CHAPTER 54

Cisco Mobility Advantage の設定

この章では、Cisco Unified Communications Mobility Advantage Proxy 機能向けに適応型セキュリティ アプライアンスを設定する方法について説明します。

この章の内容は、次のとおりです。

Cisco Mobility Advantage Proxy 機能に関する情報

ここでは、次の内容について説明します。

Cisco Mobility Advantage Proxy 機能

Cisco Mobility Advantage ソリューション向けの Cisco UMA をサポートするため、Mobility Advantage Proxy（TLS プロキシとして実装）には次の機能が含まれます。

- クライアントとのハンドシェイク中にクライアントの認証を許可しない機能
- サーバにインポートされた PKCS-12 証明書をプロキシの証明書として許可する機能

ASA には、Cisco UMA Mobile Multiplexing Protocol（MMP; モバイル多重化プロトコル）を検証するためのインスペクション エンジンが含まれます。

MMP は、Cisco UMA クライアントとサーバとの間でデータ エントリを送信するための転送プロトコルです。MMP はコネクション型プロトコルの最上部（基盤となる転送）で実行する必要があり、TLS などの安全な転送プロトコルの最上部で実行することを目的としています。Orative Markup Language（OML）プロトコルは、データの同期を目的とした MMP に加えて、大規模なファイルのアップロードとダウンロードのための HTTP プロトコルの上で実行することを意図しています。

TCP/TLS のデフォルト ポートは 5443 です。埋め込まれた NAT やセカンダリ接続はありません。

Cisco UMA クライアントおよびサーバ通信は TLS を通じてプロキシ処理ができます。ここで、データを復号化してインスペクション MMP モジュールに渡し、エンドポイントに転送する前にデータを再暗号化します。インスペクション MMP モジュールは MMP ヘッダーの整合性を確認し、OML/HTTP を適切なハンドラに渡します。ASA は、MMP ヘッダーおよびデータで次のアクションを実行します。

- クライアント MMP ヘッダーの形式が適切であることを確認します。間違った形式のヘッダーを検出すると、TCP セッションは終了します。
- クライアントからサーバへの MMP ヘッダーの長さを超えていないことを確認します。MMP ヘッダーの長さを超えている場合は（4096）、TCP セッションは終了します。
- クライアントからサーバへの MMP コンテンツの長さを超えていないことを確認します。エンティティのコンテンツの長さを超えている場合は（4096）、TCP セッションは終了します。

（注）　4096 は、MMP の実装で現在使用されている値です。

MMP ヘッダーとエンティティはパケット間で分割できるため、ASA はデータをバッファリングして、インスペクションの一貫性を確保します。Stream API（SAPI; ストリーム API）は、保留中のインスペクションを実行できるようにデータのバッファリングを処理します。MMP ヘッダー テキストは大文字と小文字を区別しないものとして処理されます。ヘッダー テキストと値の間にスペースが入ります。MMP の状態の再要求は、TCP 接続の状態をモニタすることによって実行されます。

Mobility Advantage Proxy の導入シナリオ

図 54-1 と図 54-2 に、Cisco Mobility Advantage ソリューションによって使用される TLS プロキシの 2 つの導入シナリオを示します。シナリオ 1（推奨される導入アーキテクチャ）では、ASA はファイアウォールと TLS プロキシの両方として機能します。シナリオ 2 では、ASA は TLS プロキシとしてだけ機能し、既存のファイアウォールを使用します。いずれのシナリオでも、クライアントはインターネットから接続します。

シナリオ 1 の導入では、ASA は Cisco UMA クライアントと Cisco UMA サーバの間にあります。Cisco UMA クライアントは、個々のスマートフォンにダウンロードされる実行可能ファイルです。Cisco UMA クライアント アプリケーションは、企業の Cisco UMA サーバに対するデータ接続（TLS 接続）を確立します。ASA は通信を代行受信し、クライアントが Cisco UMA サーバに送信するデータを検査します。

（注）　Cisco Mobility Advantage ソリューションの TLS プロキシは、Cisco UMA クライアントが証明書を提供できないため、クライアント認証をサポートしません。次のコマンドを使用して、TLS ハンドシェイク中の認証をディセーブルにできます。

```
hostname(config)# tls-proxy my_proxy
hostname(config-tlsp)# no server authenticate-client
```

図 54-1　Mobility Advantage Proxy と MMP インスペクションを使用したファイアウォールとして機能するセキュリティ アプライアンス

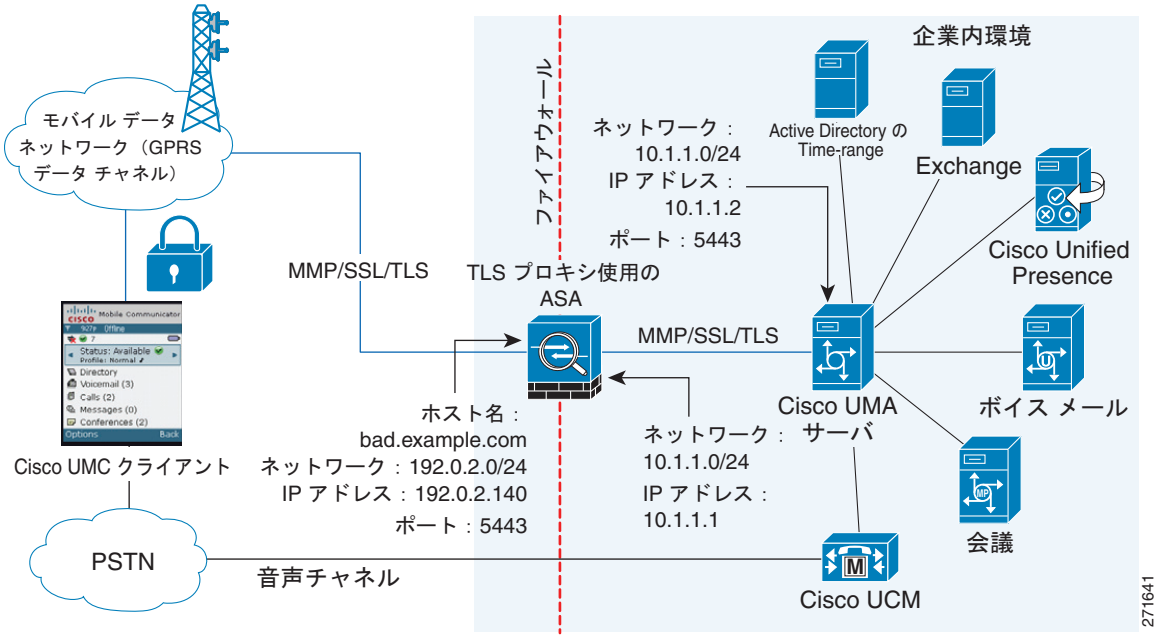


図 54-1 では、ASA は Cisco UMA サーバの 10.1.1.2 IP アドレスを 192.0.2.140 に変換することで、スタティック NAT を実行しています。

図 54-2 に導入シナリオ 2 を示します。ここでは、ASA が TLS プロキシとしてだけ機能し、企業ファイアウォールとしては機能しません。このシナリオでは、ASA と企業ファイアウォールは NAT を実行しています。企業ファイアウォールは、インターネットのどのクライアントを企業の Cisco UMA サーバに接続する必要があるのかを予測できません。したがって、この導入をサポートするために、次のアクションを実行することができます。

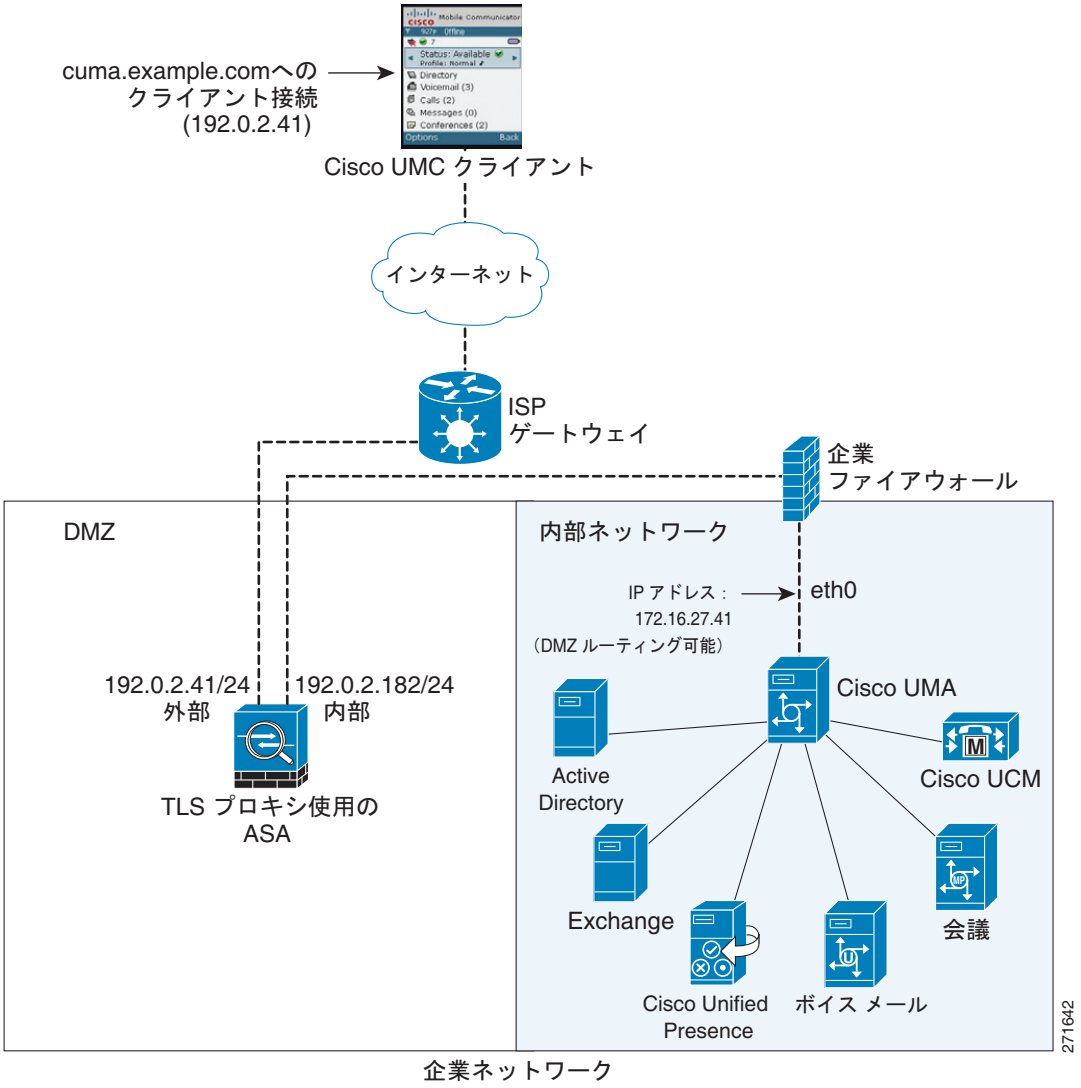
- 宛先 IP アドレス 192.0.2.41 を 172.16.27.41 に変換する着信トラフィックの NAT ルールを設定します。
- すべてのパケットの送信元 IP アドレスを変換する着信トラフィックのインターフェイス PAT ルールを設定し、企業ファイアウォールがワイルドカード ピンホールを開く必要がないようにします。Cisco UMA サーバは送信元 IP アドレスが 192.0.12.183 のパケットを受信します。

```
hostname(config)# object network obj-0.0.0.0-01
hostname(config-network-object)# subnet 0.0.0.0 0.0.0.0
hostname(config-network-object)# nat (outside,inside) dynamic 192.0.2.183
```

詳細については、第 34 章「ネットワーク オブジェクト NAT の設定」と第 35 章「Twice NAT の設定」を参照してください。

（注）　このインターフェイス PAT ルールでは、別の送信元ポートを使用して、ASA の外部インターフェイスの Cisco UMA クライアントの IP アドレスを、内部インターフェイスの 1 つの IP アドレスに収束します。このアクションは、多くの場合「外部 PAT」と呼ばれます。「外部 PAT」は、Cisco Mobility Advantage の TLS プロキシが、電話プロキシ、Cisco Unified Presence、またはアプリケーション インスペクションが必要なその他の機能を持つ ASA の同じインターフェイス上でイネーブルになっている場合には推奨しません。「外部 PAT」は、埋め込みアドレスの変換が必要な場合には、アプリケーション インスペクションによって完全にサポートされるわけではありません。

図 54-2　Cisco UMC/Cisco UMA のアーキテクチャ – シナリオ 2 : Mobility Advantage Proxy としてのみ機能するセキュリティ アプライアンス



NAT/PAT を使用した Mobility Advantage Proxy

いずれのシナリオ（図 54-1 および図 54-2）でも、NAT を使用して Cisco UMA サーバのプライベート アドレスを隠蔽できます。

シナリオ 2（図 54-2）では、PAT を使用して、すべてのクライアント トラフィックを 1 つの送信元 IP に収束し、ファイアウォールが着信トラフィックのためにワイルドカード ピンホールを開く必要がないようにします。

```
hostname(config)# access-list cumc extended permit tcp any host 172.16.27.41 eq 5443
```

および

```
hostname(config)# access-list cumc extended permit tcp host 192.0.2.183 host 172.16.27.41
eq 5443
```

Cisco UMA の導入の信頼関係

Cisco UMC クライアントと ASA との間に信頼関係を確立するために、ASA は Cisco UMA サーバの証明書とキー ペアを使用します。または、ASA は Cisco UMA サーバ FQDN を使用して証明書を取得します（証明書偽装）。ASA と Cisco UMA サーバとの間では、ASA と Cisco UMA サーバは、自己署名証明書またはローカル認証局が発行した証明書を使用します。

図 54-3 に、Cisco UMA サーバの証明書を ASA にインポートする方法を示します。Cisco UMA サーバがサードパーティ CA にすでに登録している場合は、秘密キーを使用して ASA に証明書をインポートできます。これで、ASA は Cisco UMA サーバの完全なクレデンシャルを持つことになります。Cisco UMA クライアントが Cisco UMA サーバに接続すると、ASA はハンドシェイクを代理受信し、Cisco UMA サーバの証明書を使用して、クライアントとのハンドシェイクを実行します。ASA は、サーバとのハンドシェイクも行います。

図 54-3　セキュリティ アプライアンスが Cisco UMA を表す方法 – 秘密キーの共有

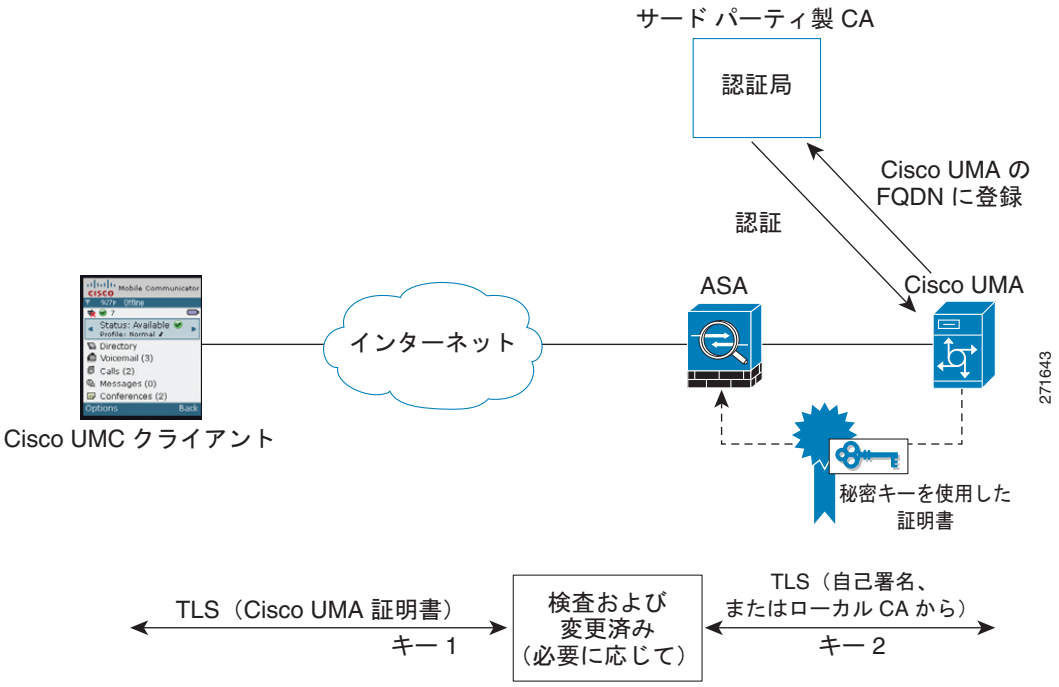
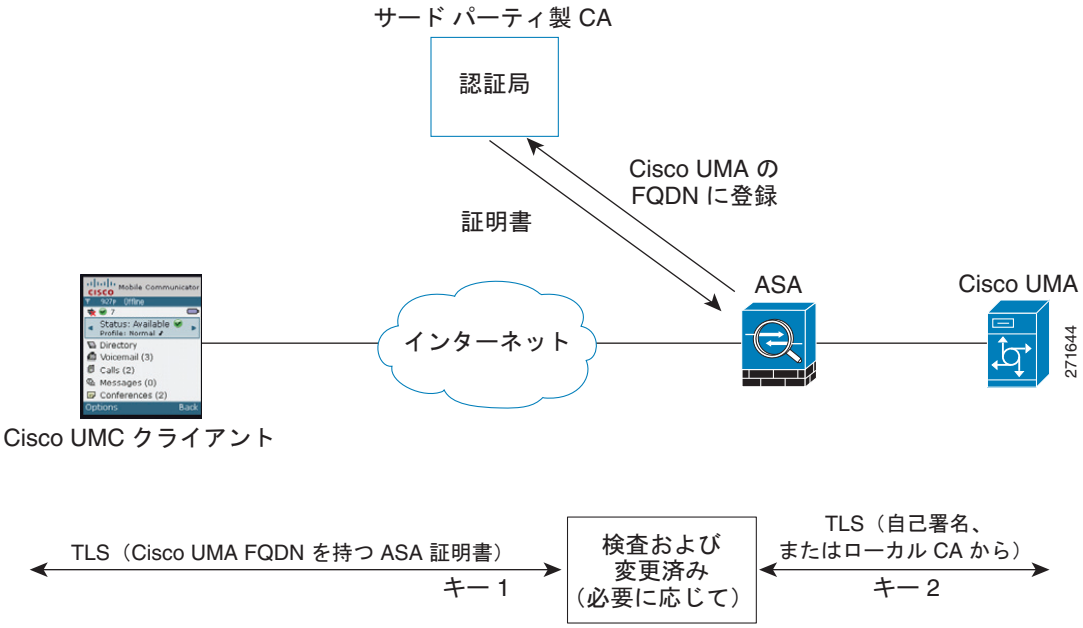


図 54-4 に、信頼関係を確立する別の方法を示します。導入に関わる各コンポーネントが新たにインストールされているため、図 54-4 は新しい場所への導入を示しています。ASA は、ASA が Cisco UMA サーバであるかのように、Cisco UMA サーバ FQDN を使用してサードパーティ CA に登録します。Cisco UMA クライアントが ASA に接続すると、ASA は、Cisco UMA サーバ FQDN を持つ証明書を示します。Cisco UMA クライアントは、通信相手が Cisco UMA サーバであるものと信じています。

図 54-4　セキュリティ アプライアンスが Cisco UMA を示す方法 – 証明書の偽装



ASA と Cisco UMA サーバの間の信頼関係は、自己署名証明書を使用して確立できます。ASA の ID 証明書はエクスポートされ、Cisco UMA サーバのトラストストアにアップロードされます。Cisco UMA サーバの証明書がダウンロードされて、トラストポイントを作成し、**crypto ca authenticate** コマンドを使用することにより、ASA のトラストストアにアップロードされます。

Cisco Mobility Advantage Proxy 機能のライセンス

ASA でサポートされる Cisco Unified Communications のプロキシ機能（Cisco 電話プロキシ、暗号化音声インスペクションの TLS プロキシ、および Cisco Presence Federation Proxy）には、Unified Communications Proxy ライセンスが必要です。ただし、バージョン 8.2(2) 以降では、Mobility Advantage Proxy に Unified Communications Proxy ライセンスは必要ありません。

次の表に、Mobility Advantage Proxy のライセンス要件を示します。

モデル	ライセンス要件
すべてのモデル	基本ライセンス

ライセンスの詳細については、第 4 章「機能のライセンスの管理」を参照してください。

Cisco Mobility Advantage の設定

この項は、次の内容で構成されています。

- 「MMP インスペクションでの TLS プロキシのイネーブル化」（P.54-9）

Cisco Mobility Advantage の設定のタスク フロー

図 54-1 と図 54-2 に示すように、TLS プロキシと MMP インスペクションを実行するように ASA を設定するには、次のタスクを実行します。

ASA と Cisco UMA サーバの間で自己署名証明書を使用するものと仮定します。

前提条件

ASA にインポートできるように、Cisco UMA サーバの証明書とキー ペアを PKCS-12 形式でエクスポートします。証明書は、Cisco UMA クライアントとのハンドシェイク中に使用されます。

ステップ 1　次のコマンドを入力し、Cisco UMA サーバのスタティック NAT を作成します。

```
hostname(config)# object network name
hostname(config-network-object)# host real_ip
hostname(config-network-object)# nat (real_ifc,mapped_ifc) static mapped_ip
```

ステップ 2　次のコマンドを入力し、Cisco UMA サーバの証明書を ASA にインポートします。

```
hostname(config)# crypto ca import trustpoint pkcs12 passphrase
[paste base 64 encoded pkcs12]
hostname(config)# quit
```

ステップ 3　Cisco UMA サーバの証明書を ASA にインストールします。「Cisco UMA サーバの証明書のインストール」（P.54-7）を参照してください。

ステップ 4　Cisco UMA サーバに接続している Cisco UMA クライアントの TLS プロキシ インスタンスを作成します。「TLS プロキシ インスタンスの作成」（P.54-8）を参照してください。

ステップ 5　MMP インスペクションで TLS プロキシをイネーブルにします。「MMP インスペクションでの TLS プロキシのイネーブル化」（P.54-9）を参照してください。

Cisco UMA サーバの証明書のインストール

Cisco UMA サーバの自己署名証明書を ASA のトラストストアにインストールします。このタスクは、ASA プロキシと Cisco UMA サーバとの間のハンドシェイク中に ASA が Cisco UMA サーバを認証するために必要です。

前提条件

ASA にインポートできるように、Cisco UMA サーバの証明書とキー ペアを PKCS-12 形式でエクスポートします。

	コマンド	目的
ステップ 1	hostname(config)# **crypto ca trustpoint** *trustpoint_name* **Example:** hostname(config)# crypto ca trustpoint cuma_server	Cisco UMA サーバのトラストポイントを作成するには、指定したトラストポイントのトラストポイント コンフィギュレーション モードに入ります。 トラストポイントは、CA が発行する証明書に基づいた CA のアイデンティティとデバイスのアイデンティティを表します。
ステップ 2	hostname(config-ca-trustpoint)# **enrollment terminal**	このトラストポイントで使用するカット アンド ペースト登録（手動登録）を指定します。
ステップ 3	hostname(config-ca-trustpoint)# **exit**	CA トラストポイント コンフィギュレーション モードを終了します。
ステップ 4	hostname(config)# **crypto ca authenticate** *trustpoint* **Example:** hostname(config)# crypto ca authenticate cuma_server Enter the base 64 encoded CA certificate. End with a blank line or the word "quit" on a line by itself [certificate data omitted] Certificate has the following attributes: Fingerprint: 21B598D5 4A81F3E5 0B24D12E 3F89C2E4 % Do you accept this certificate? [yes/no]: yes Trustpoint CA certificate accepted. % Certificate successfully imported hostname(config)#	Cisco UMA サーバに作成したトラストポイントと関連付けられている CA 証明書をインストールし、認証します。 *trustpoint* には、CA 証明書を取得するトラストポイントを指定します。名前の最大長は 128 文字です。 ASA は、base-64 形式で CA 証明書を端末に貼り付けることを求めるプロンプトを表示します。

次の作業

トラストポイントを作成し、Cisco UMA 証明書を ASA にインストールしたら、TLS プロキシ インスタンスを作成します。「TLS プロキシ インスタンスの作成」（P.54-8）を参照してください。

TLS プロキシ インスタンスの作成

Cisco UMA サーバに接続している Cisco UMA クライアントの TLS プロキシ インスタンスを作成します。

前提条件

TLS プロキシ インスタンスを作成する前に、Cisco UMA サーバの自己署名証明書を ASA のトラストストアにインストールしている必要があります。

	コマンド	目的
ステップ 1	hostname(config)# **tls-proxy** *proxy_name* **Example:** tls-proxy cuma_tlsproxy	TLS プロキシ インスタンスを作成します。
ステップ 2	hostname(config-tlsp)# **server trust-point** *proxy_name* **Example:** hostname(config-tlsp)# server trust-point cuma_proxy	TLS ハンドシェイク中に提示されるプロキシ トラストポイント証明書を指定します。 証明書は、ASA が所有している必要があります（ID 証明書）。

	コマンド	目的
ステップ 3	hostname(config-tlsp)# **client trust-point** *proxy_name* **Example:** hostname(config-tlsp)# client trust-point cuma_proxy	ASA が TLS クライアントの役割と見なす場合に、ASA が TLS ハンドシェイクで使用するトラストポイントおよび関連付けられた証明書を指定します。 証明書は、ASA が所有している必要があります（ID 証明書）。
ステップ 4	hostname(config-tlsp)# **no server authenticate-client**	クライアント認証をディセーブルにします。 TLS クライアント認証のディセーブル化は、ASA が Cisco UMA クライアントや、クライアント証明書を送信できない Web ブラウザなどのクライアントと相互運用する場合に必要になります。
ステップ 5	hostname(config-tlsp)# **client cipher-suite** *cipher_suite* **Example:** hostname(config-tlsp)# client cipher-suite aes128-sha1 aes256-sha1	暗号スイートの設定を指定します。 クライアント プロキシ（サーバに対して TLS クライアントとして機能するプロキシ）の場合、ユーザ定義の暗号スイートによってデフォルトの暗号スイートが置き換えられます。

次の作業

TLS プロキシ インスタンスを作成したら、そのインスタンスを MMP インスペクションでイネーブルにします。「MMP インスペクションでの TLS プロキシのイネーブル化」（P.54-9）を参照してください。

MMP インスペクションでの TLS プロキシのイネーブル化

Cisco UMA クライアントおよびサーバ通信は TLS を通じてプロキシ処理ができます。ここで、データを復号化してインスペクション MMP モジュールに渡し、エンドポイントに転送する前にデータを再暗号化します。インスペクション MMP モジュールは MMP ヘッダーの整合性を確認し、OML/HTTP を適切なハンドラに渡します。

	コマンド	目的
ステップ 1	hostname(config)# **class-map** *class_map_name* **Example:** hostname(config)# class-map cuma_tlsproxy	検査するトラフィックのクラスを設定します。Cisco UMA サーバとクライアントの間のトラフィックは MMP を使用し、MMP インスペクションで処理されます。 *class_map_name* には、MMP クラス マップの名前を指定します。
ステップ 2	hostname(config-cmap)# **match port tcp eq** *port* **Example:** hostname(config-cmap)# match port tcp eq 5443	MMP インスペクションのアクションを適用する TCP ポートを照合します。 MMP インスペクションの TCP/TLS のデフォルト ポートは 5443 です。
ステップ 3	hostname(config-cmap)# **exit**	クラス マップ コンフィギュレーション モードを終了します。
ステップ 4	hostname(config)# **policy-map** *name* **Example:** hostname(config)# policy-map global_policy	ポリシー マップを設定し、アクションをトラフィック クラスに関連付けます。

	コマンド	目的
ステップ 5	hostname(config-pmap)# **class** *classmap-name* **Example:** hostname(config-pmap)# class cuma_proxy	クラス マップ トラフィックにアクションを割り当てることができるように、クラス マップをポリシー マップに割り当てます。 *classmap_name* には、Skinny クラス マップの名前を指定します。
ステップ 6	hostname(config-pmap)# **inspect mmp tls-proxy** *proxy_name* **Example:** hostname(config-pmap)# inspect mmp tls-proxy cuma_proxy	SCCP（Skinny）アプリケーション インスペクションをイネーブルにし、指定したインスペクション セッションに対して電話プロキシをイネーブルにします。
ステップ 7	hostname(config-pmap)# **exit**	ポリシー マップ コンフィギュレーション モードを終了します。
ステップ 8	hostname(config)# **service-policy** *policy_map_name* **global** **Example:** service-policy global_policy global	すべてのインターフェイスでサービス ポリシーをイネーブルにします。

Cisco Mobility Advantage のモニタリング

Cisco Mobility Advantage Proxy は、IP テレフォニーと同様にデバッグできます。TLS プロキシ接続の問題をデバッグするために、SSL の syslog とともに TLS プロキシのデバッグ フラグをイネーブルにできます。

たとえば、TLS プロキシ関連のデバッグと syslog 出力だけをイネーブルにするには、次のコマンドを使用します。

```
hostname# debug inspect tls-proxy events
hostname# debug inspect tls-proxy errors
hostname# config terminal
hostname(config)# logging enable
hostname(config)# logging timestamp
hostname(config)# logging list loglist message 711001
hostname(config)# logging list loglist message 725001-725014
hostname(config)# logging list loglist message 717001-717038
hostname(config)# logging buffer-size 1000000
hostname(config)# logging buffered loglist
hostname(config)# logging debug-trace
```

TLS プロキシのデバッグ方法および出力例については、「TLS プロキシのモニタリング」（P.53-15）を参照してください。

MMP インスペクション エンジンのデバッグを行うには、**debug mmp** コマンドをイネーブルにします。

```
MMP:: received 60 bytes from outside:1.1.1.1/2000 to inside:2.2.2.2/5443
MMP:: version OLWP-2.0
MMP:: forward 60/60 bytes from outside:1.1.1.1/2000 to inside:2.2.2.2/5443
MMP:: received 100 bytes from inside:2.2.2.2/5443 to outside:1.1.1.1/2000
MMP:: session-id: ABCD_1234
MMP:: status: 201
MMP:: forward 100/100 bytes from inside:2.2.2.2/5443 to outside 1.1.1.1/2000
MMP:: received 80 bytes from outside:1.1.1.1/2000 to inside:2.2.2.2/5443
MMP:: content-type: http/1.1
MMP:: content-length: 40
```

次のコマンドを入力して、未加工のデータおよび復号化されたデータを TLS プロキシでキャプチャすることもできます。

```
hostname# capture mycap interface outside (capturing raw packets)
hostname# capture mycap-dec type tls-proxy interface outside (capturing decrypted data)
hostname# show capture capture_name
hostname# copy /pcap capture:capture_name tftp://tftp_location
```

Cisco Mobility Advantage の設定例

この項では、Cisco Mobility Advantage ソリューションによって使用される TLS プロキシの 2 つの導入シナリオに適用される設定例について説明します。シナリオ 1 では、ASA はファイアウォールと TLS プロキシの両方として機能し、シナリオ 2 では、ASA は TLS プロキシとしてだけ機能します。いずれのシナリオでも、クライアントはインターネットから接続します。

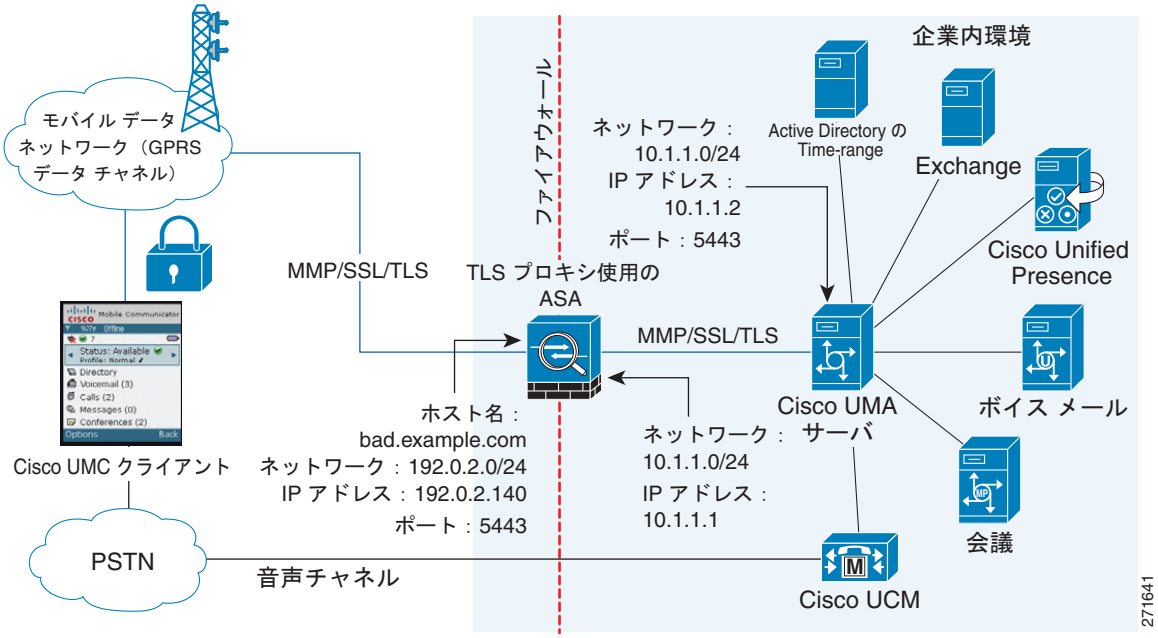
この例では、Cisco UMA サーバの証明書とキー ペアを PKCS-12 形式でエクスポートし、ASA にインポートします。証明書は、Cisco UMA クライアントとのハンドシェイク中に使用されます。

Cisco UMA サーバの自己署名証明書を ASA のトラストストアにインストールする操作は、ASA プロキシと Cisco UMA サーバとの間のハンドシェイク中に ASA が Cisco UMA サーバを認証するために必要です。Cisco UMA サーバに接続している Cisco UMA クライアントの TLS プロキシ インスタンスを作成します。最後に、MMP インスペクションで TLS プロキシをイネーブルにする必要があります。

例 1：Cisco UMC/Cisco UMA のアーキテクチャ – TLS プロキシと MMP インスペクションを使用したファイアウォールとして機能するセキュリティ アプライアンス

図 54-5（シナリオ 1 — 推奨アーキテクチャ）に示すように、ASA はファイアウォールと TLS プロキシの両方として機能します。シナリオ 1 の導入では、ASA は Cisco UMA クライアントと Cisco UMA サーバの間にあります。このシナリオでは、ASA は Cisco UMA サーバの 10.1.1.2 IP アドレスを 192.0.2.140 に変換することで、スタティック NAT を実行しています。

図 54-5　Cisco UMC/Cisco UMA のアーキテクチャ – シナリオ 1：TLS プロキシと MMP インスペクションを使用したファイアウォールとして機能するセキュリティ アプライアンス




```
object network obj-10.1.1.2-01
    host 10.1.1.2
    nat (inside,outside) static 192.0.2.140
crypto ca import cuma_proxy pkcs12 sample_passphrase
    <cut-paste base 64 encoded pkcs12 here>
    quit
!for CUMA server's self-signed certificate
crypto ca trustpoint cuma_server
    enrollment terminal
crypto ca authenticate cuma_server
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VP0NZzL+JbRTANBgkqhkiG9w0BAQUFADCB
    [ certificate data omitted ]
/7QEM8izy0EOTSErKu7Nd76jwf5e4qttkQ==
quit
tls-proxy cuma_proxy
    server trust-point cuma_proxy
    no server authenticate-client
    client cipher-suite aes128-sha1 aes256-sha1
class-map cuma_proxy
    match port tcp eq 5443
policy-map global_policy
    class cuma_proxy
        inspect mmp tls-proxy cuma_proxy
service-policy global_policy global
```

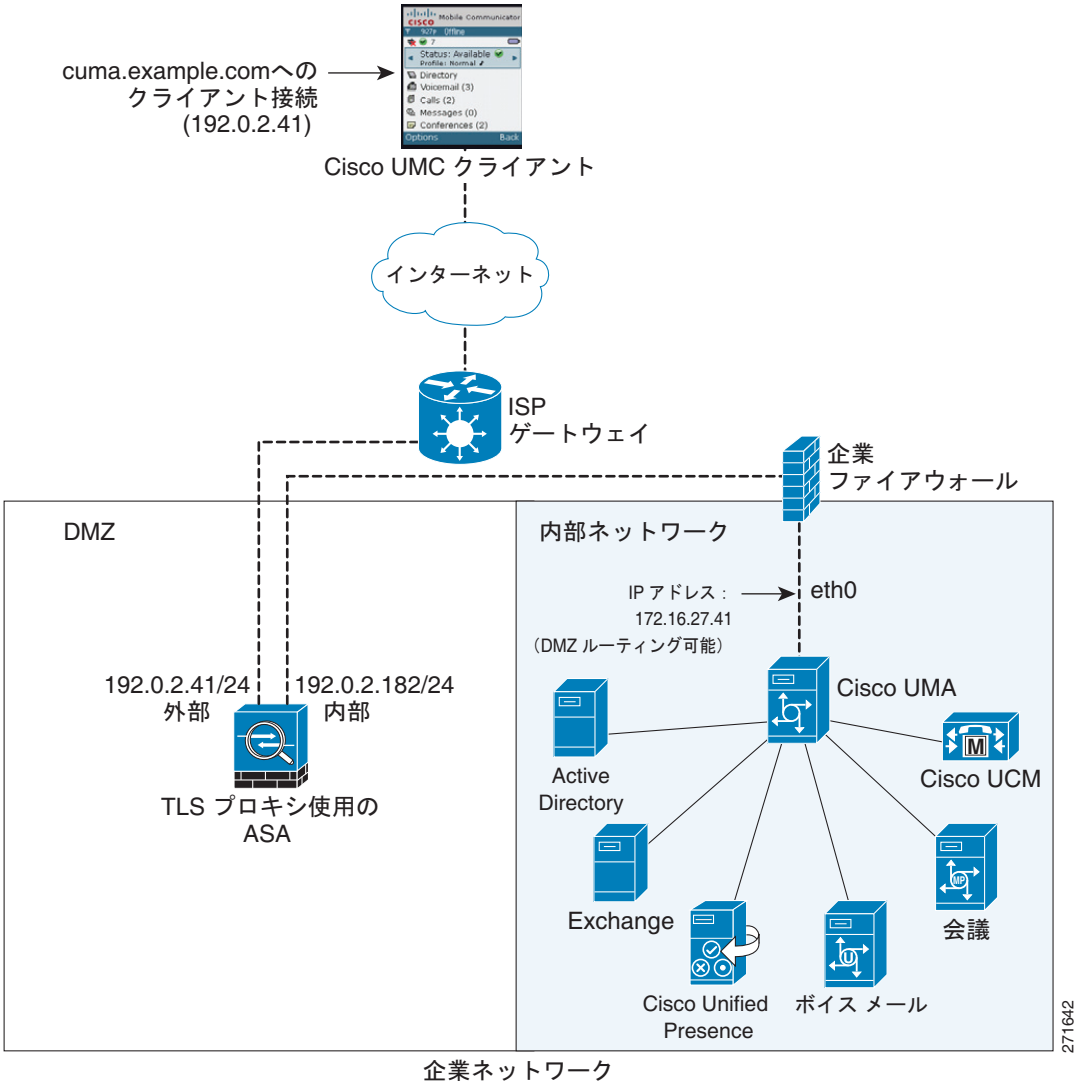
例 2：Cisco UMC/Cisco UMA のアーキテクチャ – TLS プロキシとしてだけ機能するセキュリティ アプライアンス

図 54-6（シナリオ 2）に示すように、ASA は TLS プロキシとしてだけ機能し、既存のファイアウォールを使用します。ASA と企業ファイアウォールが NAT を実行しています。企業ファイアウォールは、インターネットのどのクライアントを企業の Cisco UMA サーバに接続する必要があるのかを予測できません。したがって、この導入をサポートするために、次のアクションを実行することができます。

- 宛先 IP アドレス 192.0.2.41 を 172.16.27.41 に変換する着信トラフィックの NAT ルールを設定します。
- すべてのパケットの送信元 IP アドレスを変換する着信トラフィックのインターフェイス PAT ルールを設定し、企業ファイアウォールがワイルドカード ピンホールを開く必要がないようにします。Cisco UMA サーバは送信元 IP アドレスが 192.0.2.183 のパケットを受信します。

```
hostname(config)# object network obj-0.0.0.0-01
hostname(config-network-object)# subnet 0.0.0.0 0.0.0.0
hostname(config-network-object)# nat (outside,inside) dynamic 192.0.2.183
```

図 54-6　Cisco UMC/Cisco UMA のアーキテクチャ – シナリオ 2 : TLS プロキシとしてだけ機能するセキュリティ アプライアンス



企業ネットワーク

```
object network obj-172.16.27.41-01
    host 172.16.27.41
    nat (inside,outside) static 192.0.2.140
object network obj-0.0.0.0-01
    subnet 0.0.0.0 0.0.0.0
    nat (outside,inside) dynamic 192.0.2.183
crypto ca import cuma_proxy pkcs12 sample_passphrase
    <cut-paste base 64 encoded pkcs12 here>
    quit
!for CUMA server's self-signed certificate
crypto ca trustpoint cuma_server
    enrollment terminal
crypto ca authenticate cuma_server
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VP0NZzL+JbRTANBgkqhkiG9w0BAQUFADCB
    [ certificate data omitted ]
/7QEM8izy0EOTSErKu7Nd76jwf5e4qttkQ==
quit
```

```
tls-proxy cuma_proxy
    server trust-point cuma_proxy
    no server authenticate-client
    client cipher-suite aes128-sha1 aes256-sha1
class-map cuma_proxy
    match port tcp eq 5443
policy-map global_policy
    class cuma_proxy
        inspect mmp tls-proxy cuma_proxy
service-policy global_policy global
```

Cisco Mobility Advantage の機能履歴

表 54-1 に、この機能のリリース履歴を示します。

表 54-1　Cisco 電話プロキシの機能履歴

機能名	リリース	機能情報
Cisco Mobility Advantage Proxy	8.0(4)	Cisco Mobility Advantage Proxy 機能が導入されました。
Cisco Mobility Advantage Proxy	8.3(1)	Unified Communications Wizard が ASDM に追加されました。このウィザードを使用することにより、Cisco Mobility Advantage Proxy を設定できます。