

# CMS/Acano Call Bridge でのレコーダーの設定

## 目次

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[配備](#)

[サポートされた配備](#)

[他のセットアップ](#)

[設定](#)

[確認とトラブルシューティング](#)

## 概要

このドキュメントでは、Cisco Meeting Server ( CMS ) Call Bridge ( CB ) でレコーダーをセットアップするために必要な設定手順について説明します。

レコーダーは Acano サーバのリリース 1.9 から利用できます。レコーダーはネットワーク ファイル システム ( NFS ) 資料 ストレージで記録会議および保存の capability を記録物提供します。

## 前提条件

### 要件

このドキュメントに関しては個別の要件はありません。

### 使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

1. CMS 1.9 またはそれ以上に
2. Google Chrome からの Postman
3. CMS Application Program Interface ( API )

本書の情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。使用された削除された ( デフォルト ) コンフィギュレーションでデバイスすべてはここに起動しています。稼働中のネットワークで作業を行う場合、コマンドの影響について十分に理解したうえで作業してください。

## 背景説明

1. レコーダーはコールブリッジをホストする拡張可能なメッセージングおよび存在プロトコル (XMPP) クライアントのように、そうサーバで有効になる XMPP サーバ必要動作します。
2. レコーダー ライセンス、CB およびないレコーダー サーバで必要がある、場合バージョンをある、詳細 CMS 2.X 前にここに実行している <https://kb.acano.com/content/23/280/en/how-does-licensing-work-on-the-acano-solution.html>。
3. Windows サーバか Linux で設定することができるネットワーク ファイル システム (NFS) ディレクトリ。

Windows サーバに関しては、このリンクのステップに従って下さい:  
[https://technet.microsoft.com/en-us/library/jj574143\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/jj574143(v=ws.11).aspx)。

Linux に関しては、このリンクのステップに従って下さい:  
<https://help.ubuntu.com/lts/serverguide/network-file-system.html>。

---

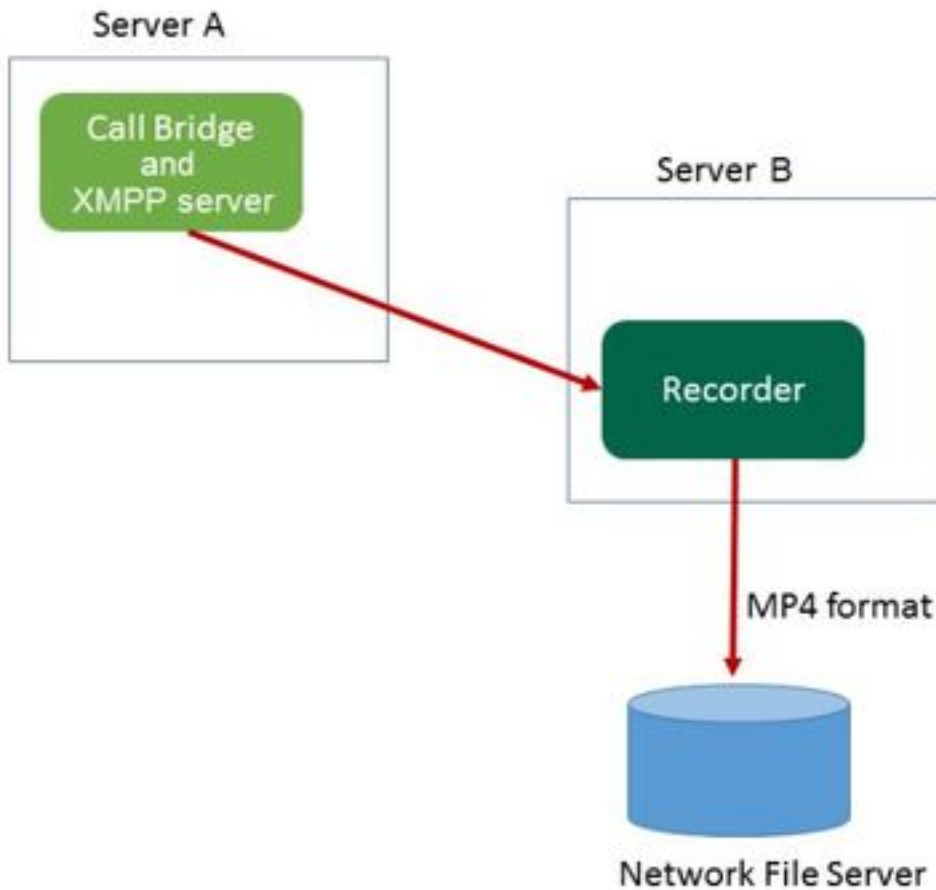
注: Windows サーバ 2008 R2 で権限問題のためのホットフィックスをそこに実行する NFS に関しては: <https://support.microsoft.com/en-us/kb/2485529>。

---

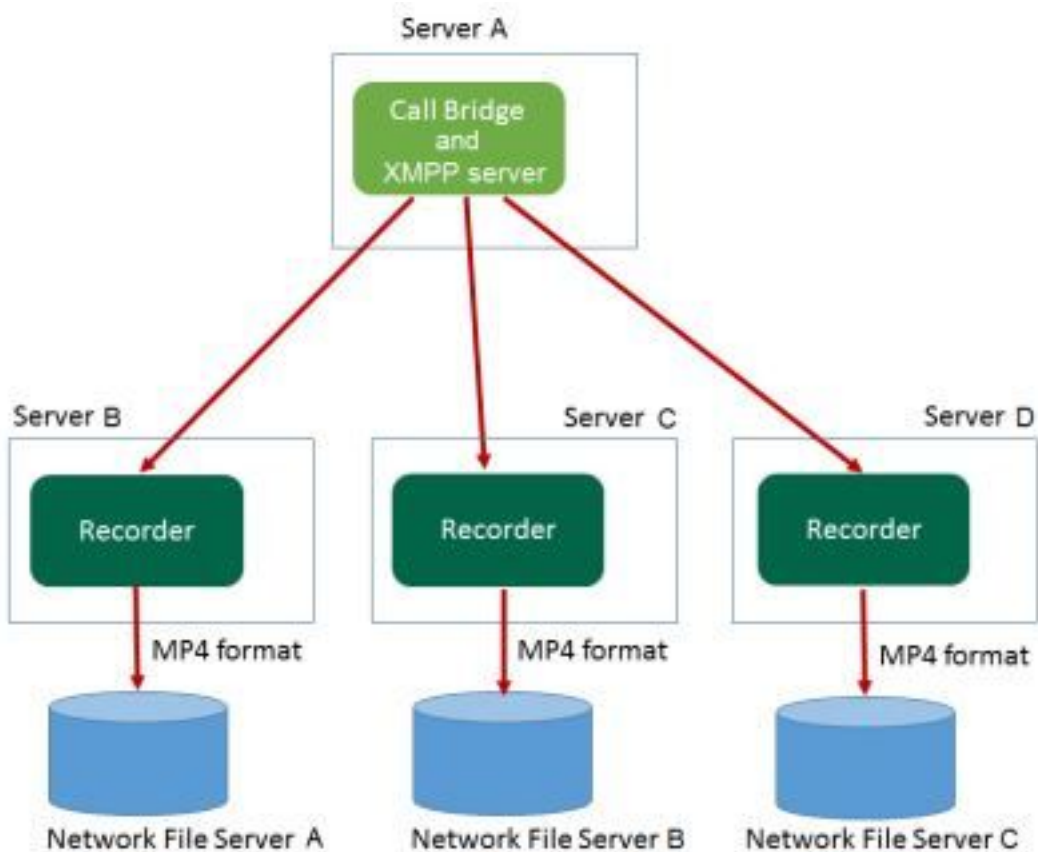
## 配備

### サポートされた配備

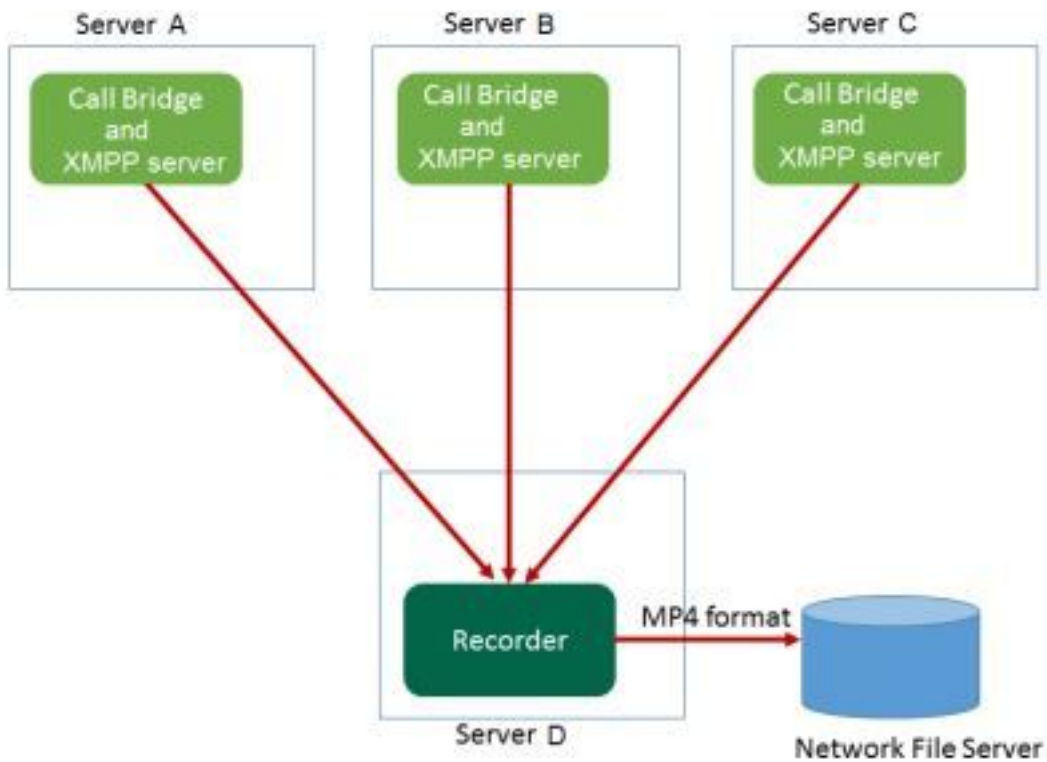
1. サーバにリモートであるレコーダーは Acano サーバでホストする必要がありますこの図に示すように CB をホストする:



2. レコーダーの冗長な配備はまたサポートされます。冗長性が設定される場合、記録物はすべての記録デバイス（サーバ）の間でバランスをとられるロードです。これはこの図が示すように各 CB が利用可能な各レコーダーを使用することを意味します：

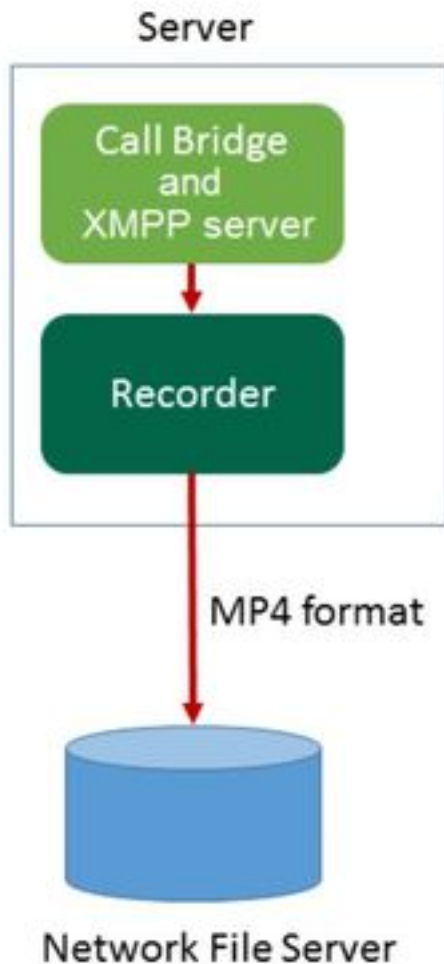


3. 同じは反対で複数の CB があるとき、適用します。すべての CB はそれらにこの図が示すように、レコーダー available を使用します:



## 他のセットアップ

レコーダーはまた CB と同じサーバでホストすることができますがこれはテストにしか使用する必要がありませんまたは非常に小さい配備は、図を参照します。ここの欠点は 1-2 に simultaneous 記録物を作れるただかもしれないことです:



## 設定

パート 1: レコーダー サーバ:

a. このコマンドでレコーダーを選択のインターフェイスで受信するために設定して下さい:

レコーダー リッスン <interface [: ポート] whitelist>

b. レコーダーがローカル CB にある場合、インターフェイスは「ループバック」へセットである必要があります従ってこのコマンドを使用して下さい:

レコーダー リッスン lo:8443

c. それが特定のインターフェイスで受信することである場合私達を「a」を言うことを許可して下さいこれを使用します:

レコーダー リッスン a:8443

---

注: クラスタ化された CB のノードのレコーダーを設定する場合、インターフェイスはレコーダーが設定されているノードのローカル受信インターフェイスであるはずでず。

---

d. レコーダーが使用される証明書ファイルを設定して下さい。既に存在およびプライベートキ

ーファイルが CB によって使用した証明書を、使用できますたとえば。

## レコーダー証明書 <keyfile> <certificatefile>

e. コマンドを使用してレコーダー信頼ストアに CB 証明書を追加して下さい:

## レコーダー信頼 <crt-bundle>

crt バンドルはもし違ったら CB によって使用される証明書が含まれているはずですが。 クラスタで、これはクラスタで各 CB の証明書が含まれている必要があります。

f. 記録物を保存するために NFS の NFS のホスト名か IP アドレス、およびディレクトリを規定して下さい:

## レコーダー nfs <ホスト名 /IP>: <directory>

---

注: レコーダーは NFS に認証しませんが、レコーダー サーバに NFS ディレクトリに読み書きアクセスがあることは重要です。

---

g. コマンドの使用のレコーダーを、有効に して下さい:

## レコーダー イネーブル

パート 2 : CB:

CB の API ユーザを、これ必要とされます API 機能を使用してそれ以上のコンフィギュレーションに作成して下さい:

これらのステップでユーザを作成して下さい:

a. セキュア シェル ( SSH ) によって接続するか、または admin 資格情報の使用の CB にコンソール接続を行って下さい。

b. ユーザは <username> API を追加し、そしてリターン キーを押し、リターン キーに先行しているパスワードを入力します。

パート 3 : API の使用:

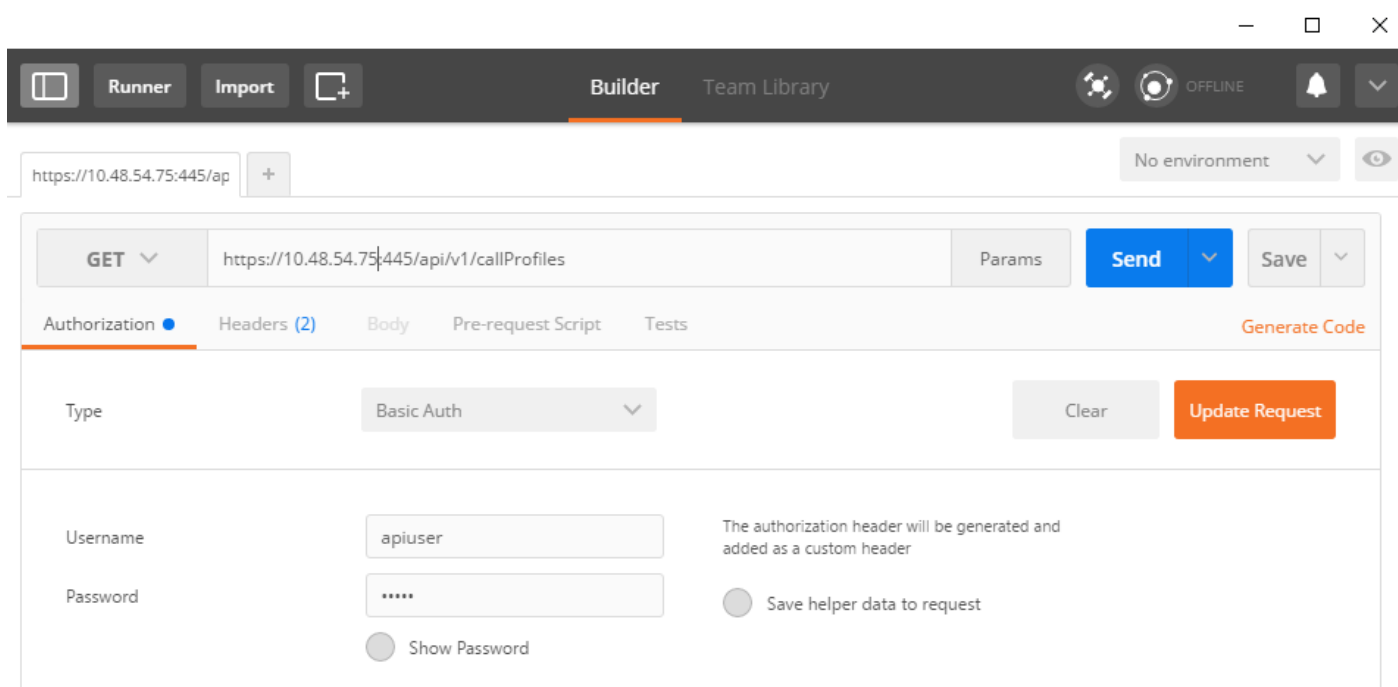
1. ダウンロードおよびインストール Postman からの

; <https://chrome.google.com/webstore/detail/postman/fhbjgbiflinjbdggehcddcbncdddomop?hl=en>

2. アドレスバーで API アクセス URL を、入力して下さいたとえば:

`https://<Callbridge_IP>:445/api/v1/<entity>`

それから、認証のセット、型として基本的な Auth の許可の下の Part 2 からのユーザ名 および パスワード、:



注: これは CB で設定される現在レコーダーまたは callProfile ないと仮定します。PUT の方式の使用とその既存および/または callProfile さもなければレコーダーを修正できます。

3. API の CB にレコーダーを追加して下さい:

a. <https://<Callbridge IP>:445/api/v1/recorders> の空 POST を送信して下さい

b. ( a ) の同じ URL の GET を送信して下さい、引用符なしで Notepad にレコーダー ID を、コピーして下さい

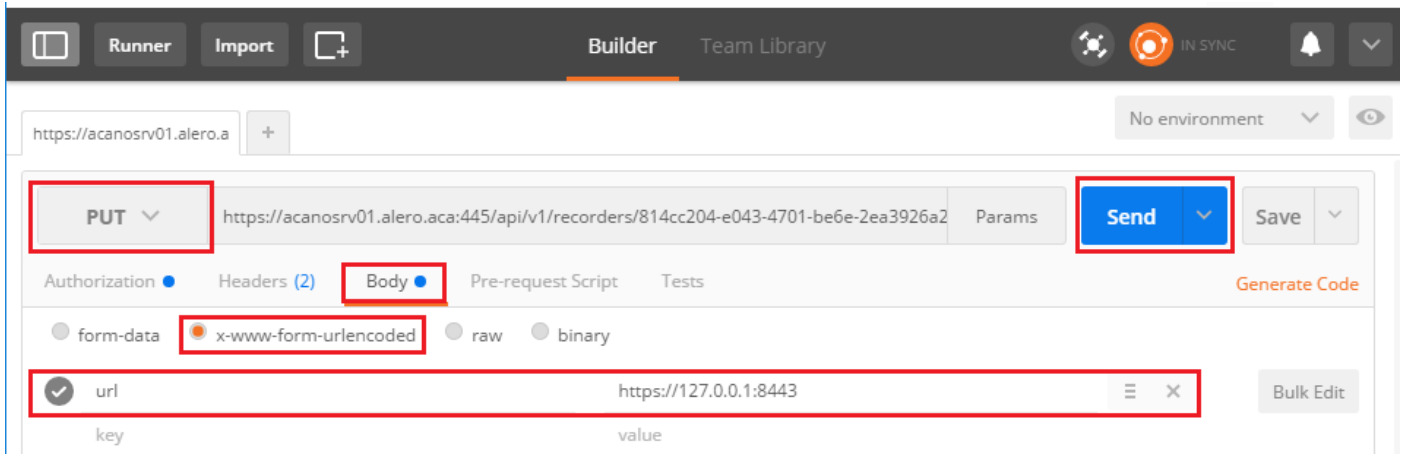
c. レコーダーの URL を送信によって PUT を実行する前に <https://<Callbridge IP>:445/api/v1/recorders/<recorderid>> の PUT 設定し、本文でこれを追加して下さい:

( レコーダーがローカル CB にあれば ) url= <https://127.0.0.1:8443>

または

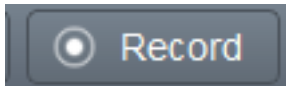
( レコーダーがローカル CB になければ ) recorder>:8443 の url= https:// < IP アドレス

次に、例を示します。



注: *dtmfProfile*、*callProfile* および *callLegProfile* cospace 会議に参加する SIP エンドポイントのために特に重要でござって下さい。それらはエンドポイントが記録有能なスタート/ストップ cospace に入出するコールのであるようにします。

CMA 1.9.3 および CMS 2.0.1 からように、DTMF トーンがそこにですレコーダーが



またはクライアントが接続される callbridge に知られていて時クライアントに追加されるボタン今必要となりません。レコード ボタンは CMS 2.3 から Webrtc に同様に追加されました。

4. callProfile 作成して下さい:

a. `https://<Callbridge_IP>:445/api/v1/callProfiles` の空 POST を送信して下さい

b. ( a ) の同じ URL の GET を送信して下さい、引用符なしで Notepad に callProfile ID を、コピーして下さい

c. callProfile の `recordingMode` を送信によって PUT を実行する前に `https://<Callbridge_IP>:445/api/v1/callProfiles/<call プロファイル ID>` の PUT 設定し、本文の追加して下さい。

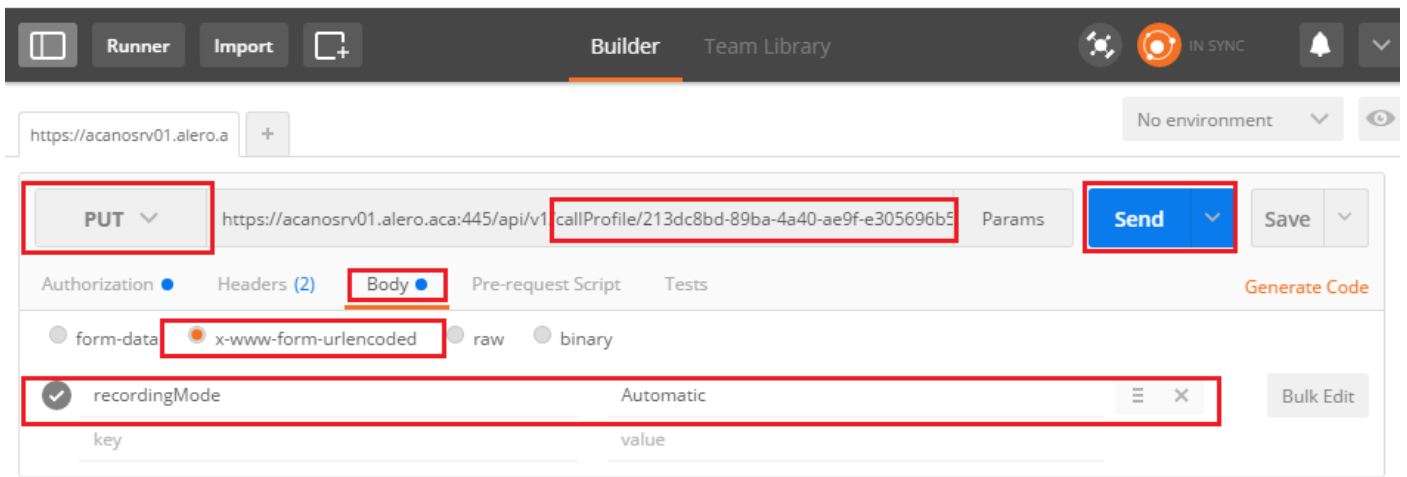
( 発信者に DTMF エントリを使用して記録し始めてほしければ ) `recordingMode=Manual`

または

コールが起動する時 ( 記録が自動的に開始すれば `recordingMode=Automatic` )

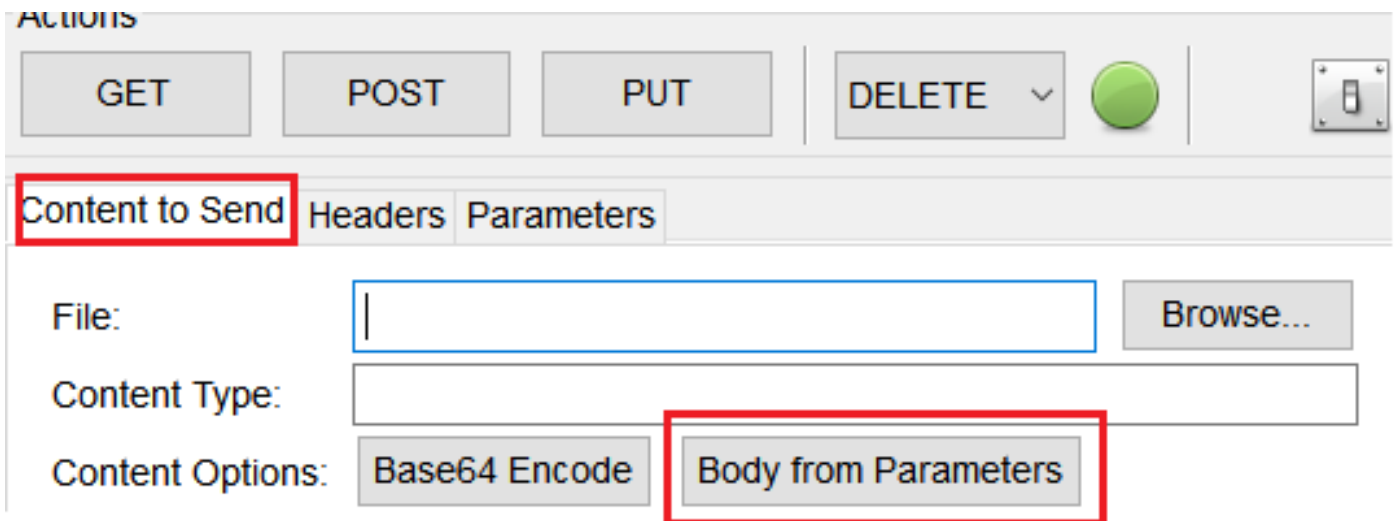
次に、例を示します。





注: firefox からのポスターを使用する場合、を送信するために「コンテンツ クリックしなければなりませんそれから選択します「パラメータからの本文」を

PUT/POST を送信する前、理解する CB が場合があるコードでコンパイルしたこうすれば。



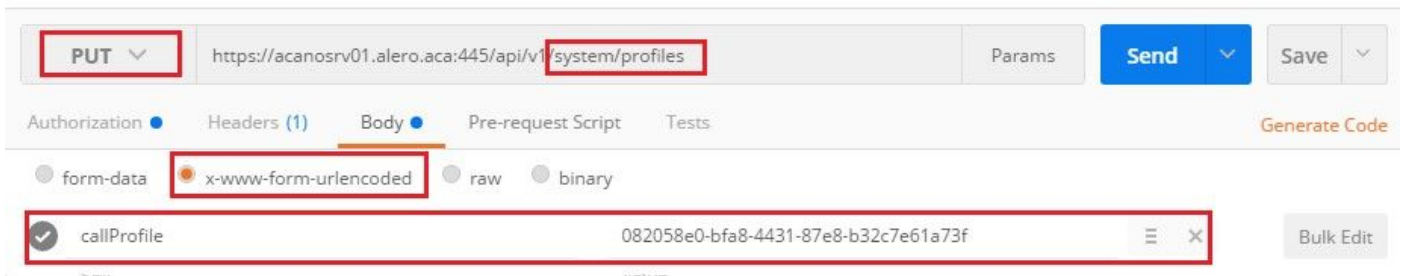
5. システム プロファイルにコール プロファイルを追加して下さい:

コールが記録物である場合もあるかどうか、そしてユーザ 介入の有無にかかわらずすることができれば callProfile 定義します。

本文の callProfile 追加した後 <https://<Callbridge IP>:445/api/v1/system/profiles> の PUT を送信して下さい

callProfile=<call プロファイル ID>

次に、例を示します。

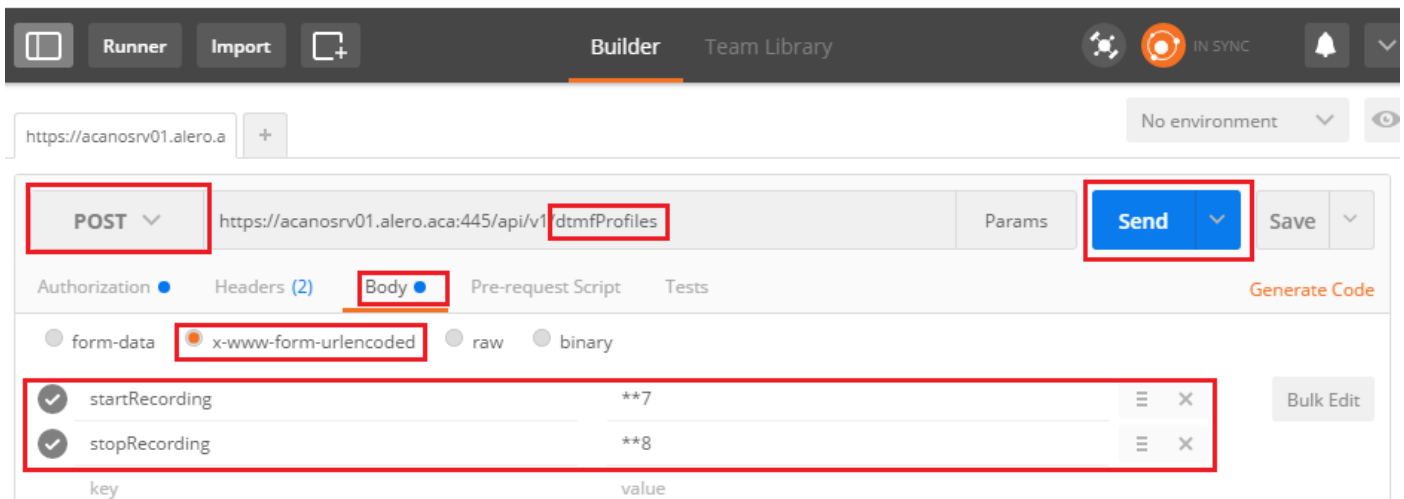


recordingMode が手動に設定される場合、ユーザがどのように DTMF トーンを使用して記録物を開始し、停止できるか定義するために DTMF プロファイルを設定して下さい。

## 6. DTMF プロファイルを作成して下さい:

a. startRecording=\*\*7&stopRecording=\*\*8 として本文の startRecording=\*\*7 および stopRecording=\*\*8 を (たとえば) 設定した後 [https:// <Callbridge\\_IP>:445/api/v1/dtmfProfiles](https://<Callbridge_IP>:445/api/v1/dtmfProfiles) のポストを送信して下さい。

例 :



b. 新しい DTMF プロファイルを見るために GET を送信しそして引用符なしでテキストエディタに ID をコピーして下さい。

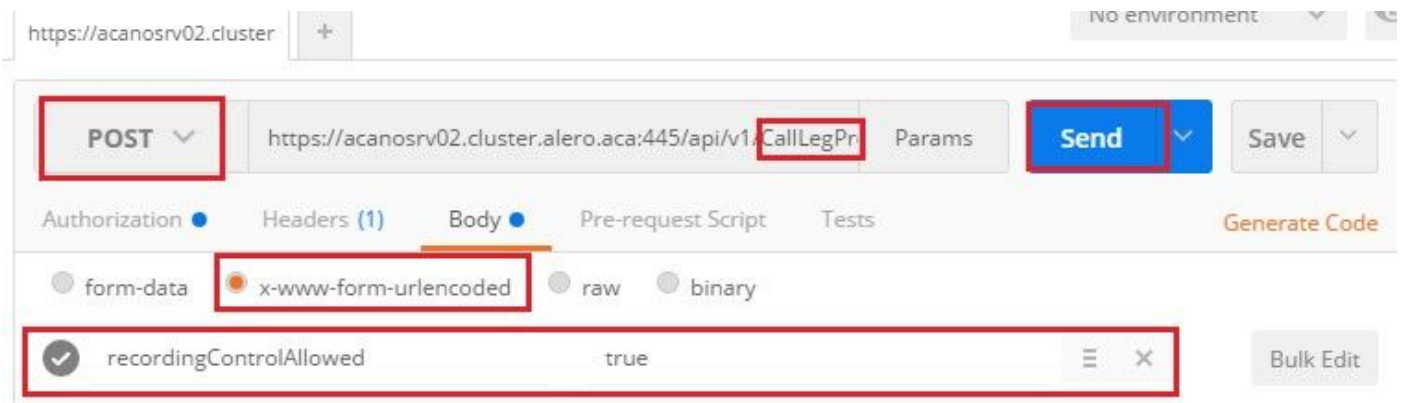
## 7. CallLeg プロファイルを作成して下さい:

CallLegProfiles は内部コール 動作を確認します。この場合コールが記録することができたかどうか確認します。

次の通りコールレグ プロファイルを作成して下さい:

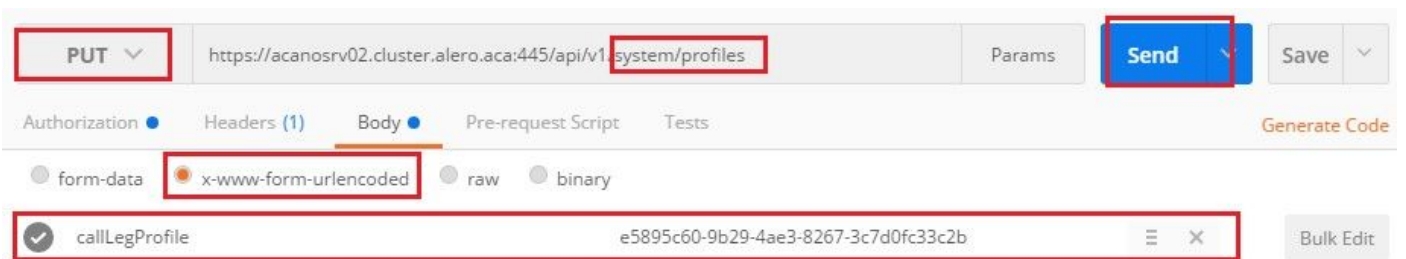
a. 本文の recordingControlAllowed=true を追加した後 [https://<Callbridge\\_IP>:445/api/v1/CallLegProfiles](https://<Callbridge_IP>:445/api/v1/CallLegProfiles) のポストを送信して下さい:

次に、例を示します。



b. [https://<Callbridge\\_IP>:445/api/v1/system/profiles](https://<Callbridge_IP>:445/api/v1/system/profiles) の PUT 送信し、本文の callLegProfile=<callLegProfile\_ID> を追加することによって CallLegProfile を、適用して下さい:

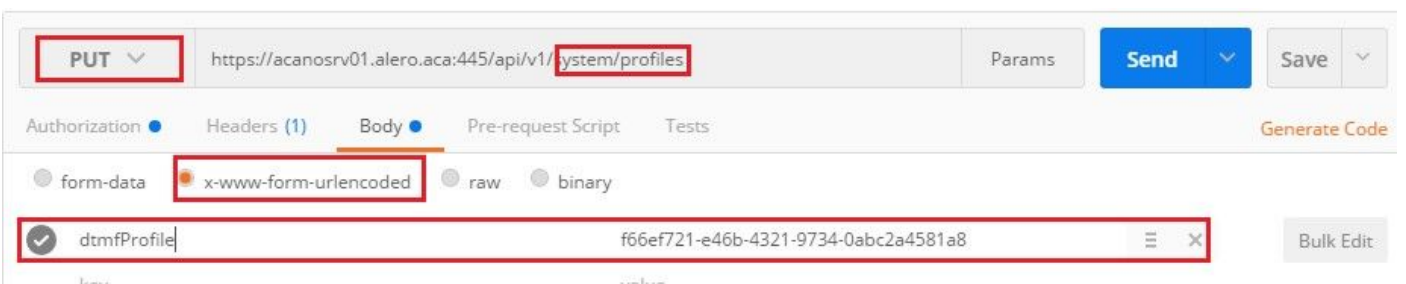
例 :



8. DTMF プロファイルを適用して下さい:

本文 dtmfProfile=<dfmt プロファイル ID> の dtmfProfile 追加した後 [https://<Callbridge\\_IP>:445/api/v1/system/profiles](https://<Callbridge_IP>:445/api/v1/system/profiles) の PUT を送信して下さい

次に、例を示します。



## 確認とトラブルシューティング

1. 設定される、これらのコマンドでステータスを、それこの出力に類似したべきですチェックして下さい:

レコーダー

ローカル スタンドアロン CB:

```
acanosrv01> recorder
Enabled                : true
Interface whitelist    : lo:8443
Key file               : callbridgecert.key
Certificate file       : callbridgecert.cer
Trust bundle          : callbridgecert.cer
NFS domain name       : 10.48.36.246
NFS directory         : /acano
```

またはクラスタ化された CB 場合:

```
acanosrv05> recorder
Enabled                : true
Interface whitelist    : a:8443
Key file               : forallcert05.key
Certificate file       : forallcert05.cer
Trust bundle          : TrustBundle.crt
NFS domain name       : 10.48.36.246
NFS directory         : /cluster-alero-aca-recordings
```

2. システム プロファイルを表示するために GET を結果の callProfile、CallLegProfile および dtmfProfile 下記のものを見るべきです送信して下さい:

[https:// <Callbridge\\_IP>:445/api/v1/system/profiles](https://<Callbridge_IP>:445/api/v1/system/profiles)

次に、例を示します。

```
1 <?xml version="1.0"?>
2 <profiles>
3   <callLegProfile>9591bd29-dc78-4656-bab1-328b2fd505fe</callLegProfile>
4   <callProfile>cf8cf197-a314-4c2e-93d5-4400551efcd6</callProfile>
5   <dtmfProfile>110ed4b0-fcb2-45e1-9b5c-724f7b037b35</dtmfProfile>
6 </profiles>
```

3. 設定されるものが CallProfile でチェックするために、API でこれを使用して下さい:

[https:// <Callbridge\\_IP>:445/api/v1/callProfiles/<callProfile\\_ID>](https://<Callbridge_IP>:445/api/v1/callProfiles/<callProfile_ID>)

これは記録がメソッド、示されているように自動が手動設定されたことを示します:

```
<?xml version="1.0"?>
<callProfile id="af73f145-829b-42ed-898d-f111f6259626">
  <recordingMode>automatic</recordingMode>
</callProfile>
```

4. 設定されるものが CallLegProfile でチェックするために、この API を使用して下さい:

[https:// <Callbridge\\_IP>:445/api/v1/callLegProfiles/<callLegProfile\\_ID>](https://<Callbridge_IP>:445/api/v1/callLegProfiles/<callLegProfile_ID>)

出力例 :

```
1 <?xml version="1.0"?>
2 <callLegProfile id="9591bd29-dc78-4656-bab1-328b2fd505fe">
3   <recordingControlAllowed>true</recordingControlAllowed>
4 </callLegProfile>
```

5. 設定されるものが DTMF プロファイルでチェックするために、API でこれを使用して下さい:

[https:// <Callbridge\\_IP>:445/api/v1/dtmfProfiles/<dtmfProfile\\_ID>](https://<Callbridge_IP>:445/api/v1/dtmfProfiles/<dtmfProfile_ID>)

これはメソッドを記録することが示されているように、自動か手動、設定されたことを示します:

```
<?xml version="1.0"?>
<dtmfProfile id="110ed4b0-fcb2-45e1-9b5c-724f7b037b35">
  <muteSelfAudio></muteSelfAudio>
  <unmuteSelfAudio></unmuteSelfAudio>
  <toggleMuteSelfAudio></toggleMuteSelfAudio>
  <lockCall></lockCall>
  <unlockCall></unlockCall>
  <muteAllExceptSelfAudio></muteAllExceptSelfAudio>
  <unmuteAllExceptSelfAudio></unmuteAllExceptSelfAudio>
  <endCall></endCall>
  <nextLayout></nextLayout>
  <previousLayout></previousLayout>
  <startRecording>**7</startRecording>
  <stopRecording>**8</stopRecording>
  <allowAllMuteSelf></allowAllMuteSelf>
  <cancelAllowAllMuteSelf></cancelAllowAllMuteSelf>
  <allowAllPresentationContribution></allowAllPresentationContribution>
  <cancelAllowAllPresentationContribution></cancelAllowAllPresentationContribution>
  <muteAllNewAudio></muteAllNewAudio>
  <unmuteAllNewAudio></unmuteAllNewAudio>
  <defaultMuteAllNewAudio></defaultMuteAllNewAudio>
  <muteAllNewAndAllExceptSelfAudio></muteAllNewAndAllExceptSelfAudio>
  <unmuteAllNewAndAllExceptSelfAudio></unmuteAllNewAndAllExceptSelfAudio>
</dtmfProfile>
```

---

注: DTMF プロファイルはポイント ツー ポイント コールではたきません、従って領域でしか手動記録を使用できません。

---

6. 記録されているものがレコーダーに関して表示するために、コマンドを実行して下さい:

Syslog は続きます

この出力と同じような何かを見るはずです:

kern.info 6 月 20 日 20:38:49 acanosrv05 レコーダー-proxy[1]: 10.48.54.75:39439 からの 2016/06/20 の 20:38:49 接続:

成功する認証

kern.info 6 月 20 日 20:38:49 acanosrv05 レコーダー-proxy[1]: 10.48.54.75:39439 からの 2016/06/20 の 20:38:49 接続:

終わる接続

kern.info 6 月 20 日 20:38:53 acanosrv05 レコーダー-proxy[1]: 10.48.54.76:35141 からの 2016/06/20 の 20:38:53 接続:

成功する認証

kern.info 6 月 20 日 20:38:53 acanosrv05 レコーダー-proxy[1]: 10.48.54.76:35141 からの 2016/06/20 の 20:38:53 接続:

終わる接続

この例で acanosrv05 はホスト サーバー レコーダーであり、それに接続される他の CB ノードは



10.48.54.75 および 10.48.54.76 です。

これは示しまりモード CB がであることをレコーダーによって正しく接続し、認証します。

レコーダーが CB にローカルである場合、接続はループバック IP から来ます:

kern.info 6 月 20 日 20:40:52 acanosrv01 レコーダーproxy[1]: 127.0.0.1:45380 からの  
2016/06/20 の 20:40:52 接続:

成功する認証

kern.info 6 月 20 日 20:40:52 acanosrv01 レコーダーproxy[1]: 127.0.0.1:45380 からの  
2016/06/20 の 20:40:52 接続:

終わる接続

---

注: レコーダー プロセスに関するほとんどのログはレコーダー プロキシとして Syslog で、これら  
ら与えますレコーダーが壊れるかもしれない示す値を示されています。

---

他の Syslog は次の通りレコーダーのために示されています:

この場合記録デバイスは発見され、記録は自動的に開始します:

「user.info 6 月 20 日 21:16:19 acanosrv02 ホスト: サーバ: INFO: 記録デバイス 1: 利用可能 ( 1  
記録物 ) 」

記録デバイスが発見される場合記録がチェックそれから失敗した:

「user.info 6 月 20 日 21:16:19 acanosrv02 ホスト: サーバ: INFO: 」発見される記録デバイス無  
し。

そのような警告が表示される場合、それを確認するためにレコーダー信頼の証明書をです CB に  
よって使用される正しい 1 チェックして下さい。

NFS ストレージがマウントされるかどうか Syslog をチェックして下さい:

NFS ストレージがマウントされない場合、見ます「NFS ストレージ」をマウントされなくて。

レコーダー サーバの NFS フォルダおよびセットことを確認して下さい: /Folder 名前は設定され  
る何が NFS ストレージで同じです。

アラームをチェックするために API を実行して下さいレコーダーに関連付ける:

[https://<callBridge\\_IP>api/v1/system/alarms](https://<callBridge_IP>api/v1/system/alarms)

低いディスクスペースがあれば「recorderLowDiskSpace」を見るはずで。

それからレコーダーによって参照される NFS ストレージに十分なディスク空間があることを確認  
して下さい。