 1

이 예에서는 전화기는 2시간마다 주기적으로 재동기화합니다. 재동기화가 실패한 경우, 장치는 30분, 1시간, 2시간, 4시간 간격으로 재시도합니다. 장치는 재동기화가 성공할 때까지 4시간 간격으로 계속 시도합니다.

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

예 2

이 예에서는 장치는 매시간(더하기 임의 지연 시간 최대 10분) 주기적으로 재동기화합니다. 재동기화가 실패한 경우, 장치는 30분(더하기 최대 5분), 1시간(더하기 최대 10분), 2시간(더하기 최대 15분) 간격으로 재시도합니다. 장치는 재동기화가 성공할 때까지 2시간(더하기 최대 15분) 간격으로 계속 시도합니다.

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

예 3

이 예에서 원격 업데이트 시도가 실패한 경우, 장치는 30분 내에 업데이트를 재시도하고, 1시간 후, 2시간 후 재시도합니다. 여전히 업데이트가 실패한 경우, 장치는 업데이트가 성공할 때까지 4-5시간마다 재시도합니다.

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

프로파일 규칙

전화기기는 여러 원격 구성 프로파일 매개 변수(Profile_Rule*)를 제공합니다. 각 재동기화 작업은 여러 다른 서버에서 관리하는 여러 파일을 검색할 수 있습니다.

가장 간단한 시나리오에서는 장치가 중앙 서버의 단일 프로파일로 주기적으로 재동기화하고 모든 관련된 내부 매개 변수를 업데이트합니다. 프로파일이 여러 다른 파일로 분할되는 경우도 있습니다. 한 파일이 모든 한 구축 내의 모든 전화기에 공통입니다. 각 계정에 대해 별도의 고유한 파일이 제공됩니다. 암호화 키 및 인증서 정보는 별도의 서버에 저장된 다른 프로파일을 통해 제공할 수 있습니다.

재동기화 작업의 일정이 되면 전화기가 네 개의 Profile_Rule* 매개 변수를 순서대로 평가합니다.

1. Profile_Rule

- 2. Profile_Rule_B
- 3. Profile_Rule_C
- 4. Profile_Rule_D

각 평가의 결과로 원격 프로비저닝 서버에서 프로파일이 검색될 수 있으며 여러 내부 매개 변수 일부가 업데이트될 수 있습니다. 평가가 실패하면 재동기화 시퀀스가 중단되며 Resync_Error_Retry_Delay 매개 변수에 지정된 시간(초)이 지난 후 재시도합니다. 모든 평가가 성공한 경우 장치는 Resync_Periodic 매개 변수에 지정된 시간(초) 동안 대기한 후 다른 재동기화를 수행합니다.

each Profile_Rule* 매개 변수의 내용은 대체 항목의 집합으로 구성될 수 있습니다. 대체 항목은 | (파이프) 문자로 구분합니다. 각 대체 항목은 조건식, 할당식, 프로파일 URL, 연관된 URL 옵션으로 구성됩니다. 이러한 모든 구성 요소는 각 대체 항목 내에서 선택 사항입니다. 다음은 올바른 조합이며, 지정해야 하는 순서입니다(있는 경우).

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

각 Profile_Rule* 매개 변수 내에서 마지막 항목을 제외한 모든 대체 항목은 조건식을 제공해야 합니다. 이 식은 다음과 같은 평가 및 처리됩니다.

1. 조건은 왼쪽에서 오른쪽으로 참으로 평가되는 항목이 나올 때까지 평가됩니다(또는 조건식이 없는 대체 항목이 나올 때까지).
2. 함께 지정된 할당식이 있으면 평가됩니다.
3. 대체 항목의 일부로 지정된 URL이 있는 경우 지정된 URL에 있는 프로파일을 다운로드하려고 시도합니다. 시스템은 내부 매개 변수를 적절하게 업데이트하려고 시도합니다.

모든 대체 항목에 조건식이 있고 참으로 평가되는 식이 없는 경우(또는 전체 프로파일 규칙이 비어 있는 경우) 전체 Profile_Rule* 매개 변수를 건너뛵니다. 순서의 다음 프로파일 규칙 매개 변수가 평가됩니다.

예 1

이 예에서는 지정된 URL에 있는 프로파일로 무조건 재동기화하며 원격 프로비저닝 서버로 HTTP GET 요청을 수행합니다.

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

예 2

이 예에서 장치는 회선 1의 등록 상태에 따라 서로 다른 두 URL에 재동기화됩니다. 등록이 해제된 경우, 장치는 CGI 스크립트에 대해 HTTP POST를 수행합니다. 장치는 장치 상태에 대한 추가 정보를 제공할 수 있는 매크로 확장 GPP_A의 내용을 전송합니다.

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

예 3

이 예에서 장치는 동일한 서버로 재동기화합니다. 장치에 인증서가 설치되지 않은 경우(2.0 이전 레저시 장치) 장치가 추가 정보를 제공합니다.

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

예 4

이 예에서 회선 1은 GPP_A가 첫 번째 URL을 통해 프로비저닝됨으로 설정될 때까지 비활성화됩니다. 그 후에는 두 번째 URL로 재동기화됩니다.

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

예 5

이 예에서 서버가 반환하는 프로파일은 XML 요소 태그를 포함하는 것으로 간주됩니다. 이러한 태그는 GPP_B에 저장된 별칭 맵에 따라 적절한 매개 변수명으로 재매핑해야 합니다.

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

일반적으로 요청된 프로파일을 서버에서 가져올 수 없으면 재동기화가 실패한 것으로 간주합니다. Resync_Fails_On_FNF 매개 변수로 이 기본 동작을 무시할 수 있습니다. Resync_Fails_On_FNF가 No로 설정되면 장치가 서버에서 파일을 찾을 수 없음 응답을 받아도 재동기화가 성공한 것으로 간주합니다. Resync_Fails_On_FNF의 기본값은 Yes입니다.

업그레이드 규칙

업그레이드 규칙은 장치에 새로운 로드를 활성화하도록 지시하며 필요한 경우 해당 로드를 얻는 위치를 알려줍니다. 장치에 로드가 이미 있는 경우 로드를 얻으려고 하지 않습니다. 따라서 원하는 로드가 비활성 파티션에 있는 경우 로드 위치의 유효성은 상관이 없습니다.

Upgrade_Rule은 펌웨어 로드를 지정하며, 현재 로드와 다른 경우 지정한 로드를 다운로드하고 조건식에 의해 제한되거나 Upgrade_Enable가 아니므로 설정되지 않은 경우 해당 로드를 적용합니다.

전화기는 하나의 구성 가능한 원격 업그레이드 매개 변수 Upgrade_Rule을 제공합니다. 이 매개 변수는 프로파일 규칙 매개 변수와 비슷한 구문을 사용합니다. 업그레이드에는 URL 옵션이 지원되지 않지만, 조건식과 할당식은 사용할 수 있습니다. 조건식이 사용된 경우 | 문자로 구분한 여러 대체 항목으로 매개 변수를 채울 수 있습니다. 각 대체 항목의 구문은 다음과 같습니다.

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

Profile_Rule* 매개 변수와 마찬가지로 Upgrade_Rule 매개 변수는 조건식이 충족되거나 대체 항목에 조건식이 없을 때까지 각 대체 항목을 평가합니다. 함께 지정된 할당식이 있는 경우 평가됩니다. 그런 다음 지정된 URL로 업그레이드를 시도합니다.

Upgrade_Rule 규칙에 조건식이 없는 URL이 포함된 경우 장치는 URL이 지정하는 펌웨어 이미지로 업그레이드합니다. 매크로를 확장하고 규칙을 평가한 후, 장치는 규칙이 수정되거나 체계 + 서버 + 포트 + 파일 경로의 실제 조합이 수정되기 전에는 업그레이드를 다시 시도하지 않습니다.

장치는 펌웨어 업그레이드를 시도하기 위해 절차를 시작할 때 오디오를 비활성화하며 절차가 끝날 때 재부팅합니다. 장치는 모든 음성 회선이 현재 비활성인 경우에만 Upgrade_Rule의 내용을 기준으로 자동으로 업그레이드를 시작합니다.

예를 들어,

- Cisco IP 전화기 6821의 경우:

```
http://p.tel.com/firmware/sip6821.11-3-1MPP-678.loads
```

- Cisco IP 6800 시리즈의 다른 전화기의 경우:

```
http://p.tel.com/firmware/sip68xx.11-3-1MPP-678.loads
```

이 예에서 Upgrade_Rule은 펌웨어를 지정된 URL에 저장된 이미지로 업그레이드합니다.

또 다른 예는 다음과 같습니다.

- Cisco IP 전화기 6821의 경우:

```
("F" ne "beta-customer")? http://p.tel.com/firmware/sip6821.11-3-1MPP-678.loads  
| http://p.tel.com/firmware/sip6821.11-3-1MPP-678.loads
```

- Cisco IP 전화기 6800 시리즈의 다른 모델의 경우:

```
("F" ne "beta-customer")? http://p.tel.com/firmware/sip68xx.11-3-1MPP-678.loads  
| http://p.tel.com/firmware/sip68xx.11-3-1MPP-678.loads
```

이 예에서는 일반 목적 매개 변수 GPP_F의 내용에 따라 두 이미지 중 하나를 장치에 로드합니다.

장치에서 펌웨어 개정 번호를 지정해 다운그레이드 제한을 적용할 수 있으며, 사용자 지정 옵션에 유용할 수 있습니다. Downgrade_Rev_Limit 매개 변수에 유효한 펌웨어 개정을 구성하면, 장치는 지정 한 한계보다 이전의 펌웨어 버전에 대한 업그레이드 시도를 거부합니다.

데이터 유형

다음과 같은 데이터 유형이 구성 프로파일 매개 변수에 사용됩니다.

- {a,b,c,...} - a, b, c, ... 중에서 선택
- Bool - 부울 값 "yes" 또는 "no".
- CadScript - 신호의 리듬 매개 변수를 지정하는 미니스크립트. 최대 127자.
구문: S₁[:S₂]

- $S_i = D_i(\text{on}_{i,1}/\text{off}_{i,1}[\text{on}_{i,2}/\text{off}_{i,2}[\text{on}_{i,3}/\text{off}_{i,3}[\text{on}_{i,4}/\text{off}_{i,4}[\text{on}_{i,5}/\text{off}_{i,5}[\text{on}_{i,6}/\text{off}_{i,6}]]]])$ 이를 섹션이라고 합니다.
- $\text{on}_{i,j}$ 및 $\text{off}_{i,j}$ 는 *segment*의 켜기/끄기 시간(초)입니다. $i = 1$ 또는 2 , 그리고 $j = 1 \sim 6$.
- D_i 는 섹션의 총 시간(초)입니다.

총 시간은 1ms 해상도로 소수점 세 자리까지 적을 수 있습니다. 와일드 카드 문자 "*"는 무한 시간을 의미합니다. 섹션 내 세그먼트는 순서대로 재생되며 전체 시간을 재생할 때까지 반복됩니다.

예 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
```

예 2 - 벨 소리 구분(짧음, 짧음, 짧음, 김):

```
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 - 회선 1 및 회선 2 다이얼 플랜을 지정하는 데 사용되는 스크립트 구문.
- Float<n> - 소수점 n 자리를 포함하는 부동소수점 값.
- FQDN - 정규화된 도메인 이름. 최대 63자를 포함할 수 있습니다. 예는 다음과 같습니다.
 - sip.Cisco.com:5060 또는 109.12.14.12:12345
 - sip.Cisco.com 또는 109.12.14.12
- FreqScript - 신호음의 주파수 및 레벨 매개 변수를 지정하는 미니스크립트. 최대 127자를 포함할 수 있습니다.

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

 - F_1 - F_6 주파수(Hz)(부호 없는 정수만).
 - L_1 - L_6 레벨(dBm)(최대 소수점 한 자리).

쉼표 앞뒤로 공백을 넣을 수 있지만 권장되지 않습니다.

예 1 - 통화 대기 신호음:

```
440@-10

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

예 2 - 다이얼톤:

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

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

- IP - x.x.x.x 형식의 유효한 IPv4 주소, 여기서 x 값의 범위는 0~255입니다. 예: 10.1.2.100.
- UserID - URL에 표시되는 사용자 ID. 최대 63자입니다.
- Phone - 전화 번호 문자열(예: 14081234567, *69, *72, 345678) 또는 일반 URL(예: 1234@10.10.10.100:5068 또는 jsmith@Cisco.com). 문자열은 최대 39자를 포함할 수 있습니다.
- PhTmpl - 전화 번호 템플릿. 각 템플릿은 쉼표(,)로 구분된 패턴을 하나 이상 포함할 수 있습니다. 각 패턴 시작 부분의 공백은 무시됩니다. "?" 및 "*" 문자는 와일드 카드 문자입니다. 해당 문자를 그대로 표시하려면 %xx 형식을 사용합니다. 예를 들어 %2a는 *입니다. 템플릿은 최대 39자를 포함할 수 있습니다. 예: "1408*, 1510*", "1408123????, 555?1".
- Port - TCP/UDP 포트 번호(0 ~ 65535). 십진수 또는 16진수 형식으로 지정할 수 있습니다.
- ProvisioningRuleSyntax - 구성 재동기화 및 펌웨어 업그레이드 규칙을 정의하는 데 사용되는 스크립트 구문.
- PwrLevel - 소수점 한 자리로 표시하는 파워 레벨(dBm)(예: 13.5 또는 1.5dBm).
- RscTmpl - SIP 응답 상태 코드의 템플릿(예: "404, 5*", "61?", "407, 408, 487, 481"). 최대 39자를 포함할 수 있습니다.
- Sig<n> - 부호 있는 n비트 값. 십진수 또는 16진수 형식으로 지정할 수 있습니다. 음수 값 앞에는 "-" 기호를 붙여야 합니다. 양수 값의 + 기호는 선택 사항입니다.
- Star Codes - 보조 서비스에 대한 활성화 코드입니다(예: *69). 코드는 최대 7자를 포함할 수 있습니다.
- Str<n> - 예약되지 않은 문자를 최대 n자 포함하는 일반 문자열.
- Time<n> - 소수점 n자리의 시간(초). 추가 지정된 소수 자릿수는 무시됩니다.
- ToneScript - 통화 연결음의 주파수, 레벨, 리듬 매개 변수를 지정하는 미니스크립트. 스크립트는 최대 127자를 포함할 수 있습니다.

구문: FreqScript;Z₁[:Z₂].

섹션 Z_1 은 CadScript의 S_1 섹션과 비슷하지만 각 켜기/끄기 세그먼트 뒤에 주파수 구성 요소 매개 변수가 나오는 점이 다릅니다. $Z_1 = D_1(\text{on}_{i,1}/\text{off}_{i,1}/f_{i,1}[\text{on}_{i,2}/\text{off}_{i,2}/f_{i,2} [\text{on}_{i,3}/\text{off}_{i,3}/f_{i,3} [\text{on}_{i,4}/\text{off}_{i,4}/f_{i,4} [\text{on}_{i,5}/\text{off}_{i,5}/f_{i,5} [\text{on}_{i,6}/\text{off}_{i,6}/f_{i,6}]]]]])$

- $f_{i,j} = n_1[+n_2]+n_3[+n_4[+n_5[+n_6]]]]]$.
- $1 < n_k < 6$ 부분은 FreqScript의 해당 세그먼트에 사용되는 주파수 구성 요소를 지정합니다.

세그먼트에 주파수 구성 요소가 둘 이상 사용되면 구성 요소가 합산됩니다.

예 1 - 다이얼톤:

```
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
```

예 2 - 서터음:

```
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
```

- $\text{Uns}\langle n \rangle$ - 부호 없는 n 비트 값(여기서 $n = 8, 16$ 또는 32). 값이 n 비트에 적합한 경우에는 10진수 또는 16진수 형식(예: 12 또는 0x18)으로 지정할 수 있습니다.



참고 다음 사항에 유의하십시오.

- <Par Name>은 설정 파라미터명을 나타냅니다. 프로파일에 해당되는 태그는 공백을 밑줄("_")로 대체하여 작성됩니다(예: **Par_Name**).
- 빈 기본값 필드는 빈 문자열 <" ">을 의미합니다.
- 전화기는 지정된 프로파일에 없는 태그에 대해서는 마지막으로 구성된 값을 계속 사용합니다.
- 템플릿은 지정된 순서로 비교됩니다. 첫 번째 가장 가깝지 않은 일치 항목이 선택됩니다. 매개 변수 이름은 정확하게 일치해야 합니다.
- 한 프로파일에 한 매개 변수에 대한 정의가 두 개 이상 있는 경우, 파일의 마지막 항목이 전화기에 적용됩니다.
- 빈 매개 변수 값으로 매개 변수를 지정하면 해당 매개 변수가 기본값으로 돌아갑니다. 빈 문자열을 지정하려면 매개 변수 값으로 빈 문자열 ""을 사용합니다.

프로파일 업데이트 및 펌웨어 업그레이드

전화기는 보안 원격 프로비저닝(구성) 및 펌웨어 업그레이드를 지원합니다. 프로비저닝되지 않은 전화기는 해당 장치를 대상으로 하는 암호화되지 않은 프로파일을 수신할 수 있습니다. SSL 기능을 사용하는 안전한 초기 프로비저닝 메커니즘이 있으므로, 전화기에는 명시적 키가 필요 없습니다.

프로파일 업데이트를 시작 또는 완료하거나 펌웨어 업그레이드, 또는 이전 릴리스에서 향후 업그레이드 상태로 도달하기 위해 중간 업그레이드가 필요한 경우 사용자의 조정은 필요 없습니다. 재동기화 때문에 소프트웨어 재부팅과 통화 중단이 발생할 수 있으므로, 프로파일 재동기화는 전화기가 유휴 상태일 때만 시도됩니다.

일반 목적 매개 변수는 프로비저닝 프로세스를 관리합니다. NPS(Normal Provisioning Server)를 주기적으로 연결하도록 전화기를 구성할 수 있습니다. 업데이트된 프로파일은 공유된 비밀 키를 사용하여 암호화되므로, NPS와의 통신에 보안 프로토콜을 사용할 필요는 없습니다. NPS는 클라이언트 인증서를 가진 표준 TFTP, HTTP 또는 HTTPS 서버일 수 있습니다.

관리자는 전화기 웹 사용자 인터페이스를 사용하여 전화기를 업그레이드, 재부팅, 재시작 또는 재동기화할 수 있습니다. 관리자는 SIP 알림 메시지를 사용하여 이러한 작업을 할 수도 있습니다.

구성 프로파일은 서비스 제공자 프로비저닝 시스템과 통합되는 일반적인 오픈소스 도구를 사용하여 생성합니다.

프로파일 업데이트 허용

프로파일 업데이트를 지정된 간격으로 수행하도록 허용할 수 있습니다. 업데이트된 프로파일은 TFTP, HTTP 또는 HTTPS를 사용하여 서버에서 전화기로 전송됩니다.

XML(cfg.xml) 코드를 사용하여 전화기 설정 파일에서 매개 변수를 설정할 수도 있습니다.

시작하기 전에

전화기 관리 웹페이지 액세스. [전화기 웹 인터페이스 액세스](#) 참조

프로시저

단계 1 음성 > 프로비저닝을 선택합니다.

단계 2 설정 프로파일 섹션의 프로비전 활성화 파라미터에서 예를 선택합니다.

다음 형식으로 문자열을 입력하여 구성 XML 파일(cfg.xml)에서 이 매개 변수를 구성할 수 있습니다.

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

기본값: 예

단계 3 [프로파일 재동기화 매개 변수](#) 표에 설명된 대로 매개 변수를 설정합니다.

단계 4 모든 변경 사항 제출을 클릭합니다.

펌웨어 업그레이드 허용 및 구성

펌웨어 업데이트를 지정된 간격으로 수행하도록 허용할 수 있습니다. 업데이트된 펌웨어는 TFTP나 HTTP를 사용하여 서버에서 전화기로 전송됩니다. 펌웨어에는 개인 정보가 포함되어 있지 않으므로 펌웨어 업그레이드에서 보안은 덜 중요한 문제입니다.

XML(cfg.xml) 코드를 사용하여 전화기 설정 파일에서 매개 변수를 설정할 수도 있습니다.

시작하기 전에

전화기 관리 웹페이지 액세스. [전화기 웹 인터페이스 액세스](#) 참조

프로시저

단계 1 음성 > 프로비저닝을 선택합니다.

단계 2 펌웨어 업그레이드 섹션의 업그레이드 활성화 파라미터에서 예를 선택합니다.

다음 형식으로 문자열을 입력하여 구성 XML 파일(cfg.xml)에서 이 매개 변수를 구성할 수 있습니다.

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

옵션: 예 및 아니요

기본값: 예

단계 3 업그레이드 오류 재시도 지연 매개 변수를 초 단위로 설정합니다.

업그레이드 실패 시 적용되는 업그레이드 재시도 간격(초)입니다. 장치에는 펌웨어 업그레이드 시도가 실패하면 활성화되는 펌웨어 업그레이드 오류 타이머가 있습니다. 해당 타이머는 이 매개 변수의

값으로 초기화됩니다. 이 타이머 카운트가 0 미만이 되는 경우 다음 펌웨어 업그레이드가 시도됩니다.

다음 형식으로 문자열을 입력하여 구성 XML 파일(cfg.xml)에서 이 매개 변수를 구성할 수 있습니다.

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

기본값: 3600

:

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

단계 4 업그레이드 조건 및 관련 펌웨어 URL을 정의하는 펌웨어 업그레이드 스크립트를 입력하여 업그레이드 규칙 매개 변수를 설정합니다. 프로파일 규칙과 동일한 구문을 사용합니다. 스크립트를 입력하고 다음 형식을 사용하여 업그레이드 규칙 입력:

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

예:

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

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

다음 형식으로 문자열을 입력하여 구성 XML 파일(cfg.xml)에서 이 매개 변수를 구성할 수 있습니다.

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

단계 5 모든 변경 사항 제출을 클릭합니다.

TFTP, HTTP 또는 HTTPS를 통해 펌웨어 업그레이드

전화기는 TFTP, HTTP 또는 HTTPS를 사용하여 펌웨어 업그레이드를 지원합니다.



참고 이전 릴리스로 다운그레이드하는 기능은 일부 장치에서 지원되지 않을 수 있습니다. 자세한 내용은 전화기 및 펌웨어 버전에 대한 릴리스 노트를 참조합니다.

시작하기 전에

액세스할 수 있는 서버로 펌웨어 로드 파일을 다운로드해야 합니다.

프로시저

단계 1 다음과 같이 이미지의 이름을 바꿉니다.

cp-x8xx-sip.aa-b-cMPP.cop를 **cp-x8xx-sip.aa-b-cMPP.tar.gz**로 바꿉니다.

여기서:

x8xx는 전화기 시리즈입니다(예: 6841).

aa-b-c는 릴리스 번호입니다(예: 10-4-1)

- 단계 **2** 타르 볼을 풀어내려면 **tar -xvzf** 명령을 사용합니다.
- 단계 **3** 폴더를 TFTP, HTTP 또는 HTTPS 다운로드 디렉터리로 복사합니다.
- 단계 **4** 전화기 관리 웹페이지 액세스. [전화기 웹 인터페이스 액세스](#) 참조
- 단계 **5** 음성 > 프로비저닝을 선택합니다.
- 단계 **6** **.loads**로 끝나는 로드 파일 이름을 찾고 이를 유효한 URL에 추가합니다.
- 단계 **7** 모든 변경 사항 제출을 클릭합니다.

브라우저 명령으로 펌웨어 업그레이드

브라우저 주소 표시줄에 업그레이드 명령을 입력하여 전화기에서 펌웨어를 업그레이드할 수 있습니다. 전화기는 유틸 상태일 때만 업데이트됩니다. 통화가 끝나면 자동으로 업데이트를 시도합니다.

프로시저

웹 브라우저에서 URL을 사용하여 전화기를 업그레이드하려면 다음이 명령을 입력합니다.

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