CHAPTER 7

PacketCable 音声設定

この章では、PacketCable 音声を配備して使用するために実行する必要がある作業について説明します。

この章は、PacketCable の次の使用形態に関する情報を記載しています。

- PacketCable eMTA のセキュア プロビジョニング（P.7-2）
- PacketCable eMTA の Basic プロビジョニング（P.7-31）
- Euro PacketCable（P.7-33）

PacketCable 音声技術の配備における問題を解決するための情報については、P.16-12 の「PacketCable eMTA プロビジョニングのトラブルシューティング」を参照してください。

この章は、PacketCable Multimedia Terminal Adapter（MTA; マルチメディア ターミナル アダプタ）デバイスのプロビジョニング仕様、PKT-SP-PROV1.5-I03-070412 の内容を熟知している読者を対象としています。詳細については、PacketCable の Web サイトを参照してください。

PacketCable eMTA のセキュア プロビジョニング

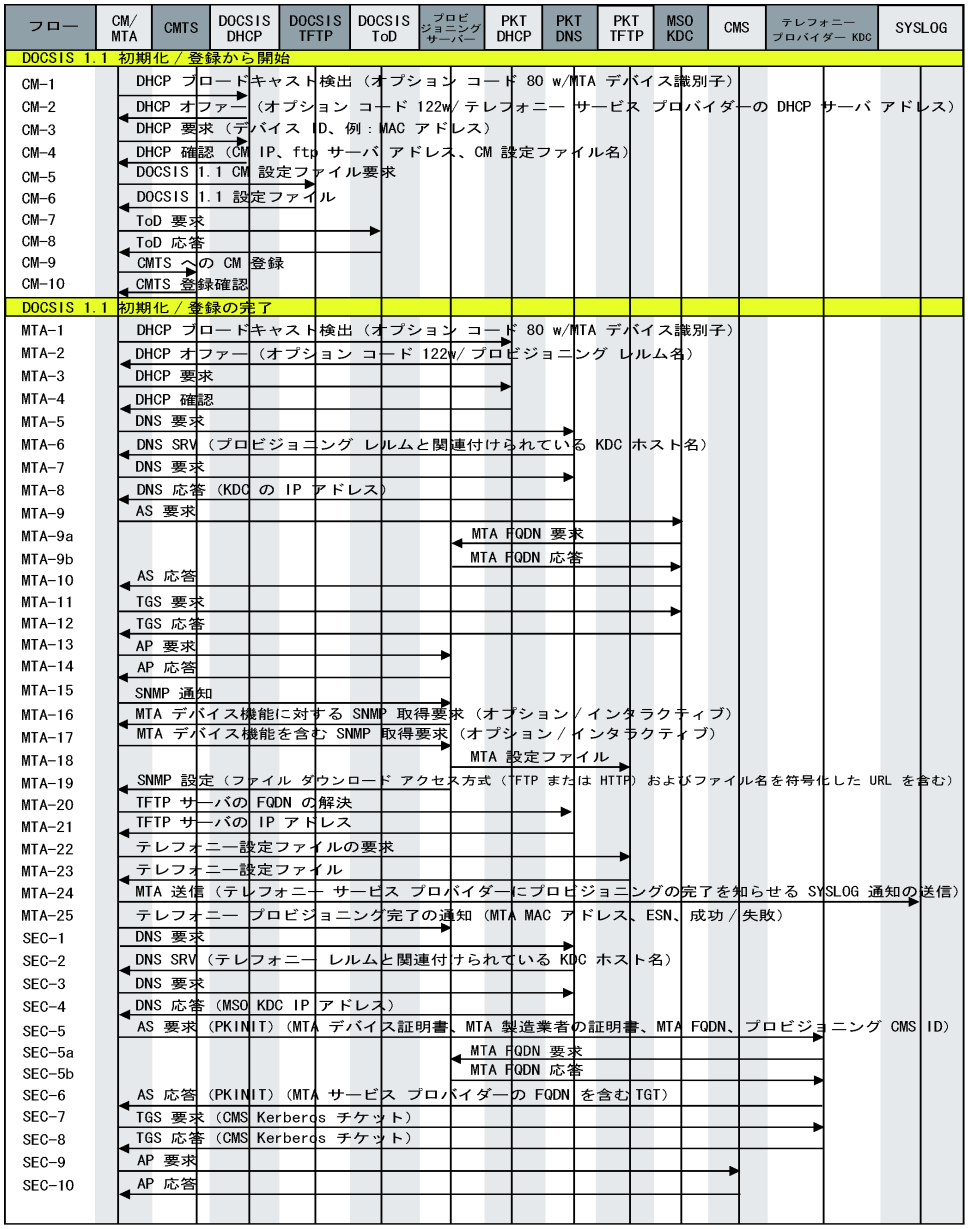
この項では、主にセキュアな PacketCable 音声プロビジョニングについて取り上げます。PacketCable Secure は、テレフォニー サービスの盗用や、悪意のあるサービス中断などが発生する可能性を最小限に抑えるように設計されています。PacketCable Secure では、Kerberos インフラストラクチャを利用して、MTA とプロビジョニング システムを相互に認証します。BAC では、Key Distribution Center（KDC; 鍵発行局）は Kerberos サーバとして機能します。MTA とプロビジョニング システム間の対話をセキュリティで保護するために、SNMPv3 も使用されます。

BAC PacketCable のセキュアなプロビジョニングのフロー

PacketCable プロビジョニングのすべてのフローは、一連のステップとして定義されます。

図 7-1 に、PacketCable eMTA のセキュアなプロビジョニングのフローを示します。

図 7-1　組み込み型 MTA のセキュアなパワーオン プロビジョニング フロー



（注）　データ パケットをキャプチャできるプロトコル アナライザ（プロトコル スニファ）を使用して、どのステップが失敗しているかを正確に把握することを強くお勧めします。

また、KDC の障害の根本的な原因を把握するには、KDC ログ ファイルの内容が重要です。

embedded Multimedia Terminal Adapter（eMTA; 組み込み型マルチメディア ターミナル アダプタ）のプロビジョニングにおける問題を診断する場合、表 7-1 のフローの説明が、PacketCable の失敗しているプロビジョニング フローのステップを特定するのに役立ちます。

表 7-1　PacketCable eMTA のセキュア プロビジョニング

手順	ワークフロー	説明
CM-1	DHCP ブロードキャスト検出	これは、DHCPv4 または DHCPv6 の DOCSIS ケーブル モデム（CM）ブート フローと同様で、PacketCable DHCP サーバのリストを MTA に提供するための DHCP オプションが付加されます。MTA はこれらの DHCP サーバから DHCP オファーを受け付けられるようになります。
CM-2	DHCP オファー	
CM-3	DHCP 要求	
CM-4	DHCP 確認	
CM-5	DOCSIS 1.1 CM 設定ファイル要求	
CM-6	DOCSIS 1.1 設定ファイル	
CM-7	ToD 要求	
CM-8	ToD 応答	
CM-9	CMTS（ケーブル モデム ターミネーション システム）への CM の登録	
CM-10	CMTS 登録確認応答	

表 7-1　PacketCable eMTA のセキュア プロビジョニング（続き）

手順	ワークフロー	説明
MTA-1	DHCP ブロードキャスト検出	DHCP を使用して、MTA が自身を PacketCable MTA としてアナウンスし、サポートしている機能およびプロビジョニング フロー（Secure、Basic など）についての情報を提供します。MTA はアドレス解決情報と DHCP Option 122 も取得します。DHCP Option 122 は、PacketCable のプロビジョニング サーバのアドレスとセキュリティ レルム名を含んでいます。この情報は、MTA から KDC およびプロビジョニング サーバへのアクセスを可能にするために使用されます。 基本的なトラブルシューティングのヒントを次に示します。 • CMTS 上で DHCP リレー エージェントが正しく設定されていることを確認する。CMTS が正しい DHCP サーバをポイントしていることを確認します。 • MTA、CMTS、DHCP サーバ、DPE 間のルーティングが正しいことを確認する。 • セカンダリ サブネットが CMTS 上で正しく設定されていることを確認する。 • Cisco Network Registrar の DHCP 設定が正しいことを確認する。スコープが設定され、IP アドレスが利用可能であり、すべてのセカンダリ サブネットが設定されていることを確認します。 • BAC の設定を確認する。*cnr_ep.properties* ファイルを確認して、必要な PacketCable Network Registrar 拡張のプロパティが設定されていることを確認します。詳細については、付録 C「PacketCable DHCP オプションと BAC プロパティのマッピング」を参照してください。 MTA がフロー ステップの MTA-1 と MTA-2 間で循環していることをパケット トレースが発見した場合は、DHCP Option 122（レルム名またはプロビジョニング サーバ FQDN のサブオプション）、DHCP Option 12（ホスト名）、または DHCP Option 15（ドメイン名）の設定に問題がある可能性があります。
MTA-2	DHCP オファー	
MTA-3	DHCP 要求	
MTA-4	DHCP 確認	
MTA-5	DNS 要求	MTA が（DHCP Option 122 で提供される）セキュリティ レルム名を使用して、KDC サービスに対して DNS SRV ルックアップを実行し、KDC の IP アドレスを解決します。 基本的なトラブルシューティングのヒントを次に示します。 • パケット スニファを使用して、Network Registrar DNS に送信される、送信先や形式が不正な DNS パケットを検出する。 • Network Registrar DNS のログ レベルをパケット詳細トレースに設定して、Network Registrar DNS にどのようなパケットが到達するかを確認する。 • DNS 設定を確認する：*cnr_ep.properties* に指定されている DNS サーバは、KDC のレルム ゾーン、SRV レコード、および DNS「A」レコードを保持している必要があります。
MTA-6	DNS Srv	
MTA-7	DNS 要求	
MTA-8	DNS 応答	
MTA-9	AS 要求	AS-REQ 要求メッセージが KDC によって使用され、MTA が認証されます。 基本的なトラブルシューティングのヒントを次に示します。 • KDC のログ ファイルを確認して、AS-REQ が到達しているかどうかを判断し、エラーや警告がないことを確認する。 • KDC が正しい MTA_Root 証明書を使用して設定されていることを確認する。MTA が AS-REQ メッセージに添付して送信する製造業者証明書およびデバイス証明書は、KDC にインストールされている MTA_Root 証明書とチェーンを構成する必要があります。

表 7-1　PacketCable eMTA のセキュア プロビジョニング（続き）

手順	ワークフロー	説明
MTA-9a	MTA FQDN 要求	KDC が MTA の MAC アドレスを MTA 証明書から抽出して、検証のためにプロビジョニング サーバに送信します。プロビジョニング サーバがこの MAC アドレスの FQDN を保持している場合は、FQDN が KDC に返されます。KDC は MTA から受信した FQDN を FQDN-REP 応答メッセージで受信した FQDN と比較します。 基本的なトラブルシューティングのヒントを次に示します。 • パケット スニファを使用して、送信先や形式が不正な DNS パケットを検出する。MTA は、（MTA が DHCP Option 122 で受信した）プロビジョニング サーバの FQDN を AS-REP メッセージ内で KDC に渡します。KDC はこの FQDN を使用して、プロビジョニング サーバの IP アドレスを解決します。 • KDC キー ファイルのファイル名と内容を確認する。DPE 内の KDC サービス キーは、KDC にあるサービス キーと一致している必要があります。KDC にあるサービス キー ファイルの名前は、非常に重要です。
MTA-9b	MTA FQDN 応答	
MTA-10	AS 応答（AS-REP）	KDC がプロビジョニング サービス チケットを MTA に付与し、サービス プロバイダー証明書、ローカル システム プロバイダー証明書（オプション）、および KDC 証明書を MTA に送信します。MTA は KDC から送信された証明書が、MTA に格納されているサービス プロバイダーのルート証明書とチェーンを構成していることを確認します。これらの証明書がチェーンを構成していない場合、MTA はプロビジョニング フローのステップ MTA-1 に処理を戻します。*KDC.cer* ファイルの詳細については、P.14-3 の「PKCert.sh ツールの使用方法」を参照してください。 基本的なトラブルシューティングのヒントを次に示します。KDC のログ ファイルを表示して、AS-REP メッセージがデバイスに送信されたことを確認する。MTA がステップ MTA-1 ～ MTA-10 を循環していることをパケット トレースが発見した場合は、サービス プロバイダー証明書チェーンに問題があります。
MTA-11	TGS 要求	ステップ MTA-10 の後、MTA がサービス チケットまたは Ticket-Granting-Ticket（TGT; チケット認可チケット）を受信します。MTA はステップ MTA-10 で、サービス チケットの代わりに TGT を取得した場合、Ticket-Granting-Server（KDC）にアクセスして、サービス チケットを取得します。
MTA-12	TGS 応答	KDC が TGS 応答内のサービス チケットを MTA に送信します。
MTA-13	AP 要求（AP-REQ）	MTA が（ステップ MTA-10 で受信した）チケットを DHCP Option 122 で指定されているプロビジョニング サーバに提示します。
MTA-14	AP 応答（AP-REP）	プロビジョニング サーバが KDC 共有秘密情報を使用して AP-REQ を復号化し、MTA が提示したプロビジョニング サーバ チケットを検証して、AP-REP を SNMPv3 キーを使用して送信します。以後の SNMPv3 は認証済みになり、必要に応じて暗号化されます。
MTA-15	SNMP 通知	MTA が、プロビジョニング情報を受信可能なことをプロビジョニング サーバに通知します。
MTA-16	SNMP 取得要求	SNMPv3：プロビジョニング サーバ（DPE）が追加のデバイス機能を必要とする場合は、MTA に 1 つ以上の SNMPv3 取得要求を送信して、MTA 機能に関する必要な情報を取得します。プロビジョニング サーバ（DPE）は、一括取得要求を使用して、1 つのメッセージに大量の情報を要求することがあります。

表 7-1　PacketCable eMTA のセキュア プロビジョニング（続き）

手順	ワークフロー	説明
MTA-17	SNMP 取得応答	SNMPv3 : MTA が、ステップ MTA-16 で要求した MTA 機能に関する情報を含む各取得要求に対する応答をプロビジョニング サーバ（DPE）に送信します。
MTA-18	MTA 設定ファイル	ステップ MTA-16 と MTA-17 で利用可能にした情報を使用して、プロビジョニング サーバ（DPE）が MTA 設定データ ファイルの内容を判別します。
MTA-19	SNMP 設定	SNMPv3 : プロビジョニング サーバが、MTA 設定ファイルの URL、このファイルの暗号キー、およびこのファイルのハッシュ値を含んだ SNMPv3 設定を MTA に対して実行します。
MTA-20	TFTP サーバの FQDN の解決	DNS 要求 : URL 符号化アクセス方式に IPv4 アドレスの代わりに FQDN が含まれている場合、MTA はサービス プロバイダー ネットワークの DNS サーバを使用して、FQDN を解決して TFTP サーバまたは HTTP サーバの IPv4 アドレスにします。
MTA-21	TFTP サーバの IP アドレス	DNS 応答 : DNS サーバが、ステップ MTA-20 で要求したサービス プロバイダー ネットワークの IPv4 IP アドレスを返します。
MTA-22	テレフォニー設定ファイル要求	MTA は、指定された TFTP サーバから VoIP 設定ファイルをダウンロードする処理に進みます。BAC は、TFTP サーバを DPE コンポーネントに統合します。
MTA-23	テレフォニー設定ファイル	
MTA-24	MTA 送信	MTA がオプションで、プロビジョニングの完了を知らせる syslog 通知をサービス プロバイダーに送信します。
MTA-25	テレフォニー プロビジョニングの完了通知	MTA が、新しい設定の受け付けが可能かどうかをプロビジョニング サーバに通知します。
SEC-1 ～ SEC-10	これらのステップは、ポスト MTA プロビジョニングのセキュリティ フローのもので、BAC プロビジョニングには適用できません。このフローには、MTA が通信する各 CMS に関連付けられている Kerberos チケットの取得が含まれます。詳細については、PacketCable Security の仕様を参照してください。	

PacketCable eMTA のセキュア プロビジョニングにおける KDC

PacketCable Secure では、Kerberos インフラストラクチャを利用して、MTA とプロビジョニング システムを相互に認証します。BAC では、KDC は Kerberos サーバとして機能します。KDC コンポーネントの概要については、P.2-14 の「Key Distribution Center」を参照してください。

KDC に関する重要な情報については、次を参照してください。

KDC のデフォルト プロパティ

KDC には、BAC インストール中に *BPR_HOME/kdc/solaris/kdc.ini* プロパティ ファイルに入力される、いくつかのデフォルト プロパティがあります。このファイルを編集して、操作要件で指示された値に変更することができます。

（注）　動作要件を記述する場合は、*kdc.ini* ファイルの編集には注意してください。誤った値を指定すると KDC が動作しなくなる場合があります。変更を加えた場合は、KDC を再起動します。

デフォルトのプロパティは次のとおりです。

- interface address：KDC が着信 Kerberos メッセージを監視するローカルのイーサネット インターフェイスの IP アドレスを指定します。

  次に例を示します。

  ```
  interface address = 10.10.10.1
  ```

- FQDN：KDC がインストールされているコンピュータの Fully Qualified Domain Name（FQDN; 完全修飾ドメイン名）を示します。

  次に例を示します。

  ```
  FQDN = kdc.example.com
  ```

（注）　インターフェイス アドレスと FQDN 値は、インストール時に KDC Realm Name 画面から入力する必要があります。具体的な情報については、『*Installation and Setup Guide for Cisco Broadband Access Center 4.0*』を参照してください。

- maximum log file size：KDC が生成するログ ファイルの拡大可能な最大サイズを KB 単位で指定します。KDC は、現在のファイルがこの最大サイズに達した場合にのみ新しいログ ファイルを作成します。

  次に例を示します。

  ```
  maximum log file size = 1000
  ```

- *n* saved log files : KDC が保存する古いログ ファイルの数を定義します。デフォルト値は 7 です。必要な数だけ指定できます。

  次に例を示します。

  ```
  n saved log files = 10
  ```

- log debug level : ログ ファイルのロギング レベルを指定します。

  ```
  log debug level = 5
  ```

  表 7-2 では、KDC のログ ファイルで使用可能なロギング レベルについて説明します。

表 7-2　KDC のロギング レベル

ログ レベル	説明
0	エラー状態が存在します。すべてのエラー メッセージ、およびさらに深刻なメッセージを保存するように、ロギング機能を設定します。
1	警告状態が存在します。すべての警告メッセージ、およびさらに深刻なメッセージを保存するように、ロギング機能を設定します。
2	情報メッセージ。利用可能なすべてのロギング メッセージを保存するように、ロギング機能を設定します。
{*3-7*}	デバッグ メッセージ。レベル 3 からレベル 7 までの、さまざまなレベルのすべてのデバッグ メッセージを保存するように、ロギング機能を設定します。

- minimum (maximum) ps backoff : KDC が FQDN-Request に応答するために BAC を待つ最短（または最長）待ち時間を 0.1 秒単位で指定します。

  次に例を示します。

  ```
  minimum ps backoff = 150
  ```

上述のサンプル値を使用する、サンプル INI ファイルには、例 7-1 に表示されているのと同様のデータが含まれます。

例 7-1　kdc.ini 設定ファイルのサンプル

```
interface address =  10.10.10.1
FQDN = kdc.example.com
maximum log file size = 1000
n saved log files = 10
log debug level = 5
minimum ps backoff = 150
maximum ps backoff = 300
```

配備中に発生する可能性がある多数のチケット要求を効率的に処理するために、最短と最長のチケット継続期間のそれぞれに時間を設定できます。この設定は、ほとんどの配備が通常の業務時間内に行われ、ときどき過剰な負荷がかかり、パフォーマンスに悪影響を与える場合に有益です。

（注） チケット期間を短縮すると、MTA はより頻繁に KDC を認証するようになります。この結果、テレフォニー エンドポイントの認可をさらに管理できるようになりますが、KDC でのメッセージ負荷が増し、ネットワーク トラフィックが増えることにもなります。ほとんどの場合はデフォルトの設定が適しているため、変更しないでください。

- maximum ticket duration : KDC が生成するチケットの最長継続期間を定義します。デフォルトの単位は時間ですが、**m** または **d** を追加すると、単位をそれぞれ分または日に変更できます。

  デフォルト値は 168（7 日）です。この値は、PacketCable Security の仕様を確認するために必要な時間の長さであるため、この値は変更しないことをお勧めします。

  次に例を示します。

  ```
  maximum ticket duration = 168
  ```

- minimum ticket duration : KDC が生成するチケットの最短継続期間を定義します。デフォルトの単位は時間ですが、**m** または **d** を追加すると、単位をそれぞれ分または日に変更できます。

  デフォルト値は 144（6 日間）です。この値は変更しないことをお勧めします。

  次に例を示します。

  ```
  minimum ticket duration = 144
  ```

KDC 証明書

KDC の認証に使用される証明書は、BAC に同梱されていません。Cable Television Laboratories, Inc.（CableLabs）から必要な証明書を入手する必要があります。これらの証明書の内容は、MTA にインストールされている証明書の内容と一致している必要があります。

（注） 証明書がインストールされていないと KDC は機能しません。

PKCert ツールを使用して、KDC が動作するために必要な証明書をインストールして、管理することができます。PKCert ツールは、CableLabs サービス プロバイダー証明書を証明書ファイルとしてインストールします。このツールの実行方法の詳細については、P.14-3 の「PKCert.sh ツールの使用方法」を参照してください。

PKCert ツールは、KDC コンポーネントをインストールしている場合のみ利用できます。

KDC ライセンス

シスコの代理店から KDC ライセンスを入手して、正しいディレクトリにインストールしてください。

KDC ライセンス ファイルをインストールするには、次の手順に従います。

ステップ 1　シスコの代理店からライセンス ファイルを入手します。

ステップ 2　BAC ホストに *root* としてログインします。

ステップ 3　ライセンス ファイルをコピーします。

注意　ファイルを ASCII ファイルとしてコピーしないように注意してください。このファイルには、ASCII 転送中に不要な変更が加えられやすいバイナリ データが含まれています。

KDC ライセンス ファイルは転送プロセスにより損傷を受ける可能性があるため、異なるオペレーティング システム間でコピーすることはできません。

ステップ 4　KDC サーバを再起動して、変更を有効にするには、*/etc/init.d* ディレクトリから **bprAgent restart kdc** コマンドを実行します。

複数レルムのサポート

BAC KDC は複数レルムの管理をサポートします。この場合、有効な PacketCable X.509 証明書と KDC　　秘密鍵の完全なセットが存在する必要があります。これらの証明書は、*BPR_HOME/kdc/solaris/packetcable/certificates* ディレクトリに存在する必要があります。

BAC は、*BPR_HOME/kdc/solaris/packetcable/certificates* ディレクトリの下にサブディレクトリをインストールすることで、追加のレルムをサポートします。各サブディレクトリには、特定のレルムと同じ名前が付けられます。

表 7-3 に、さまざまな証明書とそれぞれに対応するファイル名を示します。これらのファイルは、*BPR_HOME/kdc/solaris/packetcable/certificates* ディレクトリに存在する必要があります。

表 7-3　PacketCable の証明書

証明書	証明書のファイル名
MTA のルート	*MTA_Root.cer*
サービス プロバイダーのルート	*CableLabs_Service_Provider_Root.cer*
サービス プロバイダーの CA	*Service_Provider.cer*
ローカル システム オペレータの CA	*Local_System.cer*
KDC	*KDC.cer*

プライマリ レルムは、KDC コンポーネントのインストール中に設定されます。プライマリ レルムの場合、KDC 証明書（*KDC.cer*）は、*BPR_HOME/kdc/solaris/packetcable/certificates* ディレクトリに存在します。その秘密鍵（KDC_private_key.pkcs8）は、*BPR_HOME/kdc/solaris/* ディレクトリにあります。

追加のレルムを設定するには、次の手順に従ってください。この後で詳細を説明します。

ステップ 1　KDC 証明書を含んでいるディレクトリを検索します。

ステップ 2　そのディレクトリの下に KDC 証明書を格納するサブディレクトリを作成します。

（注）　サブディレクトリの名前を特定のレルムの名前と一致させます。大文字のみを使用して、サブディレクトリの名前を付けます。

ステップ 3　作成したサブディレクトリに、レルムの KDC 証明書と秘密鍵を配置します。

ステップ 4　新しいレルムが KDC 証明書として同じサービス プロバイダーとチェーンを構成していない場合は、証明書ディレクトリにあるものとは異なる追加の上位レベル証明書をすべて含めます。

（注）　すべてのレルムは同じ証明書チェーンにルートがある必要があるため、KDC のインストールは一度に 1 つのロケール（北米版 PacketCable または欧州版 PacketCable）のみサポートします。

表 7-4 では、プライマリ レルム（CISCO.COM など）と 2 つのセカンダリ レルム（CISCO1.COM と CISCO2.COM など）のディレクトリ構造とファイルについて説明します。この構造は、上位レベルの証明書がプライマリ レルムとそのセカンダリ レルムで同様であることを想定しています。

表 7-4　複数レルムのディレクトリ構造

ディレクトリ	ディレクトリのファイルの内容
BPR_HOME/kdc/solaris	プライマリ レルム CISCO.COM の場合： KDC 秘密鍵
BPR_HOME/kdc/solaris/packetcable/certificates	プライマリ レルム CISCO.COM の場合： • *MTA_Root.cer* • *CableLabs_Service_Provider_Root.cer* • *Service_Provider.cer* • *Local_System.cer* • *KDC.cer* Directory /*CISCO1.COM* Directory /*CISCO2.COM*
BPR_HOME/kdc/solaris/packetcable/certificates/CISCO1.COM	セカンダリ レルム CISCO1.COM の場合： • *KDC.cer* • KDC 秘密鍵
BPR_HOME/kdc/solaris/packetcable/certificates/CISCO2.COM	セカンダリ レルム CISCO2.COM の場合： • *KDC.cer* • KDC 秘密鍵

複数レルムの KDC の設定

この項では、複数レルムの KDC を設定するためのワークフローについて説明します。処理を進める前に、RDU、DPE、および Network Registrar 拡張のインストールを完了してください。インストールの説明については、『*Installation and Setup Guide for the Cisco Broadband Access Center 4.0*』を参照してください。

次のワークフローでは、サンプルのレルムとディレクトリを使用して、複数レルムの KDC を設定する方法を説明します。ここで使用するプライマリ レルムは CISCO.COM、そのセカンダリ レルムは CISCO1.COM と CISCO2.COM です。

次のワークフローのセットアップでは、3 つの MTA（Motorola SBV 5120 MTA、Linksys CM2P2 MTA、および SA WebStar DPX 2203 MTA）をプロビジョニングします。各 MTA は 1 つのレルムでプロビジョニングされます。Motorola は CISCO.COM レルム、Linksys MTA は CISCO1.COM レルム、SA MTA は CISCO2.COM レルムです。

（注）　次の手順で示されている出力例は、説明を目的としているため短く編集されています。

複数レルムの KDC を設定するには、次の手順に従います。

ステップ 1　DPE 上の次の設定内容を確認します。

a. **show run** コマンドを使用して、PacketCable サービスがイネーブルになっていることを確認します。

PacketCable サービスをイネーブルにするには、**service packetcable** *1..1* **enable** コマンドを使用します。

次に例を示します。

```
dpe# show run
  aaa authentication local
  dpe port 49186
  dpe provisioning-group primary default
  service packetcable 1 enable
  snmp-server location equipmentrack5D
  snmp-server udp-port 8001
  tacacs-server retries 2
  tacacs-server timeout 5
```

コマンドの詳細については、『*Cisco Broadband Access Center DPE CLI Reference 4.0*』を参照してください。

b. **show run** コマンドを使用して、KDC と DPE 間の通信に使用するセキュリティが設定されていることを確認します。

セキュリティ鍵を生成して設定するには、**service packetcable** *1..1* **registration kdc-service-key** コマンドを使用します。

次に例を示します。

```
dpe# show run
  aaa authentication local
  debug dpe events
  dpe port 49186
  service packetcable 1 enable
  service packetcable 1 registration kdc-service-key <value is set>
  snmp-server contact AceDuffy-ext1234
```

コマンドの詳細については、『*Cisco Broadband Access Center DPE CLI Reference 4.0*』を参照してください。

c. PacketCable SNMPv3 クローニングに対する DPE と RDU 間での安全な通信を許可するセキュリティ鍵が設定されていることを確認します。再度、**show run** コマンドを使用します。セキュリティ鍵を生成して設定するには、**service packetcable** *1..1* **snmp key-material** コマンドを使用します。

次に例を示します。

```
dpe# show run
  aaa authentication local
  debug dpe events
  dpe port 49186
  service packetcable 1 enable
  service packetcable 1 registration kdc-service-key <value is set>
  service packetcable 1 snmp key-material <value is set>
```

コマンドの詳細については、『*Cisco Broadband Access Center DPE CLI Reference 4.0*』を参照してください。

（注）　DPE 上で PacketCable を設定する場合は、**dpe reload** コマンドを実行して変更を有効にしてください。

ステップ 2　Network Registrar 拡張ポイント（*cnr_ep.properties*）の設定ファイルで、**/ccc/kerb/realm** パラメータがプライマリ レルム（この場合は、CISCO.COM）で設定されているかどうかを確認します。確認を行うには、*BPR_HOME/cnr_ep/conf* ディレクトリから **more cnr_ep.properties** コマンドを実行します。

次に例を示します。

```
/opt/CSCObac/cnr_ep/conf# more cnr_ep.properties
#DO NOT MODIFY THIS FILE.
#This file was created on Wed, March 4 06:34:34 EDT 2007
/rdu/port=49187
/rdu/fqdn=dpe4.cisco.com
/cache/provGroupList=Default
/cnr/sharedSecret=fggTaLg0XwKRs
/pktcbl/enable=enabled
/ccc/tgt=01
/ccc/kerb/realm=CISCO.COM
/ccc/dhcp/primary=10.10.0.1
/ccc/dns/primary=10.10.0.1
```

ステップ 3　静的ルートを適切にイネーブルにして、BAC と CMTS の背後にあるデバイスとの接続を確保します。

ステップ 4　*cnr_ep.properties* ファイルにリストされている DNS サーバの DNS レルム ゾーンを作成します。ゾーンは、**DNS > Forward Zones > List/Add Zones** ページから Network Registrar の管理者のユーザ インターフェイスを使用して追加できます。

（注）　追加するゾーンには、KDC サーバの SRV レコードと DNS「A」レコードを含め、各ゾーン（この場合は、CISCO.COM、CISCO1.COM、および CISCO2.COM）の SRV レコードが 1 つの KDC をポイントするようにしてください。

管理者のユーザ インターフェイスを使用してゾーンを設定する方法については、『*User Guide for Cisco Network Registrar 7.0*』を参照してください。

ステップ 5　PKCert.sh ツールを使用して証明書を設定します。

a.　セカンダリ レルム（この場合は、CISCO1.COM と CISCO2.COM）のディレクトリを *BPR_HOME/kdc/solaris/packetcable/certificates* の下に作成します。

次に例を示します。

```
/opt/CSCObac/kdc/solaris/packetcable/certificates# mkdir CISCO1.COM
/opt/CSCObac/kdc/solaris/packetcable/certificates# mkdir CISCO2.COM
```

ディレクトリの作成方法の詳細については、Solaris のマニュアルを参照してください。

b.　次の証明書をコピーするディレクトリを作成します。

- *CableLabs_Service_Provider_Root.cer*
- *Service_Provider.cer*

- *Local_System.cer*
- *MTA_Root.cer*
- *Local_System.der*

次に例を示します。

```
# cd /var
# mkdir certsInput
```

（注） */var* ディレクトリの下に作成されている */certsInput* ディレクトリは、一例です。他のディレクトリの下に任意のディレクトリを作成するように選択できます。ディレクトリの作成方法の詳細については、Solaris のマニュアルを参照してください。

c. 前のステップで示した証明書を、作成したディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

d. 次の証明書を *BPR_HOME/kdc/solaris/packetcable/certificates* ディレクトリにコピーします。

- *CableLabs_Service_Provider_Root.cer*
- *Service_Provider.cer*
- *Local_System.cer*
- *MTA_Root.cer*

ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

e. プライマリ レルムの KDC 証明書とそれに関連付けられる秘密鍵を作成します。

次に例を示します。

```
# ./opt/CSCObac/kdc/PKCert.sh -c "-s /var/certsInput -d /var/certsOutput
-k /var/certsInput/Local_System.der -c /var/certsInput/Local_System.cer
-r CISCO.COM -n 100 -a bactest.cisco.com -o"
Pkcert Version 1.0
Logging to pkcert.log
Source Directory: /var/certsInput
Destination Directory: /var/certsOutput
Private Key File: /var/certsInput/Local_System.der
Certificate File: /var/certsInput/Local_System.cer
Realm: CISCO.COM
Serial Number: 100
DNS Name of KDC: bactest.cisco.com
WARNING - Certificate File will be overwritten
SP Cert subject name: C=US,O=CableLabs\, Inc.,OU=ABC Cable Company,CN=Shared-01
CableLabs Local System CA
File written: /var/certsOutput/KDC_private_key.pkcs8
File written: /var/certsOutput/KDC_private_key_proprietary.
File written: /var/certsOutput/KDC_PublicKey.der
File written: /var/certsOutput/KDC.cer
KDC Certificate Successfully Created at /var/certsOutput/KDC.cer

Copy KDC.cer to the KDC certificate directory (i.e. /opt/CSCObac/kdc/solaris/
packetcable/certificates)
Copy KDC_private_key.pkcs8 to the KDC platform directory (i.e. /opt/CSCObac/
kdc/solaris)
Copy KDC_private_key_proprietary. to the KDC platform directory (i.e. /opt/CSCObac/
kdc/solaris)
```

ツールの詳細については、P.14-3 の「PKCert.sh ツールの使用方法」を参照してください。

f. *KDC.cer* ファイルを KDC の証明書ディレクトリ（*BPR_HOME/kdc/solaris/packetcable/certificates*）にコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

g. 秘密鍵 KDC_private_key.pkcs8 を KDC のプラットフォーム ディレクトリ（*BPR_HOME/kdc/solaris*）にコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

h. 秘密鍵 KDC_private_key_proprietary. を KDC のプラットフォーム ディレクトリ（*BPR_HOME/kdc/solaris*）にコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

i. セカンダリ レルム（この例では、CISCO1.COM）の KDC 証明書とそれに関連付けられる秘密鍵を作成します。

次に例を示します。

```
# ./opt/CSCObac/kdc/PKCert.sh -c "-s /var/certsInput -d /var/certsOutput
-k /var/certsInput/Local_System.der -c /var/certsInput/Local_System.cer
-r CISCO1.COM -n 100 -a bactest.cisco.com -o"
Pkcert Version 1.0
Logging to pkcert.log
Source Directory: /var/certsInput
Destination Directory: /var/certsOutput
Private Key File: /var/certsInput/Local_System.der
Certificate File: /var/certsInput/Local_System.cer
Realm: CISCO.COM
Serial Number: 100
DNS Name of KDC: bactest.cisco.com
WARNING - Certificate File will be overwritten
SP Cert subject name: C=US,O=CableLabs\, Inc.,OU=ABC Cable Company,CN=Shared-01
CableLabs Local System CA
File written: /var/certsOutput/KDC_private_key.pkcs8
File written: /var/certsOutput/KDC_private_key_proprietary.
File written: /var/certsOutput/KDC_PublicKey.der
File written: /var/certsOutput/KDC.cer
KDC Certificate Successfully Created at /var/certsOutput/KDC.cer

Copy KDC.cer to the KDC certificate directory (i.e. /opt/CSCObac/kdc/solaris/
packetcable/certificates)
Copy KDC_private_key.pkcs8 to the KDC platform directory (i.e. /opt/CSCObac/
kdc/solaris)
Copy KDC_private_key_proprietary. to the KDC platform directory (i.e. /opt/CSCObac/
kdc/solaris)
```

ツールの詳細については、P.14-3 の「PKCert.sh ツールの使用方法」を参照してください。

j. *KDC.cer* をセカンダリ レルムのディレクトリにコピーします。たとえば、*BPR_HOME/kdc/solaris/packetcable/certificates* の下の */CISCO1.COM* ディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

k. 秘密鍵 KDC_private_key.pkcs8 をセカンダリ レルムのディレクトリにコピーします。たとえば、*BPR_HOME/kdc/solaris/packetcable/certificates* の下の */CISCO1.COM* ディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

l. 秘密鍵 KDC_private_key_proprietary. をセカンダリ レルムのディレクトリにコピーします。たとえば、*BPR_HOME/kdc/solaris/packetcable/certificates* の下の */CISCO1.COM* ディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

m. セカンダリ CISCO2.COM レルムの KDC 証明書とそれに関連付けられる秘密鍵を作成します。

次に例を示します。

```
# ./opt/CSCObac/kdc/PKCert.sh -c "-s /var/certsInput -d /var/certsOutput
-k /var/certsInput/Local_System.der -c /var/certsInput/Local_System.cer
-r CISCO2.COM -n 100 -a bactest.cisco.com -o"
Pkcert Version 1.0
Logging to pkcert.log
Source Directory: /var/certsInput
Destination Directory: /var/certsOutput
Private Key File: /var/certsInput/Local_System.der
Certificate File: /var/certsInput/Local_System.cer
Realm: CISCO.COM
Serial Number: 100
DNS Name of KDC: bactest.cisco.com
WARNING - Certificate File will be overwritten
SP Cert subject name: C=US,O=CableLabs\, Inc.,OU=ABC Cable Company,CN=Shared-01
CableLabs Local System CA
File written: /var/certsOutput/KDC_private_key.pkcs8
File written: /var/certsOutput/KDC_private_key_proprietary.
File written: /var/certsOutput/KDC_PublicKey.der
File written: /var/certsOutput/KDC.cer
KDC Certificate Successfully Created at /var/certsOutput/KDC.cer

Copy KDC.cer to the KDC certificate directory (i.e. /opt/CSCObac/kdc/solaris/
packetcable/certificates)
Copy KDC_private_key.pkcs8 to the KDC platform directory (i.e. /opt/CSCObac/
kdc/solaris)
Copy KDC_private_key_proprietary. to the KDC platform directory (i.e. /opt/CSCObac/
kdc/solaris)
```

ツールについては、P.14-3 の「PKCert.sh ツールの使用方法」を参照してください。

n. *KDC.cer* をセカンダリ レルムのディレクトリにコピーします。たとえば、*BPR_HOME/kdc/solaris/packetcable/certificates* の下の */CISCO2.COM* ディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

o. 秘密鍵 KDC_private_key.pkcs8 をセカンダリ レルムのディレクトリにコピーします。たとえば、*BPR_HOME/kdc/solaris/packetcable/certificates* の下の */CISCO2.COM* ディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

p. 秘密鍵 KDC_private_key_proprietary. をセカンダリ レルムのディレクトリにコピーします。たとえば、*BPR_HOME/kdc/solaris/packetcable/certificates* の下の */CISCO2.COM* ディレクトリにコピーします。ファイルをコピーする方法については、Solaris のマニュアルの **cp** コマンドを参照してください。

ステップ 6　KeyGen ツールを使用して、PacketCable サービス キーを生成します。

（注） サービス キーの生成に使用するパスワードは、**packetcable registration kdc service-key** コマンドを使用して DPE に設定したパスワードと一致するようにします。

次に例を示します。

```
# /opt/CSCObac/kdc/keygen bactest.cisco.com CISCO.COM changeme
# /opt/CSCObac/kdc/keygen bactest.cisco.com CISCO1.COM changeme
# /opt/CSCObac/kdc/keygen bactest.cisco.com CISCO2.COM changeme
```

詳細については、P.14-9 の「KeyGen ツールの使用方法」を参照してください。

ステップ 7　ステップ 6 で生成したサービス キーは *BPR_HOME/kdc/solaris/keys* ディレクトリに保存してください。

次に例を示します。

```
/opt/CSCObac/kdc/solaris/keys# ls -l
total 18
-rw-r--r--   1 root     other 2 Nov  4 09:44 krbtgt,CISCO1.COM@CISCO1.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 krbtgt,CISCO2.COM@CISCO2.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 krbtgt,CISCO.COM@CISCO.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 mtafqdnmap,bactest.cisco.com@CISCO1.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 mtafqdnmap,bactest.cisco.com@CISCO2.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 mtafqdnmap,bactest.cisco.com@CISCO.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 mtaprovsrvr,bactest.cisco.com@CISCO1.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 mtaprovsrvr,bactest.cisco.com@CISCO2.COM
-rw-r--r--   1 root     other 2 Nov  4 09:44 mtaprovsrvr,bactest.cisco.com@CISCO.COM
```

詳細については、Solaris のマニュアルを参照してください。

ステップ 8　各種証明書とサービス キーが *BPR_HOME/kdc* ディレクトリにあることを確認します。

次に例を示します。

```
/opt/CSCObac/kdc# ls
PKCert.sh   internal keygen lib pkcert.log  solaris bacckdc.license

/opt/CSCObac/kdc# cd /internal/bin
/internal/bin# ls
kdc runKDC.sh shutdownKDC.sh

# cd /opt/CSCObac/kdc/lib
# ls
libgcc_s.so.1      libstdc++.so.5     libstlport_gcc.so

# cd /opt/CSCObac/solaris/logs
# ls
kdc.log    kdc.log.1

# cd /opt/CSCObac/solaris
# ls
logs kdc.ini packetcable KDC_private_key_proprietary.

# cd keys
# ls
krbtgt,CISCO1.COM@CISCO1.COM
krbtgt,CISCO2.COM@CISCO2.COM
krbtgt,CISCO.COM@CISCO.COM
mtafqdnmap,bactest.cisco.com@CISCO1.COM
mtafqdnmap,bactest.cisco.com@CISCO2.COM
mtafqdnmap,bactest.cisco.com@CISCO.COM
mtaprovsrvr,bactest.cisco.com@CISCO1.COM
mtaprovsrvr,bactest.cisco.com@CISCO2.COM
mtaprovsrvr,bactest.cisco.com@CISCO.COM

# cd ./solaris/packetcable/certificates
# ls
KDC.cer
Local_System.cer
CableLabs_Service_Provider_Root.cer  MTA_Root.cer
CISCO1.COM                           Service_Provider.cer
CISCO2.COM

# cd ./solaris/packetcable/certificates/CISCO1.COM
# ls
KDC.cer
KDC_private_key_proprietary.

# cd ./solaris/packetcable/certificates/CISCO2.COM:
# ls
KDC.cer
KDC_private_key_proprietary.
```

詳細については、Solaris のマニュアルを参照してください。

ステップ 9　KDC を再起動します。

次に例を示します。

```
# /etc/init.d/bprAgent restart kdc
```

詳細については、P.9-2 の「コマンドラインからの BAC プロセス ウォッチドッグの使用方法」を参照してください。

ステップ 10　複数レルム用の BAC の管理者のユーザ インターフェイスを設定します。

a.　セカンダリ レルム（この場合は CISCO1.COM）の DHCP 基準を追加します。

次に例を示します。

1. **Configuration > DHCP Criteria > Manage DHCP Criteria** で、**Add** ボタンをクリックします。
2. Add DHCP Criteria ページが表示されます。
3. DHCP Name フィールドに **cisco1** と入力します。
4. **Submit** をクリックします。
5. Manage DHCP Criteria ページに戻り、cisco1 DHCP criteria をクリックします。Modify DHCP Criteria ページが表示されます。
6. Property Name で、*/ccc/kerb/realm* を選択して、Property Value フィールドに CISCO1.COM と入力します。
7. **Add**、**Submit** の順にクリックします。

詳細については、P.13-15 の「DHCP 基準の設定」を参照してください。

b. セカンダリ レルム（この場合は CISCO2.COM）の DHCP 基準を追加します。

次に例を示します。

1. **Configuration > DHCP Criteria > Manage DHCP Criteria** で、**Add** ボタンをクリックします。
2. Add DHCP Criteria ページが表示されます。
3. DHCP Name フィールドに **cisco2** と入力します。
4. **Submit** をクリックします。
5. Manage DHCP Criteria ページに戻り、cisco2 DHCP criteria をクリックします。Modify DHCP Criteria ページが表示されます。
6. Property Name で、*/ccc/kerb/realm* を選択して、Property Value フィールドに cisco2.COM と入力します。
7. **Add**、**Submit** の順にクリックします。

詳細については、P.13-15 の「DHCP 基準の設定」を参照してください。

c. プロビジョニングする各デバイスの BAC にファイルとしてテンプレートを追加します。このステップでは、Motorola MTA 用に追加します。

次に例を示します。

1. **Configuration > Files** の順に選択します。Manage Files ページが表示されます
2. **Add** をクリックします。Add Files ページが表示されます。
3. File Type ドロップダウン リストから、CableLabs Configuration Template オプションを選択します。
4. *mot-mta.tmpl* ファイルを追加します。このファイルは Motorola MTA のプロビジョニングに使用するテンプレートです。テンプレートの構文については、例 7-2 を参照してください。
5. **Submit** をクリックします。

詳細については、P.13-18 の「ファイルの管理」を参照してください。

d. プロビジョニングする各デバイスの BAC にファイルとしてテンプレートを追加します。このステップでは、Linksys MTA 用に追加します。

次に例を示します。

1. **Configuration > Files** の順に選択します。Manage Files ページが表示されます
2. **Add** をクリックします。Add Files ページが表示されます。
3. File Type ドロップダウン リストから、CableLabs Configuration Template オプションを選択します。
4. *linksys-mta.tmpl* ファイルを追加します。このファイルは Linksys MTA のプロビジョニングに使用するテンプレートです。テンプレートの構文については、例 7-3 を参照してください。

5. **Submit** をクリックします。

詳細については、P.13-18 の「ファイルの管理」を参照してください。

e. プロビジョニングする各デバイスの BAC にファイルとしてテンプレートを追加します。このステップでは、SA MTA 用に追加します。

次に例を示します。

1. **Configuration > Files** の順に選択します。Manage Files ページが表示されます
2. **Add** をクリックします。Add Files ページが表示されます。
3. File Type ドロップダウン リストから、CableLabs Configuration Template オプションを選択します。
4. *sa-mta.tmpl* ファイルを追加します。このファイルは SA MTA のプロビジョニングに使用するテンプレートです。テンプレートの構文については、例 7-4 を参照してください。
5. **Submit** をクリックします。

詳細については、P.13-18 の「ファイルの管理」を参照してください。

f. プライマリ レルム（この場合は CISCO.COM）のサービス クラスを追加します。

次に例を示します。

1. **Configuration > Class of Service** の順に選択します。
2. **Add** をクリックします。Add Class of Service ページが表示されます。
3. CISCO.COM レルムの新しいサービス クラスの名前として mot-mta と入力します。
4. サービス クラス タイプとして PacketCableMTA を選択します。
5. Property Name ドロップダウン リストから */cos/packetCableMTA/file* を選択して、*mot-mta.tmpl* テンプレート ファイル（プライマリ CISCO.COM レルムの Motorola MTA のプロビジョニングに使用）に関連付けます。
6. **Add**、**Submit** の順にクリックします。

詳細については、P.13-2 の「サービス クラスの設定」を参照してください。

g. セカンダリ レルム（この場合は CISCO1.COM）のサービス クラスを追加します。

次に例を示します。

1. **Configuration > Class of Service** の順に選択します。
2. **Add** をクリックします。Add Class of Service ページが表示されます。
3. CISCO1.COM レルムの新しいサービス クラスの名前として linksys-mta と入力します。
4. サービス クラス タイプとして PacketCableMTA を選択します。
5. Property Name ドロップダウン リストから */cos/packetCableMTA/file* を選択して、*linksys-mta.tmpl* テンプレート ファイル（セカンダリ CISCO1.COM レルムの Linksys MTA のプロビジョニングに使用）に関連付けます。
6. **Add**、**Submit** の順にクリックします。

詳細については、P.13-2 の「サービス クラスの設定」を参照してください。

h. セカンダリ レルム（この場合は CISCO2.COM）のサービス クラスを追加します。

次に例を示します。

1. **Configuration > Class of Service** の順に選択します。
2. **Add** をクリックします。Add Class of Service ページが表示されます。
3. CISCO1.COM レルムの新しいサービス クラスの名前として sa-mta と入力します。
4. サービス クラス タイプとして PacketCableMTA を選択します。
5. Property Name ドロップダウン リストから */cos/packetCableMTA/file* を選択して、*sa-mta.tmpl* テンプレート ファイル（セカンダリ CISCO2.COM レルムの SA MTA のプロビジョニングに使用）に関連付けます。
6. **Add**、**Submit** の順にクリックします。

詳細については、P.13-2 の「サービス クラスの設定」を参照してください。

ステップ 11 デバイスをオンラインにして、プロビジョニングします。プロビジョニング プロセスを説明している次の例を参照してください。

例 1

次の例では、Motorola SBV5120 をプロビジョニングする方法を説明します。

a. デバイスのケーブル モデム部分を、**sample-bronze-docsis** サービス クラスを使用するように設定してプロビジョニングします。

b. MTA 部分をプロビジョニングする場合は、**Devices > Manage Devices** ページに進みます。プロビジョニングする PacketCable デバイスを検索して選択します。Modify Device ページが表示されます。

c. ドメイン名を設定します。この例では bacclab.cisco.com を使用します。

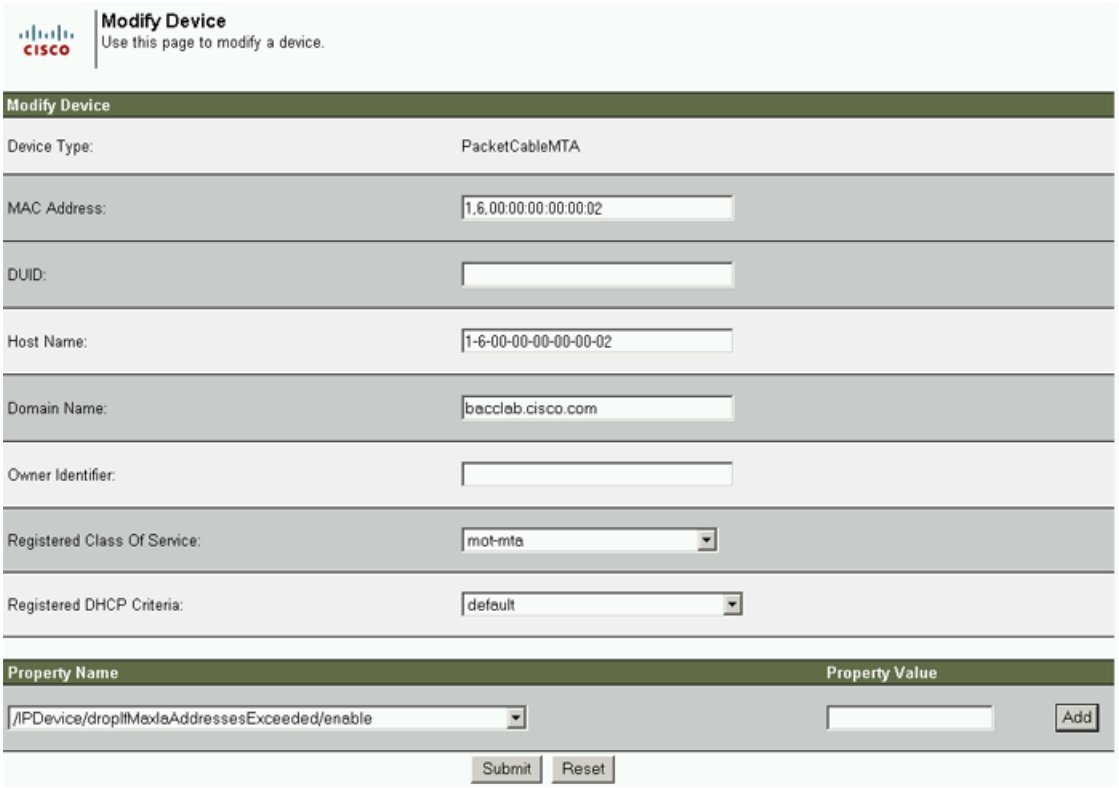
d. 登録されているサービス クラスに対応するドロップダウン リストから、**mot-mta** を選択します。これはステップ 10-f で追加したサービス クラスです。

e. 登録されている DHCP 基準に対応するドロップダウン リストから、**default** オプションを選択します。

f. **Submit** をクリックします。

図 7-2 に、Motorola MTA のデバイスの詳細を示します。

図 7-2　Motorola MTA のプロビジョニング：デバイスの詳細



例 2

次の例は、Linksys CM2P2 をプロビジョニングする方法を示しています。

a. デバイスのケーブル モデム部分を、**sample-bronze-docsis** サービス クラスを使用するように設定してプロビジョニングします。

b. MTA 部分をプロビジョニングする場合は、**Devices > Manage Devices** ページに進みます。プロビジョニングする PacketCable デバイスを検索して選択します。Modify Device ページが表示されます。

c. ドメイン名を設定します。この例では bacclab.cisco.com を使用します。

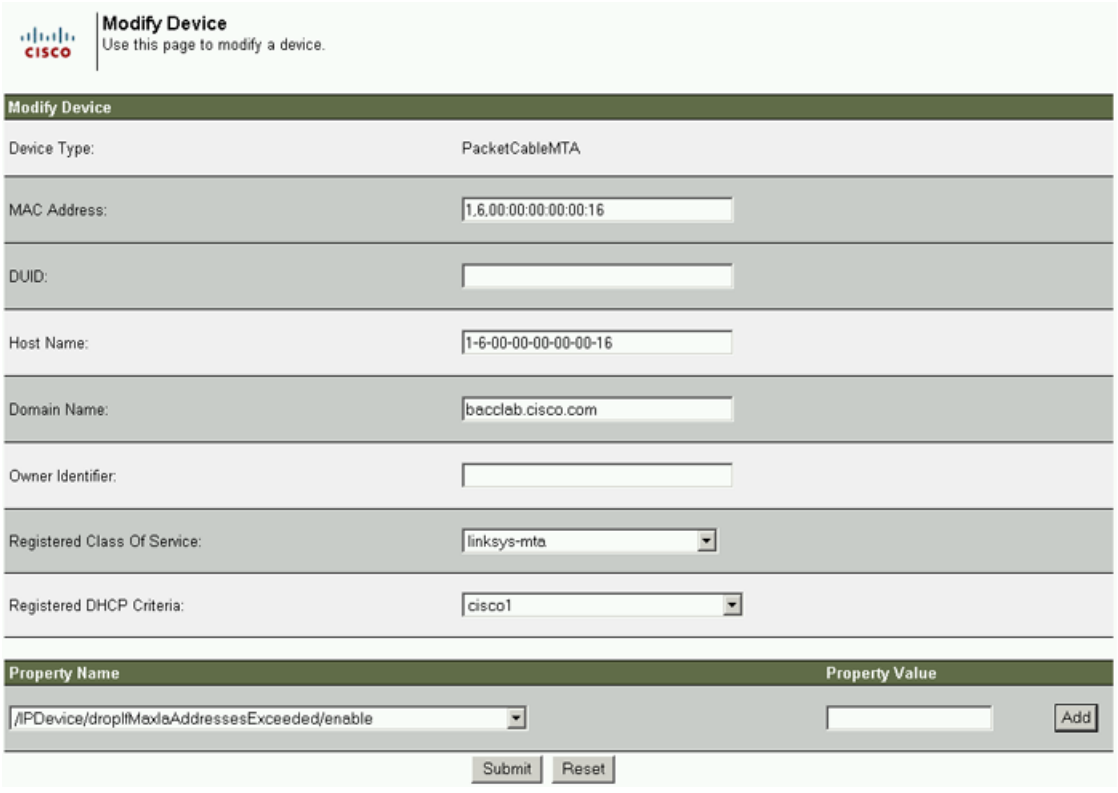
d. 登録されているサービス クラスに対応するドロップダウン リストから、**linksys-mta** を選択します。これはステップ 10-g で追加したサービス クラスです。

e. 登録されている DHCP 基準に対応するドロップダウン リストから、**cisco1** オプションを選択します。これは、ステップ 10-a でセカンダリ CISCO1.COM レルム用に追加した DHCP 基準です。

f. **Submit** をクリックします。

図 7-3 に、Linksys MTA のデバイスの詳細を示します。

図 7-3　Linksys MTA のプロビジョニング：デバイスの詳細



例 3

次の例は、SA WebStar DPX 2203 をプロビジョニングする方法を示しています。

a. デバイスのケーブル モデム部分を、**sample-bronze-docsis** サービス クラスを使用するように設定してプロビジョニングします。

b. MTA 部分をプロビジョニングする場合は、**Devices > Manage Devices** ページに進みます。プロビジョニングする PacketCable デバイスを検索して選択します。Modify Device ページが表示されます。

c. ドメイン名を設定します。この例では bacclab.cisco.com を使用します。

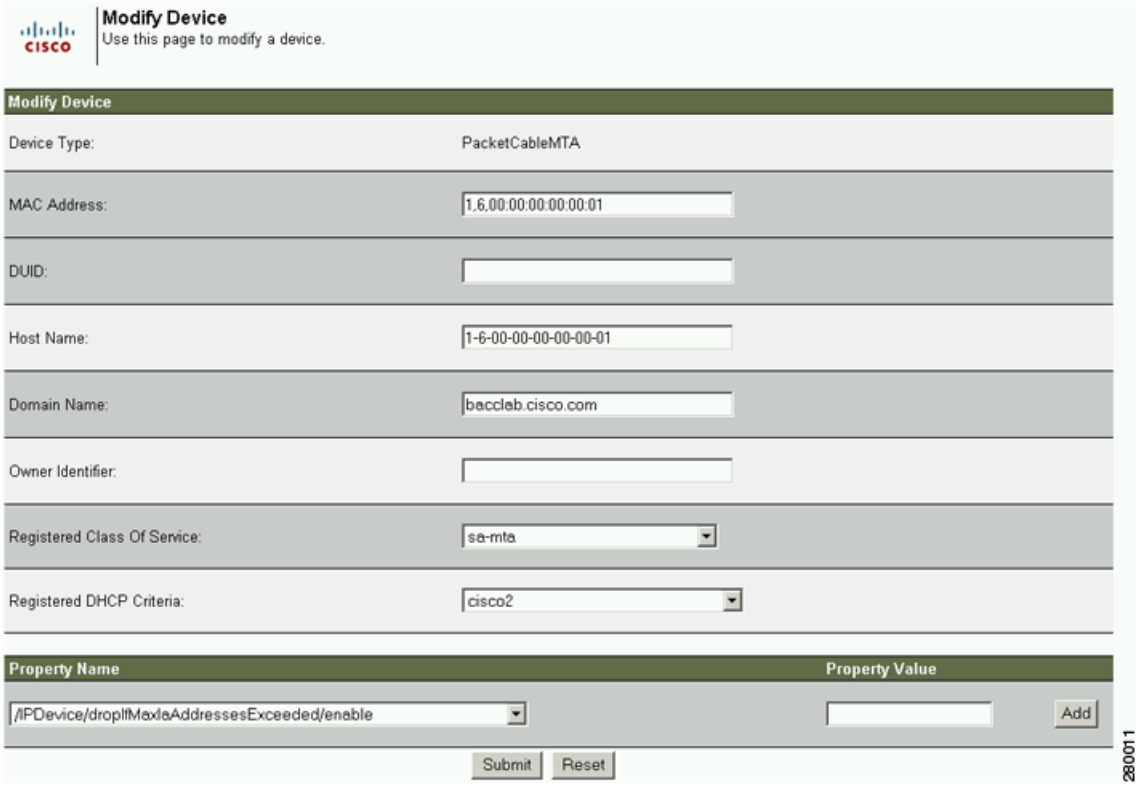
d. 登録されているサービス クラスに対応するドロップダウン リストから、**sa-mta** を選択します。これはステップ 10-h で追加したサービス クラスです。

e. 登録されている DHCP 基準に対応するドロップダウン リストから、**cisco2** オプションを選択します。これは、ステップ 10-b でセカンダリ CISCO2.COM レルム用に追加した DHCP 基準です。

f. **Submit** をクリックします。

図 7-4 に、SA MTA のデバイスの詳細を示します。

図 7-4　SA MTA のプロビジョニング：デバイスの詳細



ステップ 12　Ethereal トレースを使用して、複数レルムのサポートが動作可能かどうかを確認します。この手順で使用した設定例から表示された、KDC と DPE のログ ファイルの出力例を参照してください。

例 1

次の例は、プライマリ CISCO.COM レルムでプロビジョニングした Motorola SBV 5120 MTA の KDC および DPE ログ ファイルからの抜粋です。

KDC ログの出力例：Motorola MTA

```
INFO [Thread-4] 2007-02-07 07:56:21,133 (DHHelper.java:114) - Time to create DH key
pair(ms): 48
 INFO [Thread-4] 2007-02-07 07:56:21,229 (DHHelper.java:114) - Time to create DH key
pair(ms): 49
 INFO [Thread-4] 2007-02-07 07:56:21,287 (DHHelper.java:150) - Time to create shared
secret: 57 ms.
 INFO [Thread-4] 2007-02-07 07:56:21,289 (PKAsReqMsg.java:104) - ##MTA-9a Unconfirmed
AS Request: 1133717956 Received from /10.10.1.2
 INFO [Thread-4] 2007-02-07 07:56:21,298 (KRBProperties.java:612) - Replacing
property: 'minimum ps backoff' Old Value:'150' New Value: '150'
 INFO [Thread-4] 2007-02-07 07:56:21,324 (KDCMessageHandler.java:257) - AS-REQ
contains PKINIT - QA Tag.
 INFO [Thread-4] 2007-02-07 07:56:21,325 (KDCMessageHandler.java:279) - PK Request
from MTA received. Client is MTA - QA Tag
 INFO [Thread-4] 2007-02-07 07:56:21,365 (KDCMessageHandler.java:208) - ##MTA-9b KDC
Reply   AS-REP Sent to /10.10.1.2:1039 Time(ms): 290
 WARN [main] 2005-11-07 07:56:23,193 (KDC.java:113) - Statistics Report   ASREP's: 1
 INFO [main] 2005-11-07 07:56:23,195 (KDC.java:121) - /pktcbl/mtaAsRepSent: 10
 INFO [main] 2005-11-07 07:56:23,195 (KDC.java:121) - /pktcbl/DHKeygenTotalTime: 1043
 INFO [main] 2005-11-07 07:56:23,196 (KDC.java:121) - /pktcbl/mtaAsReqRecvd: 10
 INFO [main] 2005-11-07 07:56:23,197 (KDC.java:121) - /pktcbl/DHKeygenNumOps: 20
 INFO [main] 2005-11-07 07:56:23,197 (KDC.java:121) - /pktcbl/total: 60
```

DPE ログの出力例：Motorola MTA

```
dpe.cisco.com: 2007 02 07 07:56:24 EST: %BAC-DPE-6-4178: Adding Replay Packet:  []
dpe.cisco.com: 2007 02 07 07:56:24 EST: %BAC-PKTSNMP-6-0764: [System Description for
MTA: <<HW_REV: 1.0, VENDOR: Motorola Corporation, BOOTR: 8.1, SW_REV:
SBV5120-2.9.0.1-SCM21-SHPC, MODEL: SBV5120>>]
dpe.cisco.com: 2007 02 07 07:56:24 EST: %BAC-PKTSNMP-6-0764: [##MTA-15 SNMPv3 INFORM
Received From 10.10.1.2.]
dpe.cisco.com: 2007 02 07 07:56:24 EST: %BAC-DPE-6-0688: Received key material update
for device [1,6,01:11:82:61:5e:30]
dpe.cisco.com: 2007 02 07 07:56:24 EST: %BAC-PKTSNMP-6-0764: [##MTA-19 SNMPv3 SET Sent
to 10.10.1.2]
dpe.cisco.com: 2007 02 07 07:56:24 EST: %BAC-TFTP-6-0310: Finished handling [read]
request from [10.10.1.2:1190] for [bpr0106001182615e300001]
dpe.cisco.com: 2007 02 07 07:56:25 EST: %BAC-PKTSNMP-6-0764: [##MTA-25 SNMP
Provisioning State INFORM Received from 10.10.1.2.  Value: 1]
```

例 2

次の例は、セカンダリ CISCO1.COM レルムでプロビジョニングした Linksys CM2P2 MTA の KDC および DPE ログ ファイルからの抜粋です。

KDC ログの出力例：Linksys MTA

```
 INFO [Thread-8] 2007-02-07 08:00:10,664 (DHHelper.java:114) - Time to create DH key
pair(ms): 49
 INFO [Thread-8] 2007-02-07 08:00:10,759 (DHHelper.java:114) - Time to create DH key
pair(ms): 49
 INFO [Thread-8] 2007-02-07 08:00:10,817 (DHHelper.java:150) - Time to create shared
secret: 57 ms.
 INFO [Thread-8] 2007-02-07 08:00:10,819 (PKAsReqMsg.java:104) - ##MTA-9a Unconfirmed
AS Request: 1391094112 Received from /10.10.1.5
 INFO [Thread-8] 2007-02-07 08:00:10,828 (KRBProperties.java:612) - Replacing
property: 'minimum ps backoff' Old Value:'150' New Value: '150'
 INFO [Thread-8] 2007-02-07 08:00:10,860 (KDCMessageHandler.java:257) - AS-REQ
contains PKINIT - QA Tag.
 INFO [Thread-8] 2007-02-07 08:00:10,862 (KDCMessageHandler.java:279) - PK Request
from MTA received. Client is MTA - QA Tag
 INFO [Thread-8] 2007-02-07 08:00:10,901 (KDCMessageHandler.java:208) - ##MTA-9b KDC
Reply   AS-REP Sent to /10.10.1.5:3679 Time(ms): 296
 WARN [main] 2007-02-07 08:00:13,383 (KDC.java:113) - Statistics Report   ASREP's: 1
 INFO [main] 2007-02-07 08:00:13,384 (KDC.java:121) - /pktcbl/mtaAsRepSent: 11
 INFO [main] 2007-02-07 08:00:13,384 (KDC.java:121) - /pktcbl/DHKeygenTotalTime: 1141
```

DPE ログの出力例：Linksys MTA

```
dpe.cisco.com: 2007 02 07 08:00:10 EST: %BAC-DPE-6-4112: Adding Replay Packet:  []
dpe.cisco.com: 2007 02 07 08:00:12 EST: %BAC-DPE-6-4178: Adding Replay Packet:  []
dpe.cisco.com: 2007 02 07 08:00:12 EST: %BAC-PKTSNMP-6-0764: [System Description for
MTA: Linksys Cable Modem with 2 Phone Ports (CM2P2) <<HW_REV: 2.0, VENDOR: Linksys,
BOOTR: 2.1.6V, SW_REV: 2.0.3.3.11-1102, MODEL: CM2P2>>]
dpe.cisco.com: 2007 02 07 08:00:12 EST: %BAC-PKTSNMP-6-0764: [##MTA-15 SNMPv3 INFORM
Received From 10.10.1.5.]
dpe.cisco.com: 2007 02 07 08:00:12 EST: %BAC-DPE-6-0688: Received key material update
for device [1,6,00:0f:68:f9:42:f6]
dpe.cisco.com: 2007 02 07 08:00:12 EST: %BAC-PKTSNMP-6-0764: [##MTA-19 SNMPv3 SET Sent
to 10.10.1.5]
dpe.cisco.com: 2007 02 07 08:00:18 EST: %BAC-TFTP-6-0310: Finished handling [read]
request from [10.10.1.5:1032] for [bpr0106000f68f942f60001]
dpe.cisco.com: 2007 02 07 08:00:18 EST: %BAC-PKTSNMP-6-0764: [##MTA-25 SNMP
Provisioning State INFORM Received from 10.10.1.5.  Value: 1]
```

例 3

次の例は、セカンダリ CISCO2.COM レルムでプロビジョニングした SA WebStar DPX 2203 MTA の KDC および DPE ログ ファイルからの抜粋です。

KDC ログの出力例：SA MTA

```
 INFO [Thread-6] 2007-02-07 08:01:31,556 (DHHelper.java:114) - Time to create DH key
pair(ms): 49
 INFO [Thread-6] 2007-02-07 08:01:31,652 (DHHelper.java:114) - Time to create DH key
pair(ms): 50
 INFO [Thread-6] 2007-02-07 08:01:31,711 (DHHelper.java:150) - Time to create shared
secret: 57 ms.
 INFO [Thread-6] 2007-02-07 08:01:31,715 (PKAsReqMsg.java:104) - ##MTA-9a Unconfirmed
AS Request: 575634000 Received from /10.10.1.50
 INFO [Thread-6] 2007-02-07 08:01:31,727 (KRBProperties.java:612) - Replacing
property: 'minimum ps backoff' Old Value:'150' New Value: '150'
 INFO [Thread-6] 2007-02-07 08:01:31,752 (KDCMessageHandler.java:257) - AS-REQ
contains PKINIT - QA Tag.
 INFO [Thread-6] 2007-02-07 08:01:31,753 (KDCMessageHandler.java:279) - PK Request
from MTA received. Client is MTA - QA Tag
 INFO [Thread-6] 2007-02-07 08:01:31,792 (KDCMessageHandler.java:208) - ##MTA-9b KDC
Reply   AS-REP Sent to /10.10.1.50:3679 Time(ms): 292
 WARN [main] 2007-02-07 08:01:33,423 (KDC.java:113) - Statistics Report   ASREP's: 1
 INFO [main] 2007-02-07 08:01:33,424 (KDC.java:121) - /pktcbl/mtaAsRepSent: 12
 INFO [main] 2007-02-07 08:01:33,425 (KDC.java:121) - /pktcbl/DHKeygenTotalTime: 1240
 INFO [main] 2007-02-07 08:01:33,425 (KDC.java:121) - /pktcbl/mtaAsReqRecvd: 12
 INFO [main] 2007-02-07 08:01:33,426 (KDC.java:121) - /pktcbl/DHKeygenNumOps: 24
 INFO [main] 2007-02-07 08:01:33,426 (KDC.java:121) - /pktcbl/total: 72
```

DPE ログの出力例：SA MTA

```
dpe.cisco.com: 2007 02 07 08:01:31 EST: %BAC-DPE-6-4112: Adding Replay Packet:  []
dpe.cisco.com: 2007 02 07 08:01:33 EST: %BAC-DPE-6-4178: Adding Replay Packet:  []
dpe.cisco.com: 2007 02 07 08:01:33 EST: %BAC-PKTSNMP-6-0764: [System Description for
MTA: S-A WebSTAR DPX2200 Series DOCSIS E-MTA Ethernet+USB (2)Lines VOIP <<HW_REV: 2.0,
VENDOR: S-A, BOOTR: 2.1.6b, SW_REV: v1.0.1r1133-0324, MODEL: DPX2203>>]
dpe.cisco.com: 2007 02 07 08:01:33 EST: %BAC-PKTSNMP-6-0764: [##MTA-15 SNMPv3 INFORM
Received From 10.10.1.50.]
dpe.cisco.com: 2007 02 07 08:01:33 EST: %BAC-DPE-6-0688: Received key material update
for device [1,6,00:0f:24:d8:6e:f5]
dpe.cisco.com: 2007 02 07 08:01:33 EST: %BAC-PKTSNMP-6-0764: [##MTA-19 SNMPv3 SET Sent
to 10.10.1.50]
dpe.cisco.com: 2007 02 07 08:01:38 EST: %BAC-TFTP-6-0310: Finished handling [read]
request from [10.10.1.50:1037] for [bpr0106000f24d86ef50001]
dpe.cisco.com: 2007 02 07 08:01:39 EST: %BAC-PKTSNMP-6-0764: [##MTA-25 SNMP
Provisioning State INFORM Received from 10.10.1.50.  Value: 1]
```

複数レルムのデバイスのプロビジョニングに使用するテンプレートのオーサリング

ここで説明するテンプレートの構文を使用して、特定レルムのデバイスをプロビジョニングできます。ここで示す例は、Motorola SBV5120 MTA（例 7-2）、Linksys CM2P2 MTA（例 7-3）、および SA WebStar DPX2203 MTA（例 7-4）に固有のものです。

（注）　これらのテンプレートを修正して、自分のネットワーク内の MTA の仕様に合せる必要があります。

例 7-2　Motorola MTA のプロビジョニングに使用するテンプレート

```
#
# Example PacketCable MTA template: mot-mta.tmpl
#
# Note that this template is specific to the TI 401 MTA.
# This template must be modified to the specifics of your MTA.
#
# First, the start marker.
#
option 254 1
#
# Enable MTA
#
option 11 .pktcMtaDevEnabled.0,INTEGER,true
#
# Set CMS FQDN for each endpoint on the MTA.
# NOTE: the indexes (9 and 10 here) will differ per manufacturer.
#
option 11
.pktcNcsEndPntConfigTable.pktcNcsEndPntConfigEntry.pktcNcsEndPntConfigCallAgentId.9,ST
RING,CMS.CISCO.COM
option 11
.pktcNcsEndPntConfigTable.pktcNcsEndPntConfigEntry.pktcNcsEndPntConfigCallAgentId.10,S
TRING,CMS.CISCO.COM
#
# Set the realm org name.  This MUST match that contained in the cert chain used by
the device.
#
# "CableLabs, Inc."
option 11
.pktcMtaDevRealmTable.pktcMtaDevRealmEntry.pktcMtaDevRealmOrgName.'CISCO.COM',STRING,"
'43:61:62:6C:65:4C:61:62:73:2C:20:49:6E:63:2E'"
#
# Set the realm name and IPSec control for the CMS.
#
option 11
.pktcMtaDevCmsTable.pktcMtaDevCmsEntry.pktcMtaDevCmsIpsecCtrl.'CMS.CISCO.COM',INTEGER,
true
option 11
pktcMtaDevCmsTable.pktcMtaDevCmsEntry.pktcMtaDevCmsKerbRealmName.'CMS.CISCO.COM',STRIN
G,CISCO.COM
#
# Finally, the end marker.
#
option 254 255
```

例 7-3　Linksys MTA のプロビジョニングに使用するテンプレート

このテンプレートでは、レルムが CISCO1.COM に設定されている点に注意してください。

```
#
# Example PacketCable MTA template: linksys-mta.tmpl
#
# Note that this template is specific to the TI 401 MTA.
# This template must be modified to the specifics of your MTA.
#
# First, the start marker.
#
option 254 1
#
# Enable MTA
#
option 11 .pktcMtaDevEnabled.0,INTEGER,true
#
# Set CMS FQDN for each endpoint on the MTA.
# NOTE: the indexes (9 and 10 here) will differ per manufacturer.
#
option 11
.pktcNcsEndPntConfigTable.pktcNcsEndPntConfigEntry.pktcNcsEndPntConfigCallAgentId.9,ST
RING,CMS.CISCO.COM
option 11
.pktcNcsEndPntConfigTable.pktcNcsEndPntConfigEntry.pktcNcsEndPntConfigCallAgentId.10,S
TRING,CMS.CISCO.COM
#
# Set the realm org name.  This MUST match that contained in the cert chain used by
the device.
#
# "CableLabs, Inc."
option 11
.pktcMtaDevRealmTable.pktcMtaDevRealmEntry.pktcMtaDevRealmOrgName.'CISCO1.COM',STRING,
"'43:61:62:6C:65:4C:61:62:73:2C:20:49:6E:63:2E'"
#
# Set the realm name and IPSec control for the CMS.
#
option 11
.pktcMtaDevCmsTable.pktcMtaDevCmsEntry.pktcMtaDevCmsIpsecCtrl.'CMS.CISCO.COM',INTEGER,
true
option 11
pktcMtaDevCmsTable.pktcMtaDevCmsEntry.pktcMtaDevCmsKerbRealmName.'CMS.CISCO.COM',STRIN
G,CISCO1.COM
#
# Finally, the end marker.
#
option 254 255
```

例 7-4　SA MTA のプロビジョニングに使用するテンプレート

このテンプレートでは、レルムが CISCO2.COM に設定されている点に注意してください。

```
#
# Example PacketCable MTA template: sa-mta.tmpl
#
# Note that this template is specific to the TI 401 MTA.
# This template must be modified to the specifics of your MTA.
#
# First, the start marker.
#
option 254 1
#
# Enable MTA
#
option 11 .pktcMtaDevEnabled.0,INTEGER,true
#
# Set CMS FQDN for each endpoint on the MTA.
# NOTE: the indexes (9 and 10 here) will differ per manufacturer.
#
option 11
.pktcNcsEndPntConfigTable.pktcNcsEndPntConfigEntry.pktcNcsEndPntConfigCallAgentId.9,ST
RING,CMS.CISCO.COM
option 11
.pktcNcsEndPntConfigTable.pktcNcsEndPntConfigEntry.pktcNcsEndPntConfigCallAgentId.10,S
TRING,CMS.CISCO.COM
#
# Set the realm org name.  This MUST match that contained in the cert chain used by
the device.
#
# "CableLabs, Inc."
option 11
.pktcMtaDevRealmTable.pktcMtaDevRealmEntry.pktcMtaDevRealmOrgName.'CISCO2.COM',STRING,
"'43:61:62:6C:65:4C:61:62:73:2C:20:49:6E:63:2E'"
#
# Set the realm name and IPSec control for the CMS.
#
option 11
.pktcMtaDevCmsTable.pktcMtaDevCmsEntry.pktcMtaDevCmsIpsecCtrl.'CMS.CISCO.COM',INTEGER,
true
option 11
pktcMtaDevCmsTable.pktcMtaDevCmsEntry.pktcMtaDevCmsKerbRealmName.'CMS.CISCO.COM',STRIN
G,CISCO2.COM
#
# Finally, the end marker.
#
option 254 255
```

Network Registrar DNS サーバでの SRV レコードの設定

Network Registrar DNS サーバが KDC とともに稼動するように設定する必要があります。この設定を行う場合は、Network Registrar のマニュアルとその説明を参照してください。

（注）　目的のレルム名と一致するゾーン名を作成し、この特殊なゾーン内の DNS レコード（ゾーンを維持するために DNS サーバが必要とするレコード以外）のみをそのレルムの SRV レコードにすることをお勧めします。この例では、目的の Kerberos レルムが voice.example.com で、その他のすべての KDC、Network Registrar、および DPE の設定がすでに行われていることを想定しています。KDC の FQDN は、kdc.example.com であると仮定します。

ステップ 1　**nrcmd** コマンドライン ツールを起動します（デフォルトでは、*/opt/nwreg2/local/usrbin* ディレクトリにあります）。

ステップ 2　ユーザ名とパスワードを入力します。

ステップ 3　Kerberos レルムのゾーンを作成するには、次のように入力します。

```
nrcmd> zone voice.example.com create primary address_of_nameserver hostmaster
```

ここで、*address_of_nameserver* はネーム サーバの IP アドレスを指定します。

ステップ 4　SRV レコードを新しいゾーンに追加するには、次のように入力します。

```
nrcmd> zone voice.example.com. addRR _kerberos._udp. srv 0 0 88 KDC_FQDN
```

ここで、*KDC_FQDN* は KDC の FQDN を指定します。

ステップ 5　保存して、DNS サーバをリロードするには、次のように入力します。

```
nrcmd> save
nrcmd> dns reload
```

PacketCable MTA と安全に通信するための RDU と DPE 上での SNMPv3 クローニングの設定

BAC を使用すると、SNMPv3 の外部ネットワーク マネージャから MTA デバイスへのアクセスをイネーブルにできます。また、RDU は特定の MTA で SNMPv3 処理を実行できます。

この機能をイネーブルにするには、DPE と RDU にセキュリティ鍵関連情報を設定します。鍵関連情報が設定されると、クローニングされた SNMPv3 エントリの作成に使用する BAC Application Programming Interface（API; アプリケーション プログラミング インターフェイス）コールがイネーブルになります。

（注）　この機能をイネーブルにすると、プロビジョニングのパフォーマンスに影響を与えます。

鍵関連情報の作成と鍵の生成

鍵関連情報を作成するには、次の 2 つのステップの処理を実行します。

1. RDU でスクリプト コマンドを実行します。
2. DPE で CLI コマンドを実行します。

（注）　この共有秘密情報は、CMTS または BAC の共有秘密情報と同じものではありません。

鍵関連情報を作成するには、次の手順に従います。

ステップ 1　*BPR_HOME/rdu/bin* ディレクトリから、次のスクリプトを RDU 上で実行します。

```
# generateSharedSecret.sh password
```

password は、ユーザが作成する 6 ～ 20 文字の任意のパスワードです。このパスワードは、46 バイトの鍵の生成に使用されます。この鍵は、*keymaterial.txt* という名前のファイルに保存されます。このファイルは *BPR_HOME/rdu/conf* ディレクトリにあります。

ステップ 2　ステップ 1 でこの鍵の生成に使用した *password* を使用して、この音声技術がイネーブルになっているすべての DPE 上で **service packetcable** *1..1* **snmp key-material** DPE CLI コマンドを実行します。このコマンドは、DPE 上に同じ 46 バイトの鍵を生成し、RDU と DPE を同期して、MTA と安全に通信できるようにします。

PacketCable eMTA の Basic プロビジョニング

BAC は簡潔で DOCSIS に似た、ノンセキュアなプロビジョニング フローを提供する PacketCable Basic もサポートしています。表 7-5 で、図 7-1 のプロビジョニング ワークフローを使用する BASIC.1 のフローを説明します。

表 7-5　PacketCable eMTA の Basic プロビジョニング

手順	ワークフロー	説明
MTA-1	DHCP ブロードキャスト検出	セキュアなフローと同様に実行されます。
MTA-2	DHCP オファー	プロビジョニング システムが BASIC.1 モードで MTA をプロビジョニングするように設定されている場合、プロビジョニング システムは、Option 122 のサブオプション 6（特別な予約済みレルム名「BASIC.1」を含む）を含んでいる DHCP オファーを返します。この予約済みレルム名は、BASIC.1 プロビジョニング フローを使用するように MTA に指示します。このオファーには、Option 122.3 のプロビジョニング システムの IP アドレスも含まれており、file フィールドと siaddr フィールドには MTA 設定ファイルがある場所が入力されています。
MTA-3	DHCP 要求	MTA DHCP 交換の残りが実行されます（要求と確認の交換）。
MTA-4	DHCP 確認	
MTA-22	テレフォニー設定ファイル要求	MTA は、ステップ MTA-22 まで直接スキップします。file と siaddr の情報を使用して、MTA はその設定ファイルを TFTP 経由でプロビジョニング システムからコピーします。BAC は、TFTP サーバを DPE コンポーネントに統合します。 **（注）** MTA、プロビジョニング サーバの認証、または暗号化は一切発生しません。
MTA-23	テレフォニー設定ファイル	

BASIC.2 フローは、次の例外を除いて BASIC.1 と同一です。

- 「BASIC.2」が MTA の DHCP Option 122 のサブオプション 6 に入力される。
- フローの最後の MTA-25 で、MTA がプロビジョニング ステータス SNMPv2c INFORM を発行する（DHCP Option 122 のサブオプション 3 が通知対象を指定）。

PacketCable Basic フローは DOCSIS フローと似ていますが、次の点が異なります。

- MTA とプロビジョニング システム間で ToD 交換が行われない。
- MTA 設定ファイルに完全性ハッシュが含まれる。具体的に、設定ファイルの内容全体の SHA1 ハッシュが pktcMtadevConfigFileHash SNMP VarBind に入力され、EoF TLV ファイル直前の TLV 11 に配置される。
- MTA がその設定ファイルを受信して処理すると、BASIC.2 フローがプロビジョニング ステータス SNMPv2c 通知を発行する。この通知は、MTA のプロビジョニングが正常に完了したかどうかを BAC に通知します。問題が発生した場合は、エラーが生成され、イベントが DPE から RDU に、そして BAC クライアントに送信されます。この通知は、設定ファイルの問題をデバッグする場合に役立ちます。

DOCSIS フローの詳細については、第 6 章「DOCSIS 設定」を参照してください。

（注）　PacketCable Basic のプロビジョニング フローを使用する前に、PacketCable Basic に対応した eMTA を使用していることを確認してください。eMTA は、DHCP 検出の Option 60、TLV 5.18（サポートされているフロー）で Basic 対応であることを報告する必要があります。

PacketCable TLV 38 および MIB のサポート

BAC は一連の PacketCable 1.5 MIB を完全にサポートしています。

BAC は PacketCable 設定テンプレートで TLV 38 をサポートしています。この TLV を使用して、複数の SNMP 通知ターゲットを設定できます。この TLV を設定すると、TLV 38 で設定したターゲットにもすべての通知が発行されることになります。

SNMP v2C の通知

BAC は PacketCable MTA からの SNMP v2C TRAP と INFORM 通知の両方をサポートします。

Euro PacketCable

Euro-PacketCable サービスは、基本的に北米版 PacketCable サービスの欧州版に相当しますが、次の点が異なります。

- Euro PacketCable では、使用する MIB が異なる。
- Euro PacketCable では、使用するデバイス証明書（*MTA_Root.cer*）とサービス プロバイダー証明書（サービス プロバイダーのルート）のセットが異なる。

  Euro-PacketCable 証明書の場合、*kdc.ini* ファイルの *euro-packetcable* プロパティを true に設定する必要があります。KDC は、Euro-PacketCable（tComLabs）証明書チェーンをサポートしています。次は、Euro PacketCable がイネーブルになっている KDC 設定ファイルの例です。

```
[general]
interface address = 10.10.10.1
FQDN = servername.cisco.com
maximum log file size = 10000
n saved log files = 100
log debug level = 5 minimum
ps backoff = 150 maximum
ps backoff = 300
euro-packetcable = true
```

  Euro PacketCable を使用する場合は、PacketCable のプロパティ */pktcbl/prov/locale* の値を必ず EURO に設定してください。デフォルトは NA（North America）です。ロケールは、設定ファイル ユーティリティで指定できます。詳細については、P.5-23 の「設定ファイル ユーティリティの使用方法」を参照してください。

Euro-PacketCable MIB

Euro-PacketCable MIB は、基本的にはドラフト IETF MIB のスナップショットです。MTA 設定ファイルは、MIB を参照する SNMP VarBinds で構成されます。北米版 PacketCable と Euro-PacketCable の MIB は大きく異なるため、北米版 PacketCable と Euro-PacketCable の設定ファイルには互換性はありません。インストール時に、北米版 PacketCable のサンプル ファイル（*cw29_config.tmpl*）と Euro PacketCable のサンプル ファイル（*ecw15_mta_config.tmpl*）が *BPR_HOME/rdu/samples* ディレクトリにコピーされます。

BAC には次の Euro-PacketCable MIB が付属しています。

- DOCS-IETF-BPI2-MIB
- INTEGRATED-SERVICES-MIB
- DIFFSERV-DSCP-TC
- DIFFSERV-MIB
- TCOMLABS-MIB
- PKTC-TCOMLABS-MTA-MIB
- PKTC-TCOMLABS-SIG-MIB

Euro-PacketCable MIB の設定

Euro-PacketCable MIB を使用するように BAC を設定するには、ロードする MIB を指定する BAC RDU プロパティを変更する必要があります。デフォルトでは、このプロパティには PacketCable MIB が含まれます。

次のいずれかの方法でプロパティを変更できます。

- *rdu.properties* を修正して、RDU を再起動する。
- 管理者のユーザ インターフェイスで、**Configuration > Defaults > System Defaults** に移動して、MIB リストを下に示すリストで置き換えます。RDU を再起動する必要はありません。
- Prov API *changeSystemDefaults()* コールを使用する。RDU を再起動する必要はありません。

プロパティ名は */snmp/mibs/mibList*（properties ファイル）または `SNMPPropertyKeys.MIB_LIST`（Prov API 定数名）です。プロパティの値は、Comma-Separated Value（CSV; カンマ区切り形式）で、次に示す必須の MIB 名で構成されます。

```
/snmp/mibs/mibList=SNMPv2-SMI,SNMPv2-TC,INET-ADDRESS-MIB,CISCO-SMI,CISCO-TC,SNMPv2-MIB
,RFC1213-MIB,IANAifType-MIB,IF-MIB,DOCS-IF-MIB,DOCS-IF-EXT-MIB,DOCS-BPI-MIB,CISCO-CABL
E-SPECTRUM-MIB,CISCO-DOCS-EXT-MIB,SNMP-FRAMEWORK-MIB,DOCS-CABLE-DEVICE-MIB,DOCS-CABLE-
DEVICE-MIB-OBSOLETE,DOCS-QOS-MIB,CISCO-CABLE-MODEM-MIB,DOCS-IETF-BPI2-MIB,INTEGRATED-S
ERVICES-MIB,DIFFSERV-DSCP-TC,DIFFSERV-MIB,TCOMLABS-MIB,PKTC-TCOMLABS-MTA-MIB,PKTC-TCOM
LABS-SIG-MIB
```