

Unity IP ファックス発信のトラブルシューティング

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[背景説明](#)

[トラブルシューティングの前に](#)

[CslpFaxSvc サービスがインストールされ、設定されて動作していることの確認](#)

[ファックス アカウントを監視するためのメール クライアントの設定](#)

[レジストリ設定](#)

[各コンポーネントのトラブルシューティング](#)

[ステップ 1: エンドユーザからファックス アカウントにメッセージを送信できることの確認](#)

[ステップ 2: メッセージが IP Fax サービスにより処理されていることの確認](#)

[ステップ 3: レンダリング処理の確認](#)

[ステップ 4: ルータがメッセージを受信および配送することの確認](#)

[ステップ 5: ルータが DSN メッセージを送信することの確認](#)

[ステップ 6: DSN メッセージが受信されたことの確認](#)

[警告](#)

[付録](#)

[レンダー エンジントレース ファイル](#)

[Q.931 接続解除原因コード](#)

[関連情報](#)

概要

Cisco Unity ユーザから公衆電話交換網 (PSTN) のファクス機器への発信ファクスは、Cisco Unity 4.0(4)、Cisco IOS® のソフトウェア リリース 12.3(7)T 以降の Cisco Unity IP Fax Configuration Wizard でサポートされます。このサービスにより、ユーザは発信ファックスを簡単な E メールで送信できます。宛先ファクス電話番号は Eメールの件名行に含まれ、Cisco Unity IP Fax サービスによりチェックされるメールボックスに送信されます。サービスはメッセージを再フォーマットし、すべての添付ファイルを適切な形式にレンダリングし、アドレスを再設定して、メッセージを送信します。メッセージはシンプル メール転送プロトコル (SMTP) を介して、T.37 オフランプ機能が設定された Cisco IOS ゲートウェイに送信され、宛先電話番号にファクスが送信されます。

このドキュメントでは、トラブルシューティングに使用する一般的なステップを説明しています。Cisco IOSゲートウェイの初期[設定手順](#)については、『[Cisco Unity IPファックスアウトバウン](#)

[ドゲートウェイの設定例](#)』を参照してください。また、Cisco Unity の [IP Fax Configuration Wizard](#) も参照してください。

[前提条件](#)

[要件](#)

次の項目に関する知識があることが推奨されます。

- Cisco Unity IP ファックス発信プロセスの通常のマニュアル [IP Fax Configuration Wizard](#) と『[Cisco Unity IP ファックス発信ゲートウェイの設定例](#)』を参照してください。
- Microsoft Exchange 2000/2003 の SMTP の基本的な知識
- Cisco IOS Voice over IP (VoIP) の設定

完全な稼働システムを準備するには、このドキュメントで定義されているゲートウェイ設定とともに、Cisco Unity IP Fax Configuration Wizard バージョン 2.0.0.19 以降を搭載したアクティブな Cisco Unity サーバを使用します。IOS ゲートウェイ側では、パートナー メッセージストアとして Microsoft Exchange 2000 または 2003 を採用する Cisco Unity バージョン 4.0(4) 以降とともに、T.37 OffRamp 機能と Cisco IOS ソフトウェア リリース 12.3(7)T 以降をサポートする Cisco IOS ルータが必要です。

注：T.37 OffRampは、メディアゲートウェイコントロールプロトコル(MGCP)ネットワークではサポートされていません。T.37 を使用する際のプラットフォームとその他の制限事項の詳細については、『[T.37 ストア アンド フォワード ファックス設定ガイド](#)』を参照してください。

[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いています。

- IOS ゲートウェイとして Cisco 3725
- Cisco IOS ソフトウェア リリース 12.3(8)T4

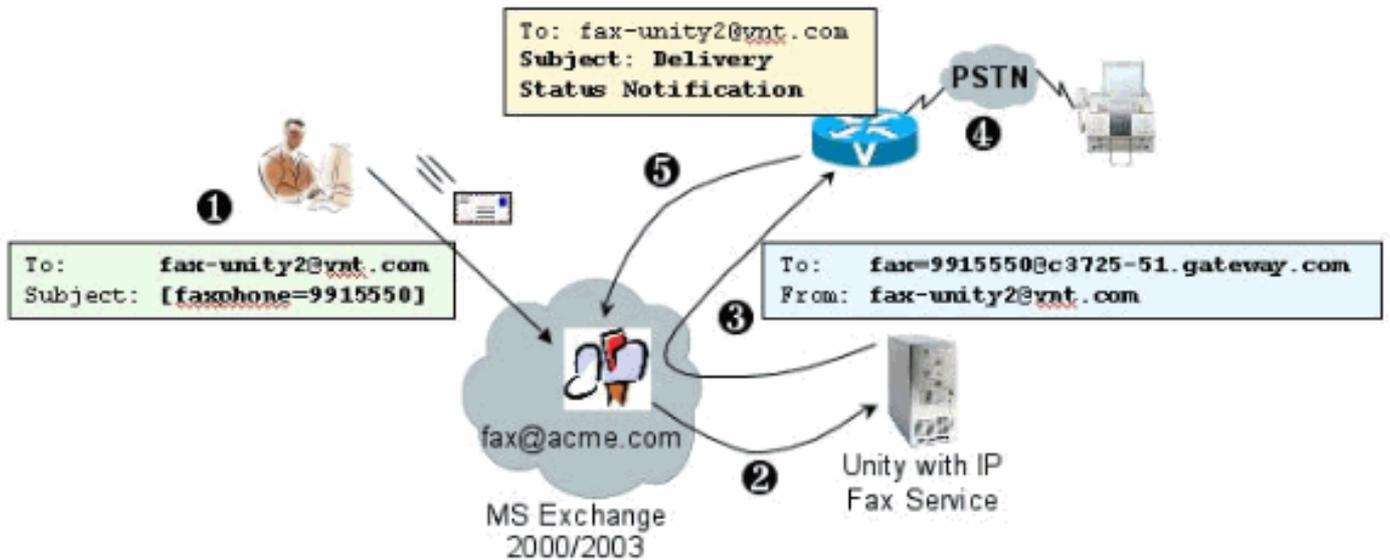
このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

[表記法](#)

ドキュメント表記の詳細は、『[シスコ テクニカル ティップスの表記法](#)』を参照してください。

[背景説明](#)

トラブルシューティング手順を説明するには、この機能の基本的な動作を理解することが重要です。続いて、特定のコンポーネントにまで問題を絞り込み、そのコンポーネントに関する詳細な情報を収集します。



1. システム上のエンドユーザは、ファックス メールボックスに E メールを送信します。このメッセージはファックス メールボックス (fax-unity2@vnt.com など) に対する「subject」内に [FAXPHONE=9915550] が含まれている必要があります。9915550 は宛先ファックスマシンの番号です。ゲートウェイで設定されているダイヤルプランに基づいて、この番号には前に 9 が 1 つ付けられるか、あるいは特定のディジットストリングのみが受け入れられます。いったん送信されると、そのメッセージは Microsoft Exchange によりこのメールボックスに配送されます。注：電子メールが別のドメインから送信される場合は、ドメインネームシステム(DNS)とリモートメールシステムが、FAXメールボックスが存在するドメインとメールアカウントにメールを配信するように設定されます。
2. Cisco Unity IP Fax サービスは定期的にファックス メールボックスにログインし、メッセージを取得します。この時点で添付ファイルが読み取られます。Cisco Unity サーバがそのサイズの添付ファイルを処理できるという前提のもとで、メッセージは Cisco Unity サーバ上でレンダリングされます (たとえば TIF ファイルに変換されます)。
3. 添付ファイルがレンダリングされると、メッセージは IOS ゲートウェイに送信されます。Microsoft Exchange は、メッセージを送信できるように設定されている必要があります。メッセージは、fax=9915550@c3725-51.gateway.com のように、受取人宛になっています。通常、これは当該ホスト (c3725-51.gateway.com) 向けの A レコードと Mail Exchange Record (MX; メール交換レコード) が DNS に追加され、メッセージをルーティングするために Exchange に SMTP Connector が追加されることを意味します。エンドユーザがゲートウェイに E メールを直接送信することはないため、ここで使用される E メールアドレスとホスト名はユーザが読みやすい形式である必要はありません。
4. IOS ゲートウェイは T.37 OffRamp 機能に対して設定されています。この機能により IOS ゲートウェイは、SMTP メッセージを受信し、「fax=9915550」アドレスを解釈し、(可能な番号操作の後に) 9915550 番号にコールを発信し、受信側ファックスマシンに TIF ファイルを転送することができます。注：ゲートウェイはこの機能を使用するために MGCP を使用できません。ただし、回線はファックス専用である必要はありません。音声とファックスは同じ音声回線上でともに使用されます。ルータ上のダイヤルピア設定で、コールで実際に何が行われるかが決定されます。
5. ファックスが配送されると、ゲートウェイは Delivery Status Notifications (DSN) メッセージをファックス メールボックス (fax-unity2@vnt.com) に返送します。これにより、ファックスが正しく配送されたかどうか明らかになります。IP Fax サービスのログイン時に、DSN メッセージが取得されます。ファックスが正しく配送されなかった場合、原因コードが調査されます。この原因コードに基づいて、メッセージが再送信される (通常はリモート

ファックスがビジー状態の場合)か、あるいはメッセージがエンドユーザに返送されてユーザに問題が通知されます。このような処理がないと、IP Fax サービスは、ファックスが正しく配送されたかどうかまったく分かりません。

このアーキテクチャの利点の 1 つに、通常はどのような問題でも関連するコンポーネントに切り分けられる点があります。

トラブルシューティングの前に

CslpFaxSvc サービスがインストールされ、設定されて動作していることの確認

始める前に設定を確認します。

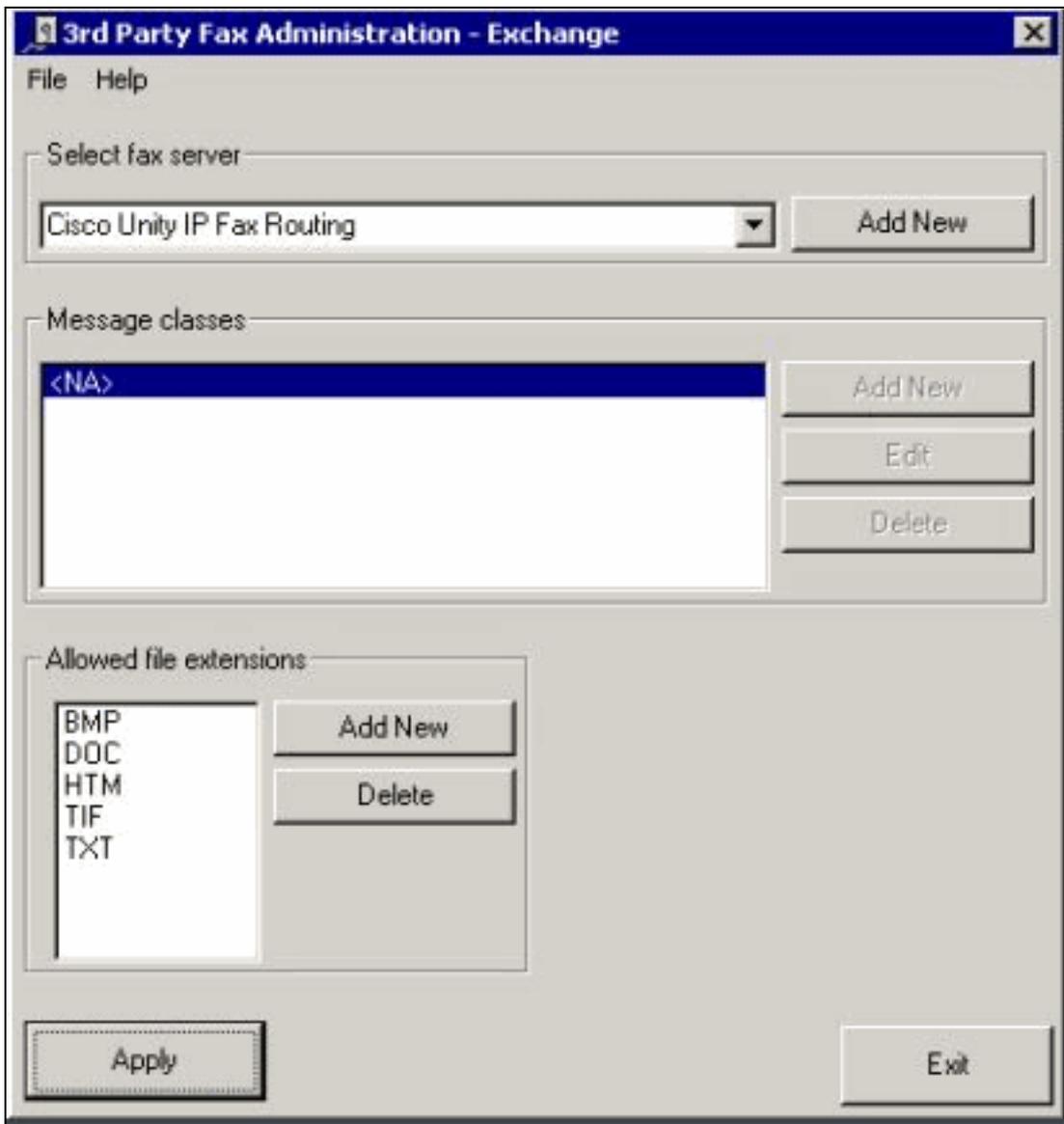
ゲートウェイを正しく設定します。『[Cisco Unity IP ファックス発信ゲートウェイの設定例](#)』および [IP Fax Configuration Wizard](#) を参照してください。

メールがエンドユーザのメールストアから、ファックス メールボックスが存在するメールストアに配送されるよう、Exchange を設定します。発信元にステータス メッセージを返送できるように、メールは逆方向でも配送されます。メールは、ファックス アカウントからゲートウェイへ、およびその反対方向で配送できる必要があります。これには、DNS と Exchange の設定が関与します。

IP Fax サービスが正しくインストールされていることを確認します。セットアッププログラムをアクティブにした後、Cisco Unity で設定ウィザードを実行し、[Start] > [Programs] > [Unity] > [Cisco Unity IP Fax Configuration Wizard] を選択します。

IP Fax サービスは、ユーザがそれに対して送信した内容をレンダリングすることはできません。IP Fax サービスは、E メールに添付されているファイルを受信した場合、まず拡張子を認識できることを確認します。これらは [サードパーティ製ファックス管理ツール](#) で設定されています。使用できるデフォルトの拡張子は、BMP、HTM、TIF、および TXT です。DOC 拡張子が必要な場合、次の例のように、ツールにその拡張子を追加する必要があります。

注：一部のリリースでは、サードパーティ製ファックス管理ツールのリストの最後の内線が正しく動作しない問題が発生しています。詳細については、「[注意](#)」のセクションを参照してください。



選択されているファックス サーバは Cisco Unity IP Fax Routing です。DOC は、Allowed file extensions 領域に追加されています。これは、IP Fax サービスが添付ファイルをレンダリング エンジンに配送することを意味します。ファイルが配送されると、レンダリング エンジンがドキュメントを開いてプリント ドライバでファイルに出力できるように、レンダリング エンジンは、拡張子タイプに関連付けられたアプリケーションを利用します。デフォルトでは Cisco Unity サーバ上に WordPad アプリケーションがインストールされていて、WordPad アプリケーションは Microsoft Word ドキュメントの大部分を開くため、通常は DOC 拡張子が使用できます。Acrobat Reader などのアプリケーションは、PDF ファイルのファックス送信に役立ちます。詳細については、このドキュメントの「[注意](#)」のセクションを参照してください。

注： Cisco for.doc拡張機能で公式にサポートされているアプリケーションはWordPadだけです。WordviewやMicrosoft Wordなどのアプリケーションをdoc拡張子とともに使用すると、レンダリングエンジンが失敗する可能性があります。

注： Cisco UnityサーバにMicrosoft Officeをインストールしないでください。インストールすると、Cisco Unity の正常な動作が停止します。Cisco Unity サーバにインストールするのは、サポートされるサードパーティ製アプリケーションだけです。Cisco Unity サーバにサードパーティ製ソフトウェアをインストールする際には注意してください。

[Start] > [Administrative Tools] > [Services]からCslpFaxSvcがアクティブで、[Log On As]フィールドがAvCsMgrサービスと同じであることを確認します。AvCsMgr サービスが起動していない場合、問題が存在する可能性があります。このことは CslpFaxSvc にも影響します。 [Cisco Unity](#)

[3.1\(x\)および4.x:AvCsMgrサービスが起動に失敗』](#)を参照してください。

ファックス アカウントを監視するためのメール クライアントの設定

IP Fax サービスとゲートウェイのインストール後も正しく動作しない場合、最初のステップは、Microsoft Outlook Express (Internet Mail Access Protocol (IMAP) 使用) または Outlook (あるいは別のメール プログラム) を使用して IP Fax メールボックスを調査することです。エンドユーザがこのメールボックスにメッセージを送信すると、IP Fax サービスがこのアカウントからのメッセージを確認して送信するため、メッセージが到着し、さらに正しく送信されることを確認します。ここで見られるすべてのものは、ログファイルとExchange Message Tracking Centerを使用して診断できます。しかし、これは製品の動作を説明するのにも役立ちます。

Outlook Express は Cisco Unity サーバに自動的にインストールされます。Outlook は、Cisco Unity の Unified Messaging 配備で選択されるメーラーです。

注意 : Cisco UnityサーバにMicrosoft Outlookをインストールしないでください。インストールすると Cisco Unity の動作が停止します。また、Exchange サーバ上にも Outlook をインストールしないでください。Exchangeサーバに接続するには、クライアントマシンにインストールしてから設定する必要があります。

Outlook Express に 1 つまたは複数のメール アカウントを追加する場合、必ず IMAP を指定してください。POP を指定すると、すべてのメール メッセージがメールストアからすぐにダウンロードされる原因になります。この場合、IP Fax サービスでは新しいメッセージがまったく検出されません。IP Fax サービス アカウントを指定します。パスワードが不明である場合は、Active Directory Users and Computers にアクセスし、IP Fax アカウントを見つけ、パスワードをリセットすると安全です。

このドキュメントに記載されているトラブルシューティング方法が可能な唯一の方法ではありません。たとえば、IP Fax アカウントにアクセスしていて、Outlook または Outlook Express 経由でログインしている場合、その場所からエンドユーザにテスト用 E メールを直接送信する方法があります。また、ゲートウェイに E メールを直接送信する方法もあります。これにより、ある要素または別の要素が正しく動作していることが明らかになるため、問題が絞り込まれます。

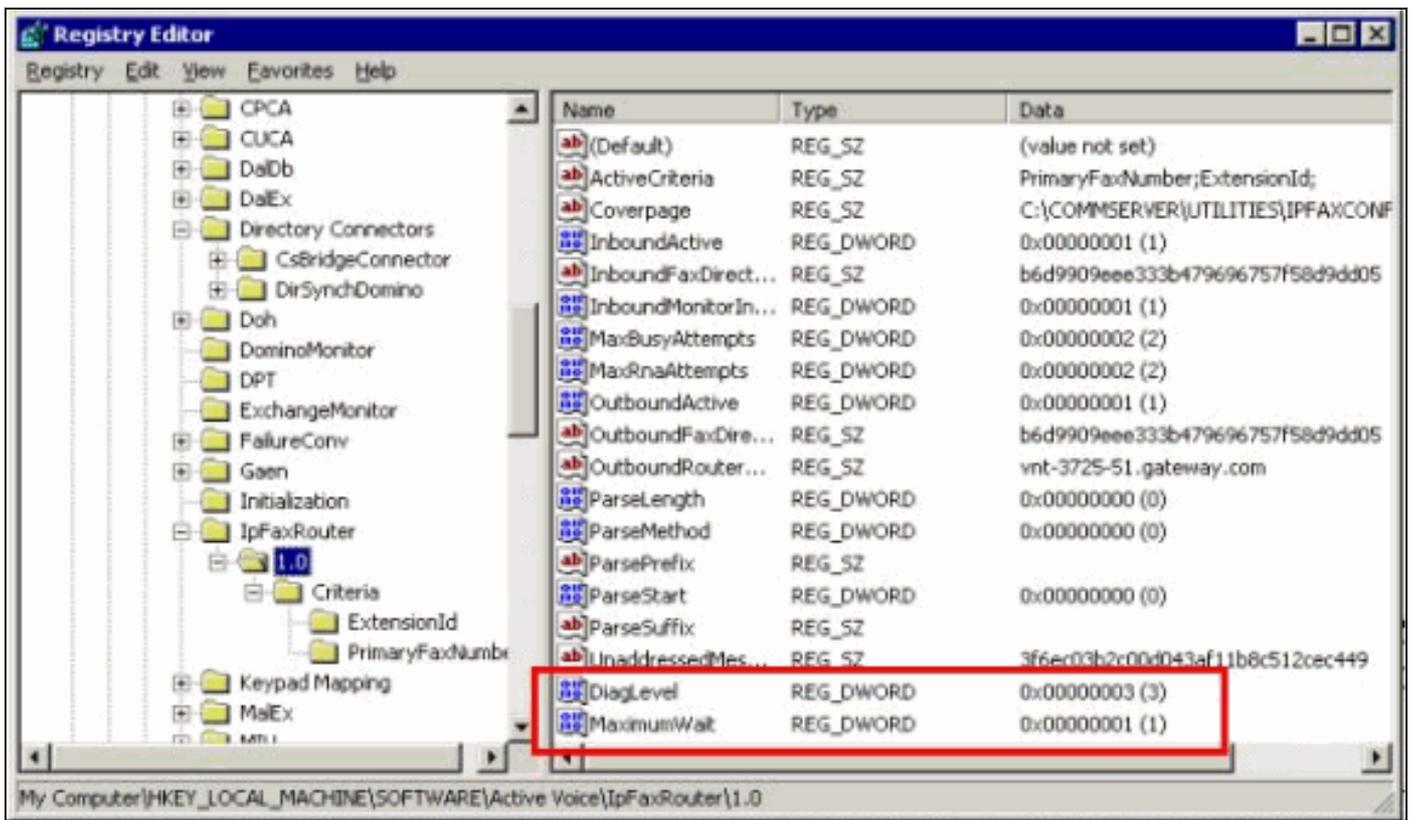
レジストリ設定

トラブルシューティングに役立てるため、より多くの出力がログ記録されるように診断レベルを変更します。CSIPFaxSvcが稼働するCisco Unityサーバで有効にするには、**Start > Run**の順に選択し、**regedit**と入力します。レジストリエディタ・ツールが表示されたら、**HKEY_LOCAL_MACHINE\SOFTWARE\Active Voice\IPFaxRouter\1.0**を選択します。

[Edit] > [New] > [DWORD Value] を選択します。指定されている名前は DiagLevel です。これが作成されている場合は、このキーをダブルクリックします。Value Data フィールドに 3 と入力します。これが最大の診断レベルです。後で 0 (デフォルト) に変更できます。このように変更してもサービスを再起動する必要はありません。

注意 : 間違ったレジストリキーを変更したり、間違った値を入力したりすると、サーバが誤動作する原因になります。レジストリ キーを編集する前に、問題が発生した場合の復元方法を確認してください。詳細は、レジストリ エディタのヘルプにある復元に関するトピックを参照してください。通常、Cisco Unity サーバのバックアップでは、レジストリのバックアップは実行されません。Cisco Unity のフェールオーバー システムに関して、1 台の Cisco Unity サーバ上でレジストリを変更した場合は、他の Cisco Unity サーバにも手動で変更を加える必要があります。これは、レジストリの変更が複製されないためです。レジストリ キーの設定の変更方法に関する詳細は

、Cisco テクニカルサポートにご連絡ください。



ある状況で便利なもう 1 つのキーは MaximumWait キーです。存在しない場合、このキーのデフォルトは 24 (時間) です。このキーは、未配送メッセージがエンドユーザに送信されるまで、メッセージを保持する最大時間を指定します。このキーはトラブルシューティングで直接役立つことはありませんが、頻繁に必要になります。

OutboundRouterDomainキーには、Cisco IOSゲートウェイに送信されるファックス配信用のアドレスにメールメッセージが配信される場合に、メールサフィックスとして使用される電子メールアドレスが含まれています。これを行うには、この例の値vnt-3725-51.gateway.comがDNS経路で解決される必要があります。このエントリにpingを実行すると便利です。このエントリが正しくない場合、電子メールが送信者に返され、ゲートウェイエントリが解決できないことを通知します。

ヒント：この事実はトラブルシューティングに役立ちます。Cisco IOSゲートウェイを除くすべてのエントリがインストールされている場合は、DNSに存在しないエントリなど、一時的にこのレジストリエントリを偽の値に変更できます。続いてファックス メールボックスにファックス メッセージを送信します。CslpFaxSvc サービスがメッセージを取得し、添付ファイルをレンダリングしますが、ゲートウェイの特定に失敗します。ゲートウェイに送信されるファックスメッセージのレンダリングされたTIF添付ファイルとともに、Non-Delivery Receipt(NDR)通知が送信者に返されます。これを行った場合、次のことを確認しました。

- エンドユーザからファックス アカウントに E メールを送信できる。
- ファックスを正しくレンダリングできる。
- ファックス アカウントからエンドユーザに E メールを返送できる。

ファックス アカウントからゲートウェイへ、およびゲートウェイからファックス アカウントへのメッセージ送信以外は、すべてが除外されます。

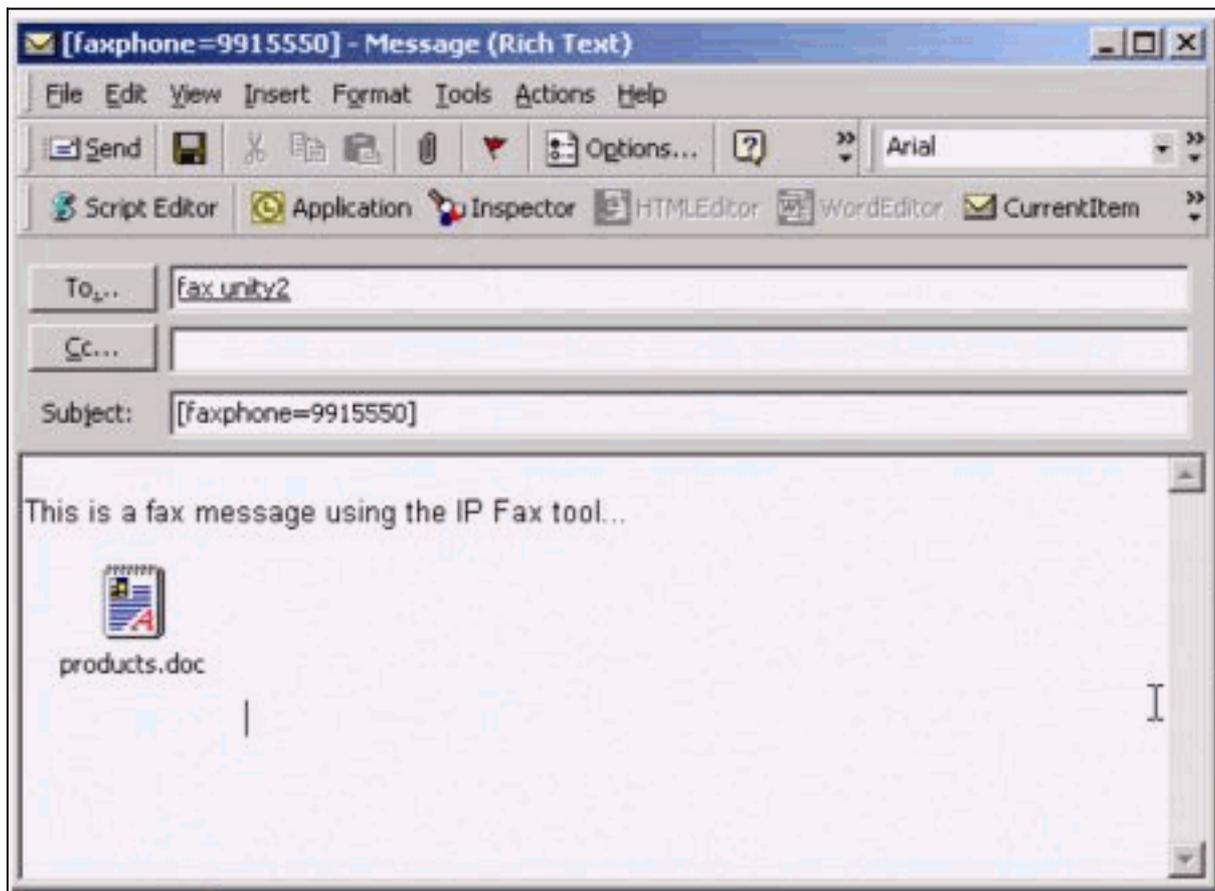
Coverpage キーには、レンダリングされるカバー ページの場所が含まれます。送信されるすべてのファックスには、ページ数などの情報が記載されたカバー ページが含まれます。このページを

編集し、カスタム情報や会社のロゴを挿入することができます。キーが、存在しないファイルに変更されると、カバー ページは送信されません。

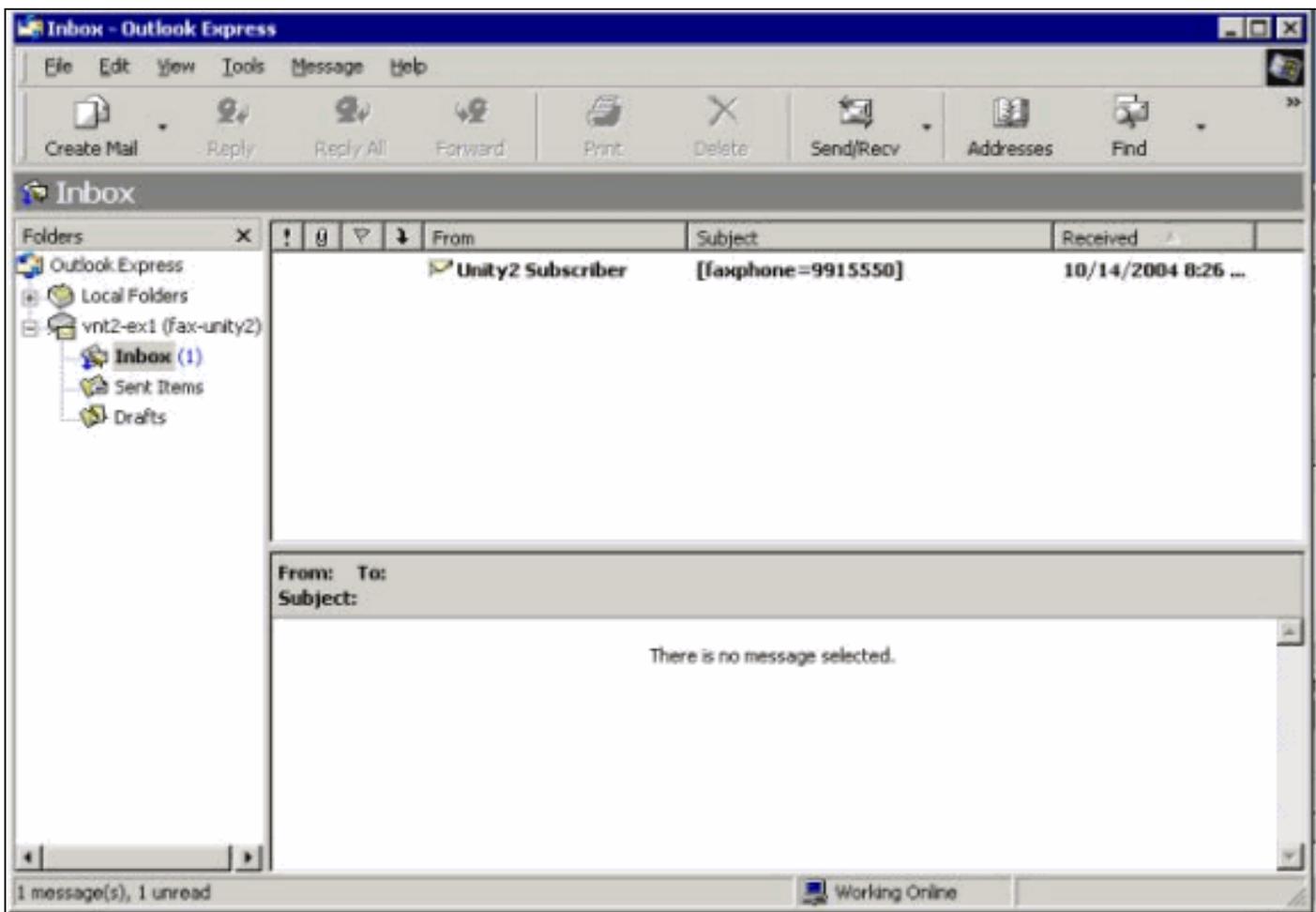
各コンポーネントのトラブルシューティング

ステップ 1: エンドユーザからファックス アカウントにメッセージを送信できることの確認

ユーザからファックス アカウントにメッセージを送信します。次の図では、メッセージの宛先はファックス メールボックスである fax-unity2 です。ユーザが呼び出す番号は、「subject」行で指定されています。この場合は 9915550 です。ユーザがファックス送信するドキュメントが添付されています。

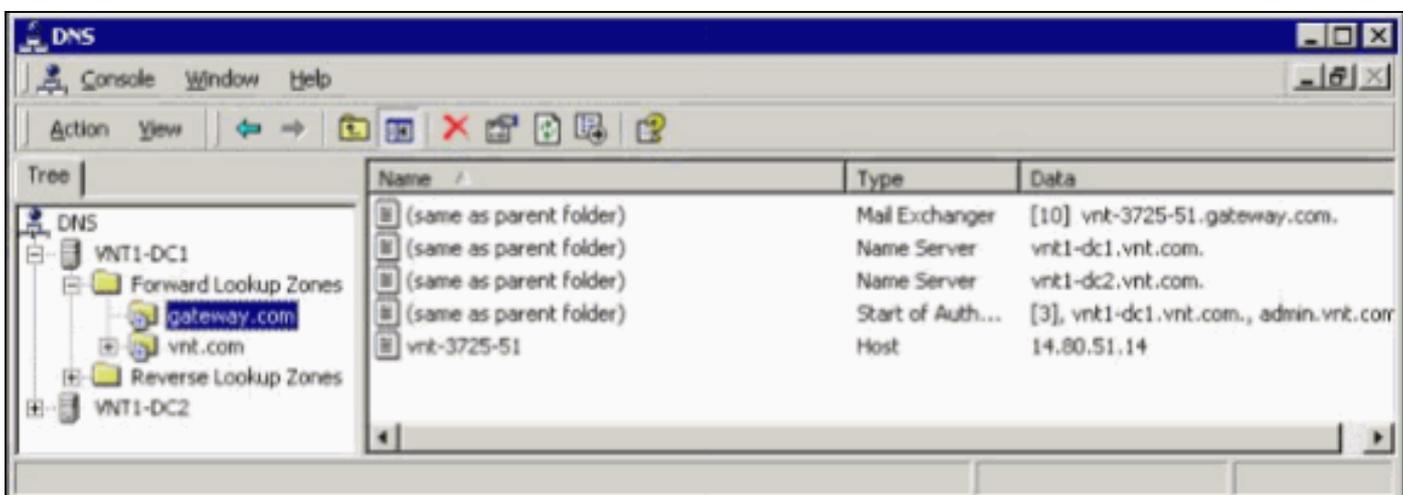


このメッセージが送信されると、メッセージは最初に設定されたファックス メールボックスに到着します。次の図は Outlook Express から見たメールボックスを示しています。

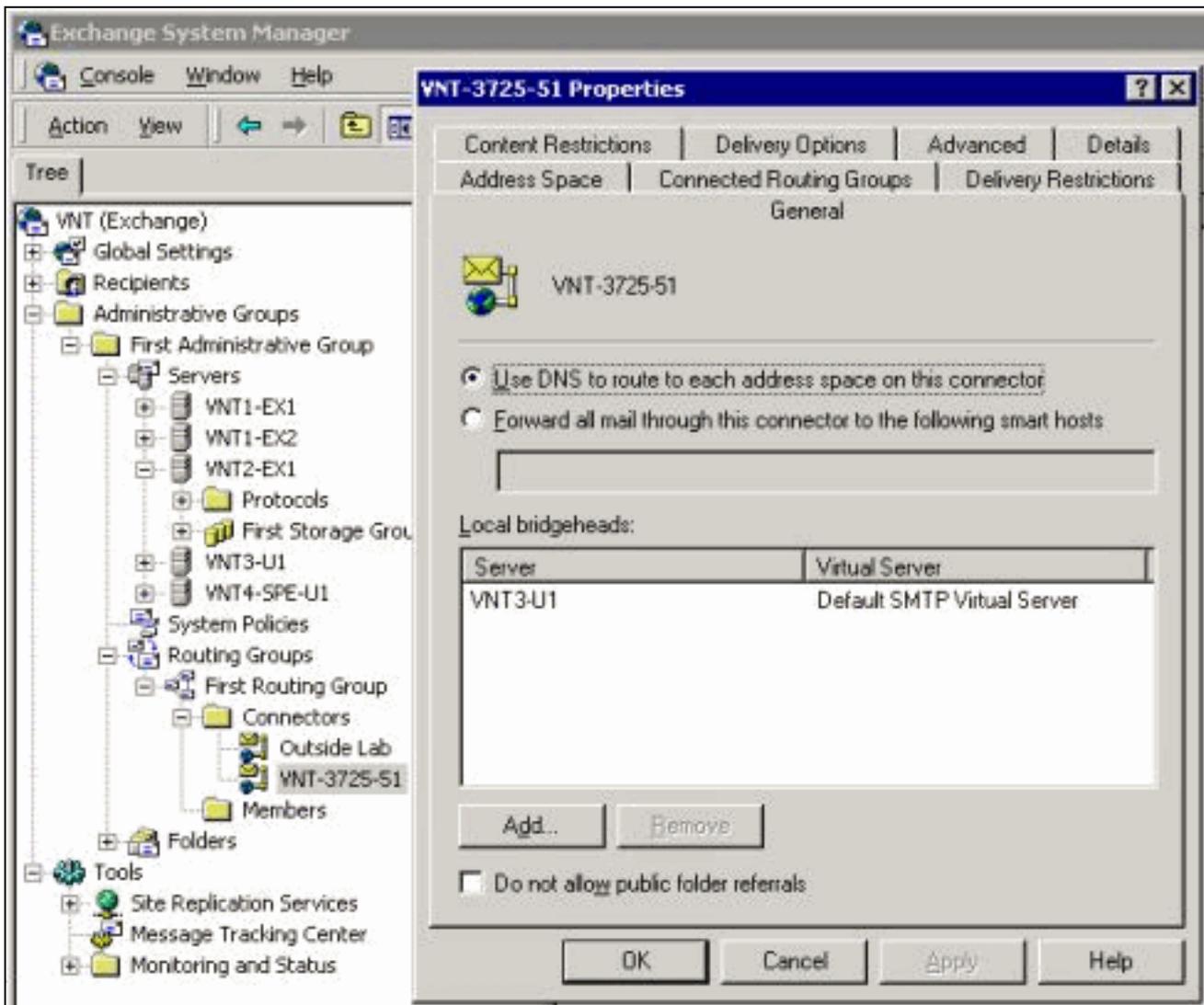


メッセージが到着していない場合は、基本的な E メール配送の問題が存在します。送信者が何らかの配信障害メッセージを受信したかどうかを確認します。Microsoft Exchange 2000/2003には Exchange Message Tracking機能があり、メール配信に関する詳細を確認できます。送信者と FAXアカウントが2つの異なるドメイン（たとえば、a.comとb.com）にある場合は、送信先メールサーバにメールを配信するように送信側メールシステムが適切に設定されていることを確認します。これには、DNS で A および MX レコードを設定し、メールをルーティングするために SMTP コネクタを追加する作業が含まれます。また、未配送メッセージがエンドユーザに返送されるよう、メールは逆方向でも送信可能である必要があります。

次の図に、A および MX レコードを設定した IOS ゲートウェイの DNS エントリを示します。



当該メール ホスト用に Exchange には SMTP コネクタが必要です。



アドレス空間へのルーティングには DNS が使用されます。アドレス空間は c3725-51.gateway.com として設定されています。これは多くの設定の 1 つにすぎません。また、テスト用の環境設定の例にすぎません。通常の企業での Exchange 環境は、はるかに複雑です。ポイントは、ユーザからファックス メールボックスへのメールを取得することを確認する点にあります。これらのトピックに関する詳細情報は、Microsoft の Web サイトを確認してください。

メールが、エンドユーザからファックス アカウントへ、およびファックス アカウントから最終的なエンドユーザに送信されることを確認します。

ステップ 2 : メッセージが IP Fax サービスにより処理されていることの確認

IP Fax サービスの設定時には、ポーリング間隔を指定します。これは、IP Fax サービスがファックス メールボックスにログインし、新しいメッセージを確認する頻度を示します。Cisco は、製品の正しい動作が確認されるまで、この値を低く設定することをお勧めします。

最適な診断ログファイルは IPFaxSvc<date/time>.log です。このファイルは \Commserver\logs ディレクトリにあります。

注 : このトレースサンプルでは、一部の情報が削除されています。

各間隔でログ ファイルがログインするメールボックスを確認できます。また、メールボックスが発信ファックスを処理していることも確認できます。

注：すべてのトレースは、前述のDiagLevel = 3レジストリ設定を反映しています。

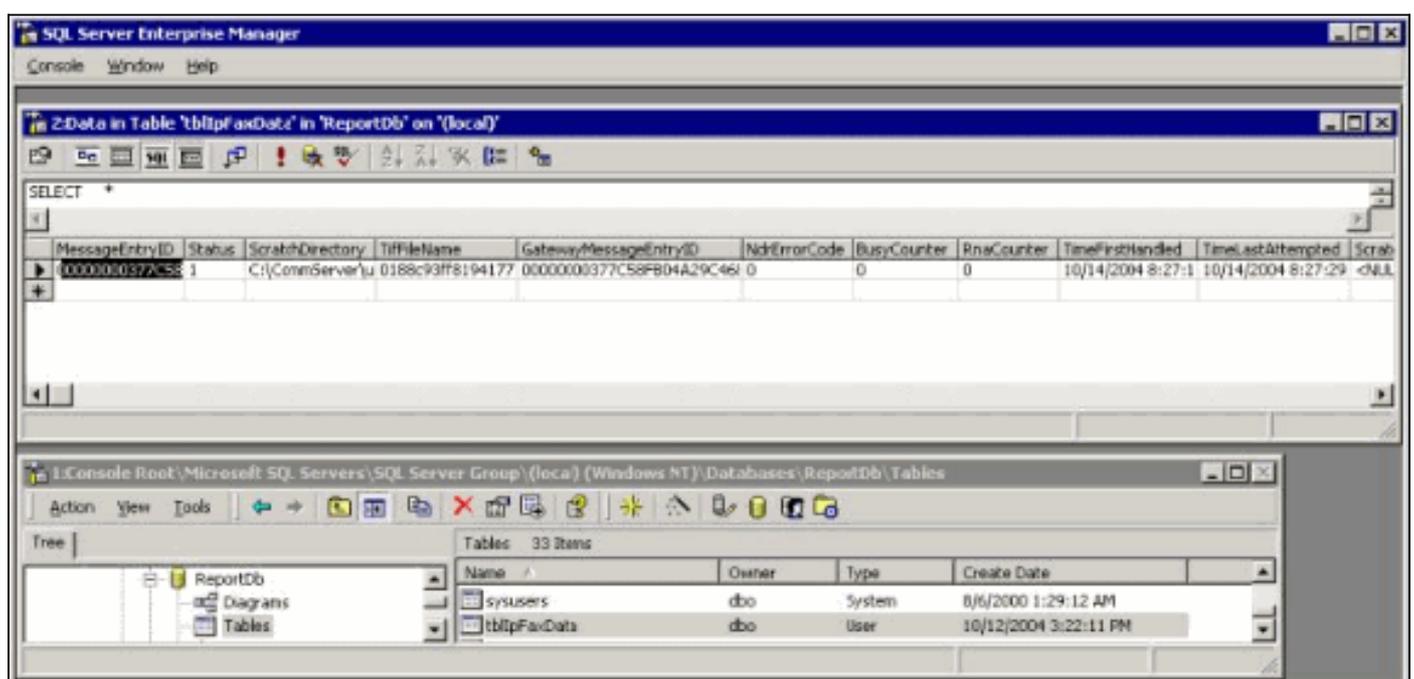
```
"Logging on to mailbox [VNT2-EX1\fax-unity2]"
"CipFaxCdoHelper::LogonToMailbox() name returned from Logon is [fax unity2]"
"Processing outbox, message count= 1"
"CipFaxCdoHelper::ProcessFaxOutbox() processing message receipts"
"CipFaxCdoHelper::DetermineMessageDisposition() current message
subject=[ [faxphone=9915550] ]"
"CipFaxCdoHelper::DetermineMessageDisposition() setting disposition to OUTBOUND"
"CipFaxCdoHelper::ProcessFaxOutbox() message looks like outbound fax - not
processing as receipt message"
```

発信プロセスでは、正しいメールボックスへのログインに成功し、そこにメッセージがあることを認識し、「subject」欄を取得して、それが発信ファックスであるとみなすことが重要です。

```
"_RecordsetPtr::Open(Select * from tblIpFaxData where MessageEntryID =
'00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C512000000
05E5B20000E7E65E238D0DEF46B5F8EAF983C6C51200000005F55A0000') returned [0] rows -
Code: 0x80004005. Method: CipFaxAdoHelper::GetRecordSet. [IpFaxAdoHelper.cpp::403]"
```

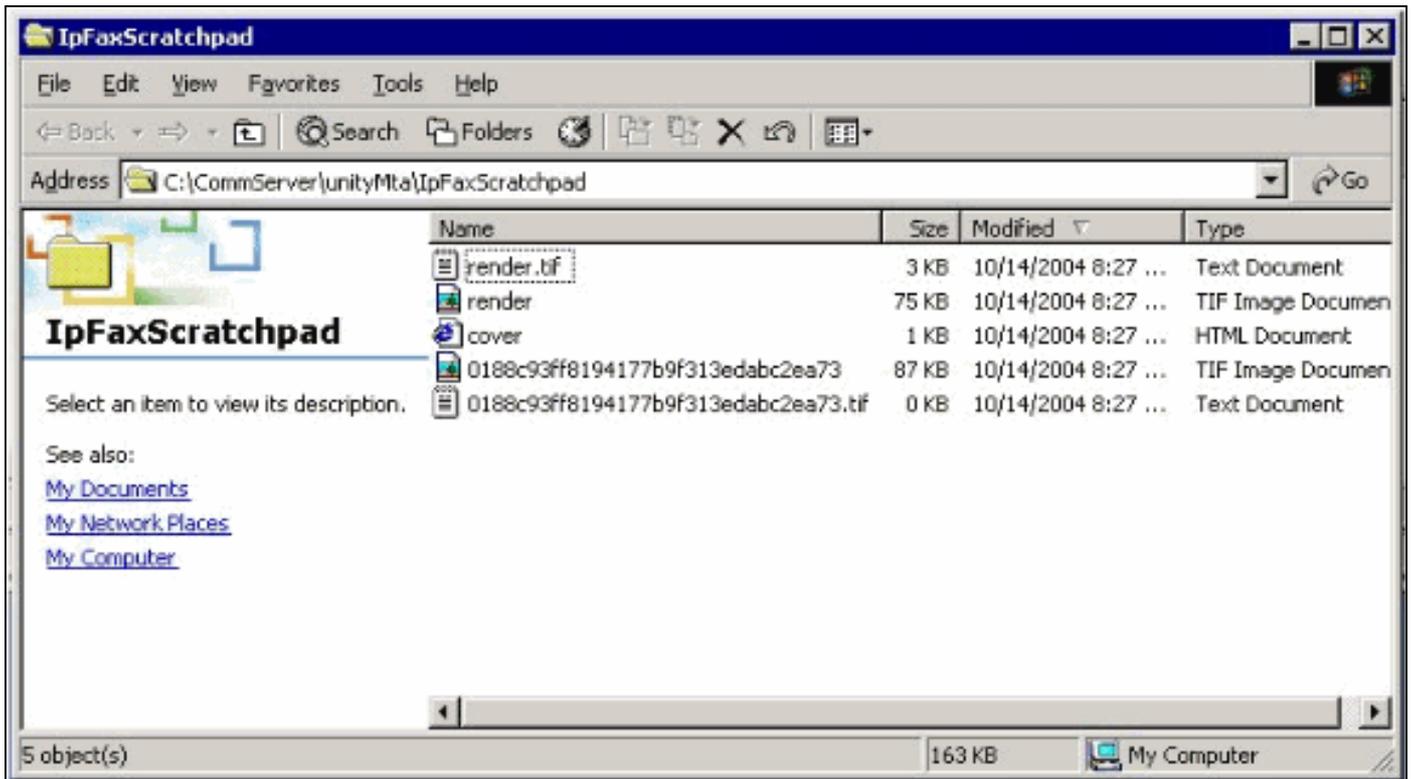
```
"CipFaxAdoHelper::Execute() Cmd=[INSERT INTO tblIpFaxData (MessageEntryID, Status,
ScratchDirectory, TifFileName, NdrErrorCode, BusyCounter, RnaCounter,
TimeFirstHandled) VALUES ('00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0
DEF46B5F8EAF983C6C51200000005E5B20000E7E65E238D0DEF46B5F8EAF983C6C51200000005F55A
0000', '0', 'C:\CommServer\unityMta\IpFaxScratchpad', '0188c93ff8194177b9f313edab
c2ea73.tif', '0', '0', '0', '2004-10-14 08:27:17')]"
```

この時点でメッセージが読み取られます。第1の文は、当該 MessageEntryID (メッセージの MAPI ID) を持つメッセージが存在するかどうかを確認するため、Cisco Unity サーバの Reports データベースにある tblIpFaxDatatable に対して、サービスが SQL クエリーを実行することを示しています。サービスはこのようなエントリを検出しないため(0行が返される)、この情報を持つ新しい行を追加します。MessageEntryID は、ゲートウェイに送信される Eメールの Subject フィールドとして使用されます。SQL を直接調べると、次のことが確認できます。



各メッセージを追跡するために SQL が使用されます。ファックス メールボックスが空でない場合、それぞれの間隔で SQL はファックス メールボックスにアクセスし、ファックス メールボックスを占めているのがこのメッセージ ID に基づく各メッセージであることを確認します。この場合、添付ファイルはスクラッチ ディレクトリ C:\CommServer\unityMta\IpFaxScratchpad にコピーさ

れます。この同じメッセージに DSN が存在する場合、必要なのは BusyCounter または類似の要素を更新することだけです。ただし、これはサービスが各メッセージを追跡する手法です。ファイルがスクラッチ ディレクトリにコピーされたため、レンダリングが行われます。



```
"CIpFaxCdoHelper::RenderCurrentMessage() scratch file=[C:\CommServer\unityMta\IpFaxScratchpad\render.tif]"
```

```
"CIpFaxCdoHelper::RenderCurrentMessage() outputting message body to file name=[C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt]"
```

```
"Rendering file [C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt] to [C:\CommServer\unityMta\IpFaxScratchpad\render.tif]"
```

```
"CIpFaxCdoHelper::RenderCurrentMessage() added [1] pages to TIFF"
```

レンダリング プロセスが始まります。すべてのファックスには、カバー ページ、メッセージの本文があるページ、および一部またはすべての添付ファイルが含まれます。カバー ページには、ファックス送信されるページ数が表示されます。本文は、常に最初に送信されるため、最初にレンダリングされます。一度にレンダリングされるのは 1 つのメッセージだけなので、メッセージの本文 MsgBody.txt と一時出力ファイル render.tif は、常に呼び出されます。レンダリングは ImageMaker ソフトウェアにより行われます。各レンダリング プロセスの完了後、レンダリングされたページ数が返されます。この場合は 1 ページです。

ステップ 3 : レンダリング処理の確認

ImageMaker レンダリング エンジン は プリント ドライバ です。このエンジンは入力ファイルを取得し、それに関連付けられたデフォルト アプリケーション (Wordpad、Notepad など) を開き、ImageMaker ドライバを使用してファイルに出力します。これにより TIF ファイルが作成されます。

完全な診断ファイルは「[付録](#)」セクションにあります。レンダリング エンジンに関連する問題を Cisco テクニカルサポートに送信する場合は、次のファイルが必要です。

- C:\ImgMaker\Conversion_Toolkit\log\ODCDAEMON.log : バッチファイルを使用して変換を行うデーモン
- C:\log\ImageMakerDocCnvr Driver.txt : ドライバレベル情報

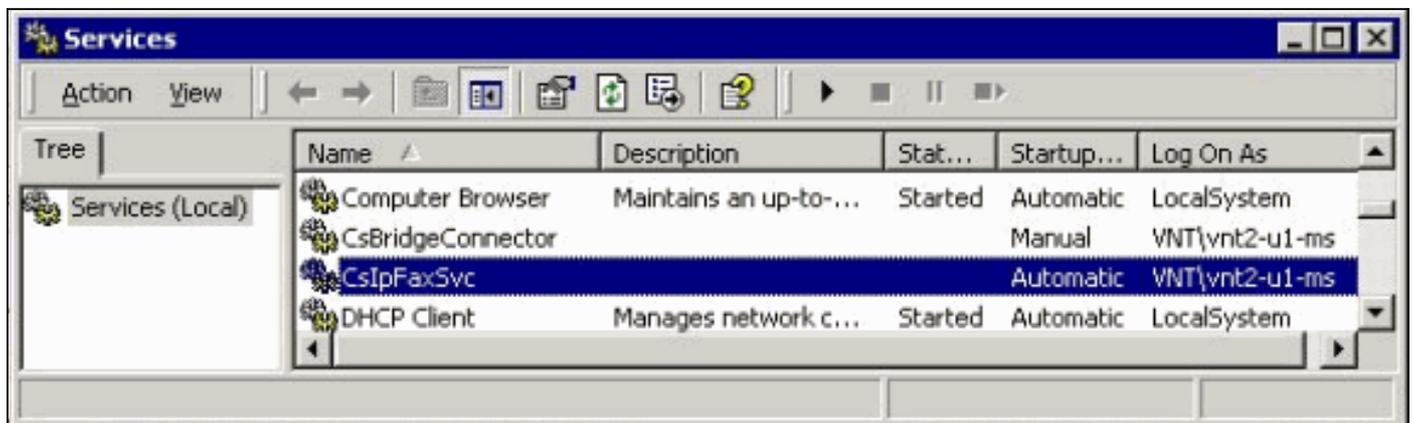
ODCDAEMON.log ファイルに最も有用な情報があります。レンダリングされる各ファイルについて、次のように表示されます。

```

10/14/04 08:27:17
Start time: Thursday, October 14, 2004 08:27:17 AM
Daemon logged in as user: vnt2-u1-ms
Using registry root: HKEY_LOCAL_MACHINE
08:27:17: Pipe name: PrintDocDaemonPipe
08:27:17: Default printer name: ImageMaker DocCnvr Driver
08:27:17: Registry server: <local>
08:27:17: Service started
08:27:19: 08:27:19: 10/14/04 08:27:19 ServiceRequest thread started
08:27:19: Processing 'Convert v1.0' request
08:27:19: Processing file: C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt
08:27:19: Output file: C:\CommServer\unityMta\IpFaxScratchpad\render.tif
08:27:19: Printer name: ImageMaker DocCnvr Driver
08:27:19: AppendTo filename set to C:\CommServer\unityMta\IpFaxScratchpad\render.tif
08:27:19: Input filename: C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt
08:27:19: Output filename: C:\CommServer\unityMta\IpFaxScratchpad\render.tif
08:27:19: 0xE90: <10/14/04 08:27:19> StatusThread timeout value for jobstart: 60000
08:27:21: AutoShutdown feature activated
08:27:21: Sending response 1 to client
08:27:21: 10/14/04 08:27:21 ServiceRequest thread ended

```

まず調査する要素は logged in as user です。これは、CsIpFaxSvc サービスがログインに使用するのと同じアカウントです。



このユーザはすべてのアプリケーションを起動し、それらをレンダリングするため、この点を認識することが重要です。次の要素は Input and output file names です。第 3 の要素は response で、レンダリングされるページ数を示します。この場合は 1 ページです。応答が負の場合はエラーです。エラーが存在する場合は、その直前の数行を調べる必要があります。多くの場合、エラーが発生するのは特定のファイルタイプに対してだけです。ODCDAEMON log ファイルには、すべてのレンダリングに対してエラーが発生したか、特定の拡張子タイプに対してのみ発生したかが示されています。

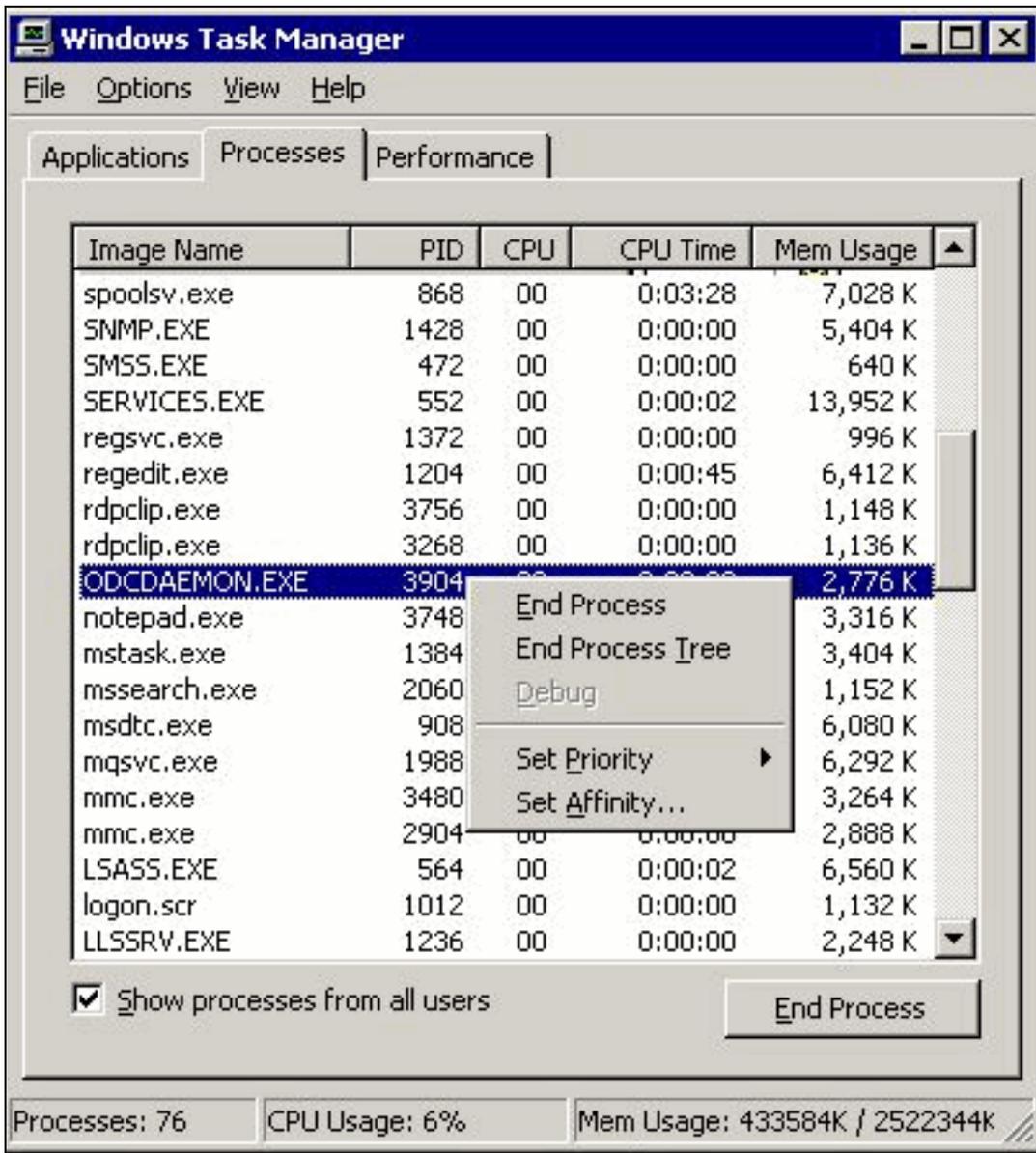
さらにこれをテストするためには、レンダリング エンジンを手動で実行することが有効です。CsIpFaxSvc サービスを実行するのと同じユーザでログインします。これを行うのは、レンダリング エンジンがサービスから起動された場合と同じエラーが発生するかどうかを確認するためです。

使用できる各拡張子タイプのファイルがある場合、それをダブルクリックすることで各ファイル

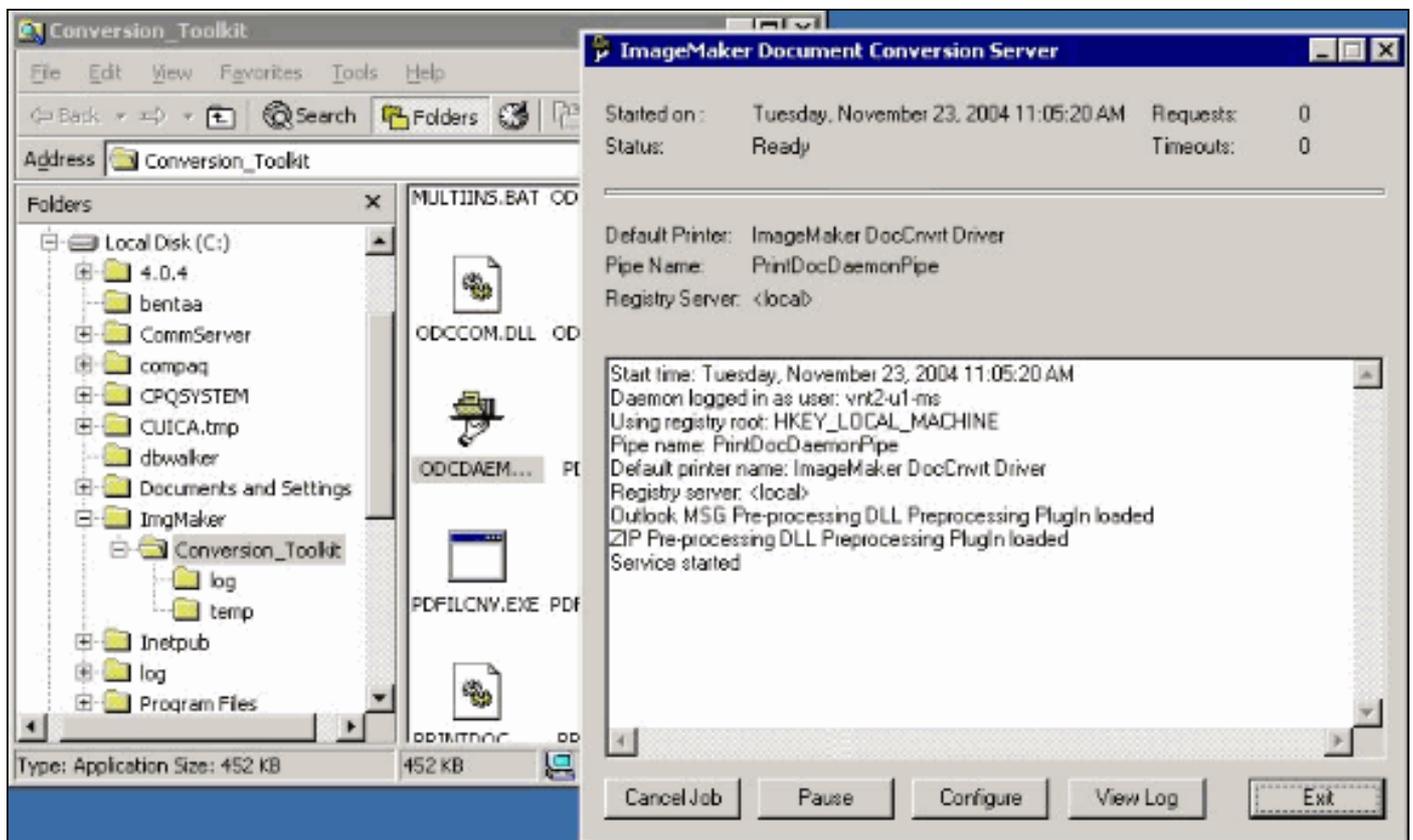
を開きます。関連付けられたアプリケーションが開き、更新をチェックします（または同様のプロセス）。たとえば Acrobat の場合は、この動作を無効にできます。

CslpFaxSvc を停止します。いずれかの ODCDAEMON プロセスがまだ動作中である場合は、タスク マネージャでそれを停止します。

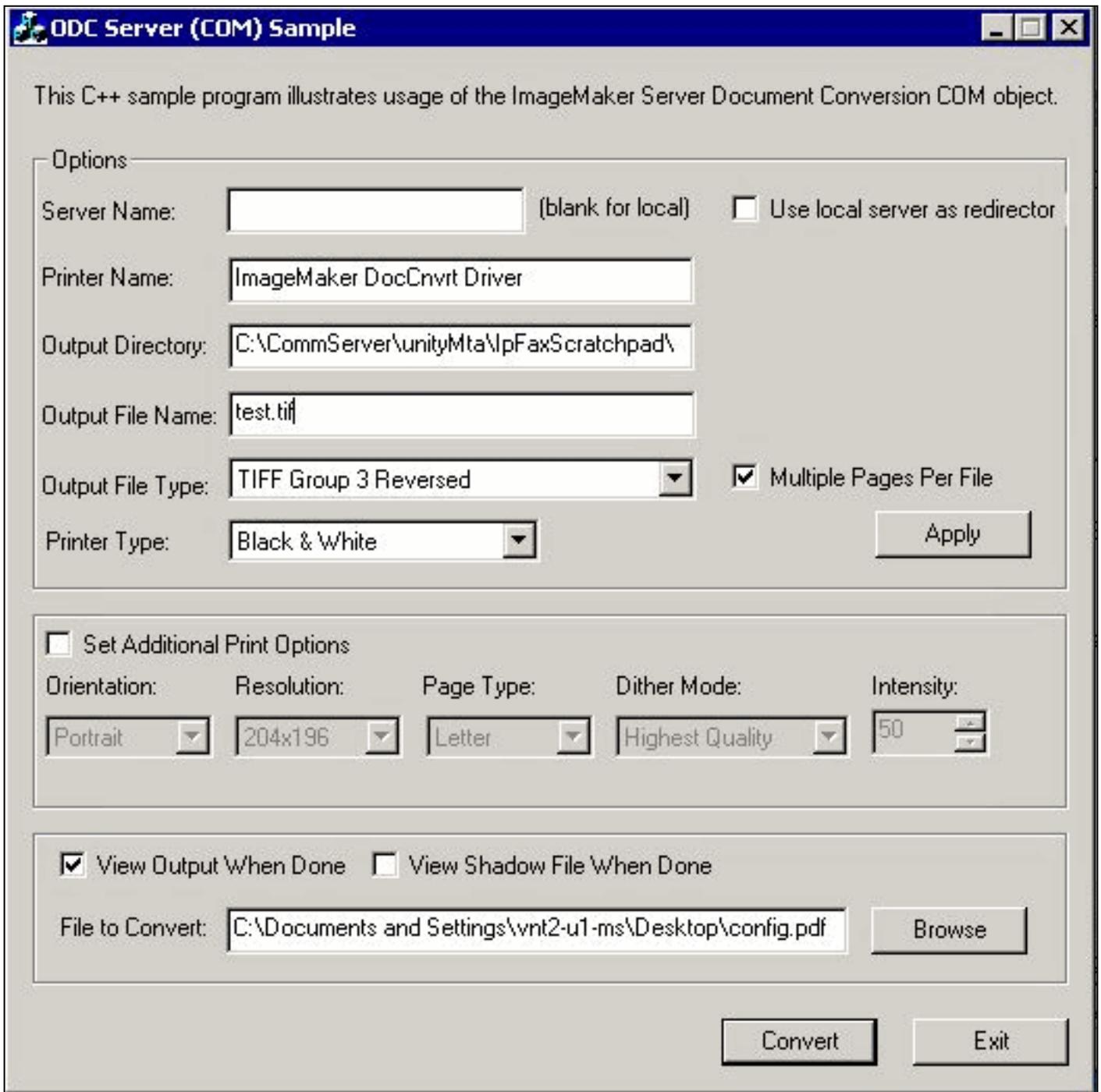
注： 今後参照するために ODCDAEMON ログ ファイルを保存する必要がある場合は、CslpFaxSvc に ping を実行する前に行う必要があります。サービスを再起動すると、ODCDAEMON ログ ファイルがクリアされ、出力が失われます。



C:\imgMaker\Conversion_Toolkit フォルダから ODCDAEMON.EXE プロセスを起動します。次の図のようにウィンドウが表示されます。



同じフォルダから CPCOM.EXE ファイルを起動します。



デフォルト設定の大部分に問題はありません。File to Convert フィールドでは目的の入力ファイルが選択されています。Cisco は、各拡張子タイプを別々にテストすることをお勧めします。出力ファイルが変更されていない場合、この出力ファイルに入力ファイルが追記されます。

[変換]をクリックします。アプリケーションが印刷を開始し、次のダイアログボックスが表示されます。



[OK] をクリックします。「View Output When Done」オプションを選択します。TIF ビューアが起動され、出力ファイルが表示されます。エラーがあると、場合によっては次の例のように、エラーはログ ファイル内の内容と同じものになります。

```
03:07:39: Input filename: d:\CommServer\UnityMTA\IpFaxScratchpad\render.tif
03:07:39: Output filename:
d:\CommServer\UnityMTA\IpFaxScratchpad\9484f347996b4ea18b0caeb52ffda465.tif
03:07:39: 0x15F0: <11/19/04 03:07:39> StatusThread timeout value for jobstart: 60000
PrintTo command does not exist for file type: MSPaper.Document
03:07:39: AutoShutdown feature activated
Conversion unsuccessful. Deleting file
d:\CommServer\UnityMTA\IpFaxScratchpad\9484f347996b4ea18b0caeb52ffda465.tif
03:07:39: Sending response -20 [PrintTo command does not exist] to client
```

入力ファイルは TIF 添付ファイルです。ただし、ファイル タイプは DOC (MSPaper.Document) です。この場合の問題は、Word や Excel などの Microsoft アプリケーションがサーバにインストールされ、レジストリのファイルの関連付けが変更されることです。このエラーを修正するには、HKEY_CLASSES_ROOT\.tif と TIF を編集し、Default key を MSPaper.Document ではなく TIFImage.Document に変更します。

各ファイル タイプが検証されたら、CPCOM と ODCDAEMON のウィンドウを閉じ、CslpFaxSvc サービスを再起動します。

すべての添付ファイルが失敗する、または初期化エラーが存在する場合は、レンダリング エンジンを再インストールします。IPFaxConfigWizard が存在する ImageMaker フォルダから instfax.exe -z コマンドを使用してアンインストールします。その後、再起動が必要な再インストールが行われ、[スタート] > [プログラム] > [Unity] から Cisco Unity IP Fax Configuration Wizard が再実行されま

す。
エラーをテストする最後の選択肢は、デーモン プロセスをバイパスし、コマンドラインからファイルをレンダリングする方法です。通常、以前に失敗している場合は、この方法でも失敗します。ただし、場合によっては追加情報を提供します。これは、PDFILCNV 実行プログラムの支援を使用して実行されます。このプログラムは、IPFaxConfigWizard インストール パスの ImageMaker サブディレクトリ内にあります。以下に、いくつかの例を示します。

```
PDFILCNV products.doc -Fproducts.tif
```

表示されているファイルを開けられることを確認します。

次の例は IpFaxSvc ログ ファイルの例です。

```
"CIpFaxCdoHelper::RenderCurrentMessage() processing attachment=[products.doc]"
```

```
"CIpFaxCdoHelper::RenderCurrentMessage() attachment written to file=  
[C:\CommServer\unityMta\IpFaxScratchpad\products.doc]"
```

```
"Rendering file [C:\CommServer\unityMta\IpFaxScratchpad\products.doc] to  
[C:\CommServer\unityMta\IpFaxScratchpad\render.tif]"
```

```
"CIpFaxCdoHelper::R\  
  
)ll  
\-  
\]  
'=e-nderCurrentMessage() added [2] pages to TIFF"
```

products.doc 添付ファイルがレンダリングされています。これが、以前に作成された render.tif ファイルに追記されます。これは 2 ページであるため、合計ページ数は 3 になります。

```
"CIpFaxCoverPage::GenerateFile() file=[C:\CommServer\unityMta\IpFaxScratchpad  
\cover.htm] name=[Unity2 Subscriber] num=[9915550] pages=[3]"
```

```
"Rendering file [C:\CommServer\unityMta\IpFaxScratchpad\cover.htm] to  
[C:\CommServer\unityMta\IpFaxScratchpad\0188c93ff8194177b9f313edabc2ea73.tif]"
```

```
"Rendering file [C:\CommServer\unityMta\IpFaxScratchpad\render.tif] to  
[C:\CommServer\unityMta\IpFaxScratchpad\0188c93ff8194177b9f313edabc2ea73.tif]"
```

カバー ページは cover.htm ファイルです。これはいくつかの変数を持つ単純な HTML テンプレートです。メッセージを送信する加入者の名前、呼び出される番号、および送信されるページ数を入力します。

カスタム cover.htm ファイルが作成されると、そのファイルが messageid.tif ファイルにレンダリングされます。以前に作成した render.tif ファイルは、たとえば messageid.tif ファイルに追加してレンダリングされます。この時点では、messageid.tif ファイルが、E メール送信またはファックス送信される実際のメッセージです。

ファイルは正しくレンダリングされています。SQL の一部のフィールドが更新され、メッセージが送信されます。

```
"2004-10-14", "08:27:29", "CIpFaxAdoHelper::Execute()  
Cmd=[UPDATE tblIpFaxData SET Status = '1' WHERE MessageEntryID =  
'00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C  
51200000005E5B20000E7E65E238D0DEF46B5F8EAF983C6C51200000005F55A0000']"
```

```
"2004-10-14", "08:27:29", "CIpFaxAdoHelper::Execute() Cmd=[update tblipfaxdata  
set TimeLastAttempted = '2004-10-14 08:27:29' where MessageEntryID =  
'00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C5120  
0000005E5B20000E7E65E238D0DEF46B5F8EAF983C6C51200000005F55A0000']";
```

メッセージのステータスは 1 に設定され、最後に更新された時刻が入力されます。ゲートウェイに送信されます。

[ステップ 4: ルータがメッセージを受信および配送することの確認](#)

IP Fax アカウントからゲートウェイに送信されるメッセージは、fax=####@router.gateway.com という形式である必要があります。一般的に、DNS には DNS A および MX レコードが必要で、メッセージを当該ホストにルーティングするために SMTP コネクタを Exchange に追加する必要

があります。この作業を行えば、ゲートウェイのトラブルシューティングに進むことができます

。

ルータだけを設定している場合でも、Telnetを使用してSMTPポートにアクセスし、次の例に示すように手動でメッセージを送信すれば、ルータが動作していることを確認できます。

```
telnet vnt-3725-51.gateway.com 25
Trying 14.80.51.14...
Connected to 14.80.51.14.
Escape character is '^]'.
220 vnt-3725-51.gateway.com Cisco NetWorks ESMTP server
ehlo
250-vnt-3725-51.gateway.com, hello <empty HELO/EHLO> [172.18.106.123] (really )
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-PIPELINING
250-HELP
250-DSN
250 XACCOUNTING
mail from:test@test.com
250 2.5.0 Sender <test@test.com> ok
rcpt to:fax=9915550@vnt-3725-51.gateway.com
250 2.1.5 Recipient <fax=9915550@vnt-3725-51.gateway.com> ok, maps to `9915550' (cp=no)
data
354 Enter mail, end with a single "."
This is a test message from the router.
.
554 5.3.0 An unknown error occurred
quit
221 2.3.0 Goodbye from vnt-3725-51.gateway.com; closing connection
Connection closed by foreign host.
```

この場合、未知のエラーがあります。このエラーが発生したのは、発信ファックスの配送に使用される Primary Rate Interface (PRI; 一次群速度インターフェイス) がダウンしているためです。次に別の例を示します。

```
telnet vnt-3725-51.gateway.com 25
Trying 14.80.51.14, 25 ... Open
220 vnt-3725-51.gateway.com Cisco NetWorks ESMTP server
ehlo
250-vnt-3725-51.gateway.com, hello <empty HELO/EHLO> [172.18.106.66] (really )
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-PIPELINING
250-HELP
250-DSN
250 XACCOUNTING
mail from:test@test.com
250 2.5.0 Sender <test@test.com> ok
rcpt to:fax=9915550@error.com
553 5.4.4 <fax=9915550@error.com> Unable to relay to remote host
quit
221 2.3.0 Goodbye from vnt-3725-51.gateway.com; closing connection
```

この場合、「Unable to relay to remote host」エラーメッセージは、Cisco IOS設定の`mta receive aliases error.com`が設定されていないことを意味します。

ゲートウェイの直接的なトラブルシューティングを始めるには、ルータに Telnet 接続するか、コ

ンソール経由でルータにアクセスします。Telnet で接続したら、terminal monitor コマンドを発行します。debug オプションが表示されます。debug fax mta all を有効にします。発信音声回線が PRI である場合、debug isdn q931 も関連コマンドになります。

警告： ルータ上でトレースを有効にすると、時としてパフォーマンスに影響が生じる場合があります。トレースが完了したら、no debug all コマンドを使用してトレースをオフにします。

注：これらのログは、スペースを節約するために切り捨てられています。

次の debug 出力は、アクティブな例からのものです。

```
Oct 15 08:27:26.003: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=1
Oct 15 08:27:35.051: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
Calling smtp verb: ehlo
Oct 15 08:27:35.051: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=1
Oct 15 08:27:35.599: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
Calling smtp verb: mail
Oct 15 08:27:35.599: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=1
Oct 15 08:27:36.143: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
Calling smtp verb: rcpt
Oct 15 08:27:36.143: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_command_rcpt:
context(0x66CA9AC4)
Oct 15 08:27:36.143: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_command_rcpt:
context(0x66CA9AC4)
Oct 15 08:27:36.143: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=1
Oct 15 08:27:36.691: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
Calling smtp verb: data
Oct 15 08:27:36.691: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=2
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Received: from vnt2-ex1.vnt.com ([14.84.31.12]) by VNT1-EX2.vnt.com
with Microsoft SMTPSVC(5.0.2195.6713);`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: ` Thu, 14 Oct 2004 08:27:29 -0400`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Return-Receipt-To: "fax unity2"`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Subject: 0000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B
5F8EAF983C6C51200000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C5120000005FB680000`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Date: Thu, 14 Oct 2004 08:27:29 -0400`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Message-ID: `
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `X-MS-Has-Attach: yes`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `MIME-Version: 1.0`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Content-Type: image/tiff;`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `name="BODY.TIF"`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `X-MS-TNEF-Correlator: `
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Content-Transfer-Encoding: base64`
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
```

(S)R: `Content-Description: BODY.TIF'
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Content-Disposition: attachment;'
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: ` **filename="BODY.TIF"** '
Oct 15 08:27:37.239: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Thread-Topic: 00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0
DEF46B5F8EAF983C6C51200000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C51200000005
FB680000 '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=2
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Thread-Index: AcSx6SvU9sc3nVffS3a4/Dwqi2Bpww== '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `**From: "fax unity2"** '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `content-class: urn:content-classes:message '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `X-MimeOLE: Produced By Microsoft Exchange V6.0.5762.3 '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `To: '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `Return-Path: fax-unity2@vnt.com '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: `X-OriginalArrivalTime: 14 Oct 2004 12:27:29.0487 (UTC) FILETIME=[2BE0DDF0:01C4B1E9] '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_work_routine:
(S)R: ``
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
**Field name: `subject: '
Field value: `00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F
8EAF983C6C512000000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C51200000005FB680000 '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
Field name: `date: '
Field value: `Thu, 14 Oct 2004 08:27:29 -0400 '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
Field name: `message-id: '
Field value: ``
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
Field name: `Content-Type: '
Field value: `image/tiff; name="BODY.TIF" '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
Field name: `Content-Transfer-Encoding: '
Field value: `base64 '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
**Field name: `from: '
Field value: `**"fax unity2"** '
Oct 15 08:27:37.243: //-1/xxxxxxxxxxxxx/SMTPS/esmtp_server_engine_new_part:
**Field name: `to: '
Field value: ``
Oct 15 08:27:37.247: //-1/E7B232638008/SMTPS/esmtp_server_engine_new_part:exit@1977
Oct 15 08:27:37.251: ISDN Se2/0:23 Q931: pak_private_number: Invalid type/plan
0x0 0x1 may be overridden; sw-type 13
Oct 15 08:27:37.251: ISDN Se2/0:23 Q931: Applying typeplan for sw-type 0xD is
0x0 0x0, Called num 99915550
Oct 15 08:27:37.255: ISDN Se2/0:23 Q931: TX -> SETUP pd = 8 callref = 0x0003
Bearer Capability i = 0x8090A2
Standard = CCITT
Transer Capability = Speech
Transfer Mode = Circuit
Transfer Rate = 64 kbit/s
Channel ID i = 0xA98397
Exclusive, Channel 23
Called Party Number i = 0x80, '99915550'
Plan:Unknown, Type:Unknown******

```

...
Oct 15 08:27:46.003: ISDN Se2/0:23 Q931: RX <- CONNECT pd = 8 callref = 0x8003
Oct 15 08:27:46.003: ISDN Se2/0:23 Q931: TX -> CONNECT_ACK pd = 8 callref = 0x0003
Oct 15 08:27:52.003: %ISDN-6-CONNECT: Interface Serial2/0:22 is now connected to
99915550
...
Oct 15 08:30:11.735: //-1/E7B232638008/SMTPS/esmtp_server_engine_buf_mgmt:
buffer offset=1368, state=3
Oct 15 08:30:11.735: //-1/E7B232638008/SMTPS/esmtp_server_engine_dot_check:
Not found the terminating Dot!
Oct 15 08:30:11.739: //-1/E7B232638008/SMTPS/esmtp_server_engine_buf_mgmt:
state=3
Oct 15 08:30:11.739: //-1/E7B232638008/SMTPS/esmtp_server_engine_buf_mgmt:
buffer offset=1412, state=3
Oct 15 08:30:11.739: //-1/E7B232638008/SMTPS/esmtp_server_engine_dot_check:
Found the terminating Dot!
Oct 15 08:30:11.739: //-1/E7B232638008/SMTPS/esmtp_server_engine_work_routine:
context(0x66CA9AC4), state=10Oct 15 08:30:12.483: //-1/xxxxxxxxxxxx/SMTPS/
esmtp_server_engine_update_recipient_status: faxmail status=6Oct 15 08:30:27.883:
//-1/xxxxxxxxxxxx/SMTPS/esmtp_server_engine_update_recipient_status:
faxmail status=7

```

ルータ上のメールサーバがメッセージを受信します。その番号に対してコールが行われるまで、SMTPカンパセーション内のすべてのフィールドが、ルータコードのさまざまなレベルで解析されます。注意すべき重要な要素を、いくつかの設定ステップに示します。

show running-config の出力では、次の内容が確認できます。

- **fax interface-type fax-mail**
- OffRamp ソフトウェアを、Flash 内または FTP/TFTP サーバ上にロードする必要があります。続いてアプリケーションを設定します。たとえば、**call application voice offramp flash:app_faxmail_offramp.2.0.1.1.tcl**。注：オンランプ機能に使用する Tool Command Language(TCL)ソフトウェアは、[Cisco Downloads](#)(登録ユーザ専用)からダウンロードできます。
- **mta receive maximum <recipients>**:0より大きい数値
- **mta receive aliases <string>**:DNSサフィックスと一致する必要があります。たとえば、Cisco Unity からの E メールが fax=912345@myrouter.com に送信される場合、エイリアスは myrouter.com である必要があります。mta receive aliases は複数使用できます。
- **mta send server <IP address of Exchange server>**:DSNを送信するには、ゲートウェイがメールサーバに接続してメッセージを配信する必要があります。設定する必要がある Exchange サーバは 1 つだけですが、複数の Exchange サーバを設定することもできます。すべての無効な mta send server アドレスが削除されていることを確認します。それ以外の場合、リスト内の最初のアドレスが常に使用されます。
- **mta receive generate permanent-error** : ルータがDSNをCisco Unityに正しく返送するために必要です。
- **ダイヤルピア設定例**。以下に、いくつかの例を示します。

```

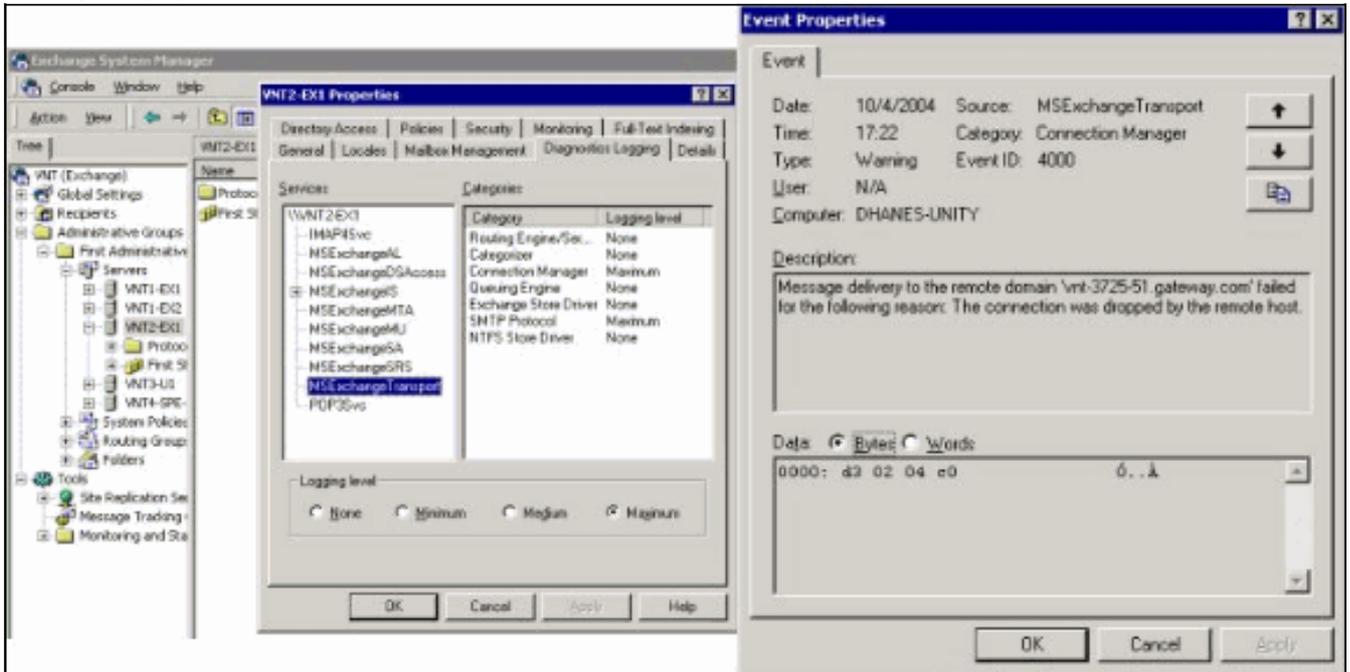
dial-peer voice 5590 pots
destination-pattern 991...
port 2/0:23
forward-digits all
prefix 9
!
dial-peer voice 2 mmoip
application offramp
information-type fax
incoming called-number 991
dsn delayed

```

dsn success
dsn failure

!

メッセージがゲートウェイに到達しない場合、Exchange をトラブルシューティングする必要があります。これには、Exchange 2000 内の診断が非常に有用です。以下に、いくつかの例を示します。



カテゴリ SMTP Protocol および Connection Manager の MSExchangeTransport 診断が有効です。これは、個別のサーバのプロパティをもとの Exchange System Manager です。メッセージを送信すると、Application Event Viewer のログにエラーが確認されます。最も一般的な設定ミスは、ルータの mta receive aliases パラメータが E メール サフィクスと一致しない場合に生じます。たとえば、ファックス サービスが fax=91234@router.abc.com にメッセージを送信しようとする場合、ゲートウェイでは mta receive aliases router.abc.com が設定されている必要があります。

ステップ 5: ルータが DSN メッセージを送信することの確認

生成される通知は次のとおりです。スペースを節約するため、トレースの部分は省略されています。

```
Oct 15 08:30:41.739: //-1/E7B232638008/SMTSPS/  
esmtplib_server_engine_generate_notification:  
context(0x66CA9AC4)  
Oct 15 08:30:41.739: //-1/E7B232638008/SMTSPS/sendDSN:  
context(0x66CA9AC4)  
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPC/esmtplib_client_engine_open:  
from=, to=fax-unity2@vnt.com  
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPC/esmtplib_client_engine_open:exit@1866  
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPC/esmtplib_client_engine_add_headers:  
from_comment=Fax Daemon  
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPC/esmtplib_client_engine_dispose_context:  
exit@2109  
Oct 15 08:30:41.739: //-1/E7B232638008/SMTSPS/sendDSN:  
DSN sent, context(0x66CA9AC4)  
Oct 15 08:30:41.739: //-1/E7B232638008/SMTSPS/  
esmtplib_server_engine_generate_notification:exit@35  
Oct 15 08:30:41.739: //-1/E7B232638008/SMTSPS/esmtplib_server_engine_work_routine:
```

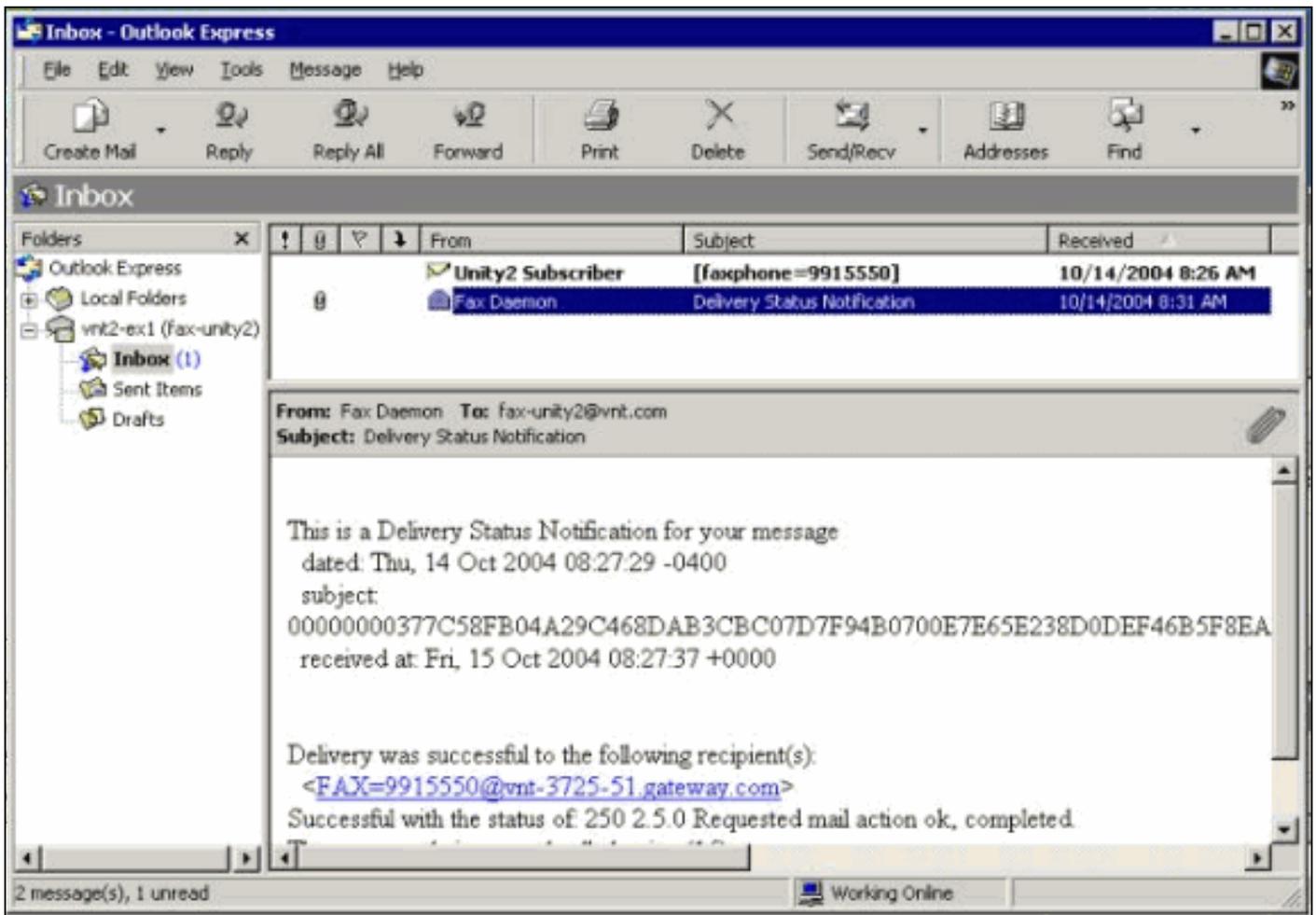
```
context(0x66CA9AC4), state=1
Oct 15 08:30:41.983: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_work_routine:
    socket 0 readable for first time
Oct 15 08:30:41.983: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_work_routine:
    socket 0 readable for first time
Oct 15 08:30:41.983: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 220 vnt2-ex1.vnt.com Microsoft ESMTP MAIL Service, Version: 5.0.2195.6713
    ready at Thu, 14 Oct 2004 08:31:09 -0400
Oct 15 08:30:41.987: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
    (C)S: EHLO vnt-3725-51.gateway.com
Oct 15 08:30:41.987: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
    return code=0
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-vnt2-ex1.vnt.com Hello [14.80.51.14]
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-TURN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-ATRN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-SIZE
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-ETRN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-PIPELINING
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-DSN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-ENHANCEDSTATUSCODES
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-8bitmime
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-BINARYMIME
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-CHUNKING
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-VRFY
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-X-EXPS GSSAPI NTLM LOGIN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-X-EXPS=LOGIN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-AUTH GSSAPI NTLM LOGIN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-AUTH=LOGIN
Oct 15 08:30:42.463: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-X-LINK2STATE
Oct 15 08:30:42.467: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250-XEXCH50
Oct 15 08:30:42.467: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250 OK
Oct 15 08:30:42.467: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
    (C)S: MAIL FROM:<>
Oct 15 08:30:42.467: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
    return code=0
Oct 15 08:30:43.011: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250 2.1.0 <>...Sender OK
Oct 15 08:30:43.011: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
    (C)S: RCPT TO:
Oct 15 08:30:43.011: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
    return code=0
Oct 15 08:30:43.555: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
    (C)R: 250 2.1.5 fax-unity2@vnt.com
Oct 15 08:30:43.555: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
    return code=0
```

```
Oct 15 08:30:44.103: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_getln:
(C)R: 354 Start mail input; end with .
Oct 15 08:30:44.103: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Received: by vnt-3725-51.gateway.com for (with Cisco NetWorks);
Fri, 15 Oct 2004 08:30:44 +0000
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: To:
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Message-ID: <00042004083044107@vnt-3725-51.gateway.com>
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Date: Fri, 15 Oct 2004 08:30:44 +0000
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Subject: Delivery Status Notification
...
```

正常に動作しない場合、設定の mta send server コマンドを、正しい Exchange サーバーとポート (デフォルトは 25) に設定する必要があります。送信時にファックス メッセージが照合される mmoip ダイアルピア上で DSN を設定する必要があります。mta receive generate permanent-error コマンドが存在します。時として、このコマンドが存在しないか、コマンドが正しくないメールサーバをポイントしていることがあります。

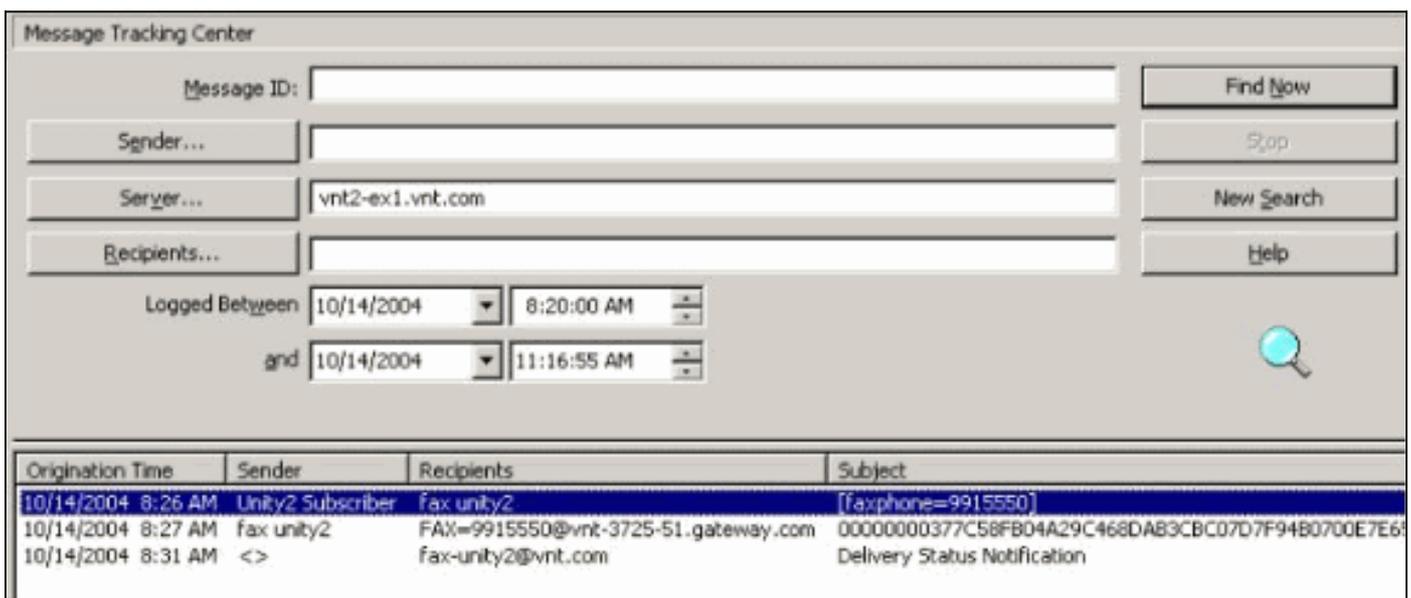
注 : Cisco IOS の設定では、複数の mta send server コマンドを設定することができます。ただし、デバイスが完全に到達不能にならない限り、最初のデバイスだけが使用されます。そのため、mta send server を変更した場合、以前のコマンドを削除する必要があります。

Outlook または Outlook Express があり、ユーザがファックス メールボックスを監視している場合、IP Fax サービスが次の間隔で処理を行うより前に DSN が配送されることが確認されます。



メッセージの本文にはメッセージ ID が含まれています。以前は「subject」でした。エラーは、メッセージの本文の最後の付近に記載されています。これらは ISDN Q.931 エラーコードのアブストラクションです。詳細は、「[付録](#)」を参照してください。

同じように有用なツールには、Exchange Message Tracking Center があります。このツールは Exchange 2000/2003 System Manager に組み込まれています。次のようなメールメッセージが示されます。



ステップ 6 : DSN メッセージが受信されたことの確認

DSNメッセージがIP Faxサービスのメールボックスにある場合は、メッセージが処理されていることを確認します。これには IpFaxSvc ログ ファイルが有用です。

```
"2004-10-14", "08:31:30", "Logging on to mailbox [VNT2-EX1\fax-unity2]"
"2004-10-14", "08:31:30", "CIpFaxCdoHelper::LogonToMailbox() name returned from
Logon is [fax unity2]" "2004-10-14", "08:31:30", "Processing outbox, message count= 2"
"2004-10-14", "08:31:30", "CIpFaxCdoHelper::ProcessFaxOutbox() processing message
receipts"
"2004-10-14", "08:31:30", "CIpFaxCdoHelper::DetermineMessageDisposition()
current message subject=[Delivery Status Notification]"
"2004-10-14", "08:31:30", "CIpFaxCdoHelper::MessageIsReceipt() processing message,
type=[IPM.Note]" "2004-10-14", "08:31:30", "CIpFaxCdoHelper::MessageIsReceipt()
processing message as DELIVERY STATUS NOTIFICATION"
```

メッセージは処理されています。

```
"2004-10-14", "08:31:30", "CIpFaxMessageBodyParser::GetOriginalSubject
returning subject=[00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF4
6B5F8EAF983C6C51200000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C51200000005FB680000]"
"2004-10-14", "08:31:30", "CIpFaxMessageBodyParser::GetCauseCode returning cause
code =[16]"
```

Original Subject は、メッセージの本文から解析されます。これにより、データベース内でメッセージを検索できます。原因コードも16として解析されます。これは通常のコールクリアです。これらの原因コードは Q.931 原因コードに対応します。原因コードの詳細については、「[Q.931 接続解除原因コード](#)」を参照してください。

エントリは安全にデータベースから削除され、Status 0 で再度追加されます。

```
"2004-10-14", "08:31:30", "CIpFaxAdoHelper::Execute() Cmd=[DELETE
FROM tblIpFaxData WHERE MessageEntryID = '00000000377C58FB04A29C468DAB3CBC07D
7F94B0700E7E65E238D0DEF46B5F8EAF983C6C51200000005E5B20000E7E65E238D0DEF46B5F8
EAF983C6C51200000005F55A0000']"
```

```
"2004-10-14", "08:31:30", "_RecordsetPtr::Open(Select * from tblIpFaxData where
MessageEntryID = '00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46
B5F8EAF983C6C51200000005E5B20000E7E65E238D0DEF46B5F8EAF983C6C51200000005F55A0000')
returned [0] rows - Code: 0x80004005. Method: CIpFaxAdoHelper::GetRecordSet.
[IpFaxAdoHelper.cpp::403]"
```

```
"2004-10-14", "08:31:30", "CIpFaxAdoHelper::Execute() Cmd=[INSERT INTO tblIpFaxData
(MessageEntryID, Status, ScratchDirectory, TifFileName, NdrErrorCode,
BusyCounter, RnaCounter, TimeFirstHandled) VALUES ('00000000377C58FB04A29C4
68DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C51200000005E5B20000E7E65E
238D0DEF46B5F8EAF983C6C51200000005F55A0000', '0', 'C:\CommServer\unityMta\
IpFaxScratchpad', '2fd36f2af94d4137835928894fb397eb.tif', '0', '0', '0',
'2004-10-14 08:31:30')]"
```

両方のメールメッセージが削除されています。発信ファックスメッセージをチェックしても、何もありません。

```
2004-10-14", "08:32:30", "Processing outbox, message count= 0"
```

警告

現在のテスト対象バージョンには、注意点がいくつか存在します。これらの問題に対処するより新しいリリースについては、[IP Fax Configuration Wizard](#) またはヘルプ ファイルを参照してくだ

さい。

- Cisco Bug ID [CSCef61241](#)(登録ユーザ専用):Outbound IP Fax can not render PDFs.
- Cisco Bug ID [CSCef61256](#)(登録ユーザ専用):Outbound IP Fax can't handle gateway DSNs with Exchange 5.5.
- Cisco Bug ID [CSCef96036](#) (登録ユーザ専用) :IP Faxサービスは、件名の形式が正しくない場合にNDRを送信する必要があります。
- Cisco Bug ID [CSCef96051](#)(登録ユーザ専用):Outbound IP Fax service cover page should allow a callback number.
- Cisco Bug ID [CSCeg04158](#)(登録ユーザ専用):IP Faxサービスは、不明な添付ファイルを含むメッセージをNDRに送信する必要があります。
- Cisco Bug ID [CSCeg04180](#)(登録ユーザ専用):IP Faxサービスがサードパーティ製ファックスツールの最後の内線番号を認識しません。
- Cisco Bug ID [CSCeg39033](#)(登録ユーザ専用):IP Faxサービスでは、電子メールメッセージの本文を表示できないようにする必要があります。
- Cisco Bug ID [CSCeg39046](#)(登録ユーザ専用):IP Faxレンダリングエンジンとログディレクトリは常にC:ドライブにします。
- Cisco Bug ID [CSCeg39100](#)(登録ユーザ専用):IP Faxサービスは、複数のレンダリング障害の後にメッセージをNDRに送信する必要があります。
- Cisco Bug ID [CSCeg45439](#)(登録ユーザ専用):IP Faxサービスで送信者へのファックス配信の確認が可能です。
- Cisco Bug ID [CSCeg45666](#) (登録ユーザ専用) :IP Faxサービスで起動時にアプリケーションの依存関係を確認する必要があります。
- Cisco Bug ID [CSCeg46458](#) (登録ユーザ専用) :IP Faxサービスの停止によりODCDAEMONプロセスが強制終了されます。
- Cisco Bug ID [CSCeg52736](#)(登録ユーザ専用) : 発信IP Faxサービスでは、MaximumWaitキーが機能しません。

付録

レンダリングエンジントレースファイル

このドキュメントで説明した例のトレースを次に示します。ファイルは2つあり、一方のファイルはデーモン用で、もう一方のファイルはプリントドライバ用です。通常、これらは必要ありません。ただし、各サンプルは比較のために含まれています。レンダリングエンジンに関する問題を送信する場合は、通常のIP Faxのログに加えてこれらのファイルが必要です。

ODCDAEMON.log

C:\ImgMaker\Conversion_Toolkit\log\ODCDAEMON.log

このファイルは頻繁に使用され、また非常に読みやすいファイルです。入力および出力ファイルが独立したセクションで一覧表示され、また応答コードが一覧表示されています。応答コードが正であれば、それは出力TIFファイルにレンダリングされたページ数を示しています。負の応答コードはエラーを示しています。as userで示されているプロセス実行ユーザをチェックしてください。このユーザは個別のアプリケーションを開いてそれらをTIFに変換できる必要があるユーザであるため、これは非常に重要です。

10/14/04 08:27:17

Start time: Thursday, October 14, 2004 08:27:17 AM

Daemon logged in as user: vnt2-u1-ms

Using registry root: HKEY_LOCAL_MACHINE

08:27:17: Pipe name: PrintDocDaemonPipe

08:27:17: Default printer name: ImageMaker DocCnvrt Driver

08:27:17: Registry server: <local>

08:27:17: Service started

08:27:19:

08:27:19: 10/14/04 08:27:19 ServiceRequest thread started

08:27:19: Processing 'Convert v1.0' request

08:27:19: **Processing file: C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt**

08:27:19: **Output file: C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:19: Printer name: ImageMaker DocCnvrt Driver

08:27:19: **AppendTo filename set to C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:19: **Input filename: C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt**

08:27:19: **Output filename: C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:19: 0xE90: <10/14/04 08:27:19> StatusThread timeout value for jobstart: 60000

08:27:21: AutoShutdown feature activated

08:27:21: **Sending response 1 to client**

08:27:21: 10/14/04 08:27:21 ServiceRequest thread ended

08:27:21:

08:27:21: 10/14/04 08:27:21 ServiceRequest thread started

08:27:21: Processing 'Convert v1.0' request

08:27:21: **Processing file: C:\CommServer\unityMta\IpFaxScratchpad\products.doc**

08:27:21: **Output file: C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:21: Printer name: ImageMaker DocCnvrt Driver

08:27:21: **AppendTo filename set to C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:21: **Input filename: C:\CommServer\unityMta\IpFaxScratchpad\products.doc**

08:27:21: **Output filename: C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:21: 0xEB8: <10/14/04 08:27:21> StatusThread timeout value for jobstart: 60000

08:27:23: AutoShutdown feature activated

08:27:23: **Sending response 2 to client**

08:27:23: 10/14/04 08:27:23 ServiceRequest thread ended

08:27:23:

08:27:23: 10/14/04 08:27:23 ServiceRequest thread started

08:27:23: Processing 'Convert v1.0' request

08:27:23: **Processing file: C:\CommServer\unityMta\IpFaxScratchpad\cover.htm**

08:27:23: **Output file: C:\CommServer\unityMta\IpFaxScratchpad\
0188c93ff8194177b9f313edabc2ea73.tif**

08:27:23: Printer name: ImageMaker DocCnvrt Driver

08:27:23: **AppendTo filename set to C:\CommServer\unityMta\
IpFaxScratchpad\0188c93ff8194177b9f313edabc2ea73.tif**

08:27:23: **Input filename: C:\CommServer\unityMta\IpFaxScratchpad\cover.htm**

08:27:23: **Output filename: C:\CommServer\unityMta\IpFaxScratchpad\
0188c93ff8194177b9f313edabc2ea73.tif**

08:27:23: 0xED8: <10/14/04 08:27:23> StatusThread timeout value for jobstart: 60000

08:27:24: AutoClose skipping window: <Print> with class: <#32770> because parent is IE6

08:27:24: AutoClose skipping window: <Print> with class: <#32770> because parent is IE6

08:27:26: AutoShutdown feature activated

08:27:26: **Sending response 1 to client**

08:27:26: 10/14/04 08:27:26 ServiceRequest thread ended

08:27:26:

08:27:26: 10/14/04 08:27:26 ServiceRequest thread started

08:27:26: Processing 'Convert v1.0' request

08:27:26: Processing file: C:\CommServer\unityMta\IpFaxScratchpad\render.tif

08:27:26: **Output file: C:\CommServer\unityMta\IpFaxScratchpad\
0188c93ff8194177b9f313edabc2ea73.tif**

08:27:26: Printer name: ImageMaker DocCnvrt Driver

08:27:26: **AppendTo filename set to C:\CommServer\unityMta\IpFaxScratchpad\
0188c93ff8194177b9f313edabc2ea73.tif**

08:27:26: **Input filename: C:\CommServer\unityMta\IpFaxScratchpad\render.tif**

08:27:26: **Output filename: C:\CommServer\unityMta\IpFaxScratchpad\
0188c93ff8194177b9f313edabc2ea73.tif**
08:27:26: 0xEDC: <10/14/04 08:27:26> StatusThread timeout value for jobstart: 60000
08:27:29: AutoShutdown feature activated
08:27:29: **Sending response 3 to client**
08:27:29: 10/14/04 08:27:29 ServiceRequest thread ended

ドライバログ

C:\log\ImageMaker DocCnvrt Driver.log

このトレースが必要になるのは、レンダリング エンジンに関する問題を Cisco テクニカルサポートに送信する場合です。

```
----->
Log session started: 04-10-14, 08:27:19 AM
Platform: WinNT 4.0
Release Compilation
pszPrinterName ImageMaker DocCnvrt Driver
pszAnsiFileName C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt
pszOutFileName C:\CommServer\unityMta\IpFaxScratchpad\render.tif
FileType:0 multiPages:1 command:2 timeout:1800

Entering pdcf_WriteInfoForControlDialog()
opening file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrt Driver.FWD"
to transfer data (printdoc.dll -> printdoc.exe dialog)
data flushed!
MS: 18808737
deleted file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrt Driver.BWD"
used to transfer data from dialog back to here!
MS: 18808737

Leaving pdcf_WriteInfoForControlDialog()

verb:[printto] params:["ImageMaker DocCnvrt Driver" "winspool" "Ne00:"]
lpFile:[C:\CommServer\unityMta\IpFaxScratchpad\MsgBody.txt]

Calling ShellExecuteEx()
MS: 18808737
ShellExecute() succeeded

Calling pdcf_WaitForDoneFromControlDialog()
MS: 18808769

Entering pdcf_WaitForDoneFromControlDialog()
Monitoring cancel event: ImageMaker DocCnvrt Driver.printdoc.cancel
Event already exists [this is good]
  fopen() failed
  fread() failed
!DONE: timeout = 1799
reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrt Driver.BWD"
to get 'done' signal from dialog
MS: 18809776
ppd_trans_feedback.done = TRUE

Calling pdcf_WriteAckToControlDialog()
MS: 18809776

Entering pdcf_WriteAckToControlDialog
flushing 'done ack' signal to dialog
MS: 18809776
```

Have to wait

Leaving pdcf_WriteAckToControlDialog

Leaving pdcf_WaitForDoneFromControlDialog()

pdcf_WaitForDoneFromControlDialog() returned
MS: 18810425

Entering pdcf_CleanUpControlDialog()

MS: 18810425

Log session stopped: 04-10-14, 08:27:21 AM

----->

----->

Log session started: 04-10-14, 08:27:21 AM

Platform: WinNT 4.0

Release Compilation

pszPrinterName ImageMaker DocCnvrtdriver

pszAnsiFileName C:\CommServer\unityMta\IpFaxScratchpad\products.doc

pszOutFileName C:\CommServer\unityMta\IpFaxScratchpad\render.tif

FileType:0 multiPages:1 command:2 timeout:1800

Entering pdcf_WriteInfoForControlDialog()

opening file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.FWD"

to transfer data (printdoc.dll -> printdoc.exe dialog)

data flushed!

MS: 18810518

deleted file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"

used to transfer data from dialog back to here!

MS: 18810518

Leaving pdcf_WriteInfoForControlDialog()

verb:[printto] params:["ImageMaker DocCnvrtdriver" "winspool" "Ne00:"]

lpFile:[C:\CommServer\unityMta\IpFaxScratchpad\products.doc]

Calling ShellExecuteEx()

MS: 18810518

ShellExecuteEx() succeeded

Calling pdcf_WaitForDoneFromControlDialog()

MS: 18810561

Entering pdcf_WaitForDoneFromControlDialog()

Monitoring cancel event: ImageMaker DocCnvrtdriver.printdoc.cancel

Event already exists [this is good]

fopen() failed

fread() failed

!DONE: timeout = 1799

reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"

to get 'done' signal from dialog

MS: 18811581

!DONE: timeout = 1798

reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"

to get 'done' signal from dialog

MS: 18812585

ppd_trans_feedback.done = TRUE

Calling pdcf_WriteAckToControlDialog()

MS: 18812585

Entering pdcf_WriteAckToControlDialog
flushing 'done ack' signal to dialog
MS: 18812585
Have to wait

Leaving pdcf_WriteAckToControlDialog

Leaving pdcf_WaitForDoneFromControlDialog()

pdcf_WaitForDoneFromControlDialog() returned
MS: 18812858

Entering pdcf_CleanUpControlDialog()
MS: 18812858
Log session stopped: 04-10-14, 08:27:23 AM

----->

----->

Log session started: 04-10-14, 08:27:23 AM

Platform: WinNT 4.0

Release Compilation

pszPrinterName ImageMaker DocCnvrtdriver

pszAnsiFileName C:\CommServer\unityMta\IpFaxScratchpad\cover.htm

pszOutFileName C:\CommServer\unityMta\IpFaxScratchpad

0188c93ff8194177b9f313edabc2ea73.tif

FileType:0 multiPages:1 command:2 timeout:1800

Entering pdcf_WriteInfoForControlDialog()
opening file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.FWD"
to transfer data (printdoc.dll -> printdoc.exe dialog)
data flushed!

MS: 18812890

deleted file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"
used to transfer data from dialog back to here!

MS: 18812890

Leaving pdcf_WriteInfoForControlDialog()

verb:[printto] params:["ImageMaker DocCnvrtdriver" "winspool" "Ne00:"]
lpFile:[C:\CommServer\unityMta\IpFaxScratchpad\cover.htm]

Calling ShellExecuteEx()

MS: 18812894

ShellExecuteEx() succeeded

Calling pdcf_WaitForDoneFromControlDialog()

MS: 18812897

Entering pdcf_WaitForDoneFromControlDialog()
Monitoring cancel event: ImageMaker DocCnvrtdriver.printdoc.cancel
Event already exists [this is good]

fopen() failed

fread() failed

!DONE: timeout = 1799

fopen() failed

fread() failed

!DONE: timeout = 1798

reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"
to get 'done' signal from dialog

MS: 18814909

ppd_trans_feedback.done = TRUE

Calling pdcf_WriteAckToControlDialog()

MS: 18814909

Entering pdcf_WriteAckToControlDialog
flushing 'done ack' signal to dialog

MS: 18814909

Have to wait

Leaving pdcf_WriteAckToControlDialog

Leaving pdcf_WaitForDoneFromControlDialog()

pdcf_WaitForDoneFromControlDialog() returned

MS: 18815620

Entering pdcf_CleanUpControlDialog()

MS: 18815620

Log session stopped: 04-10-14, 08:27:26 AM

----->

----->

Log session started: 04-10-14, 08:27:26 AM

Platform: WinNT 4.0

Release Compilation

pszPrinterName ImageMaker DocCnvrtdriver

pszAnsiFileName C:\CommServer\unityMta\IpFaxScratchpad\render.tif

pszOutFileName C:\CommServer\unityMta\IpFaxScratchpad

0188c93ff8194177b9f313edabc2ea73.tif

FileType:0 multiPages:1 command:2 timeout:1800

Entering pdcf_WriteInfoForControlDialog()

opening file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.FWD"
to transfer data (printdoc.dll -> printdoc.exe dialog)

data flushed!

MS: 18815632

deleted file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"

used to transfer data from dialog back to here!

MS: 18815632

Leaving pdcf_WriteInfoForControlDialog()

verb:[printto] params:["ImageMaker DocCnvrtdriver" "winspool" "Ne00:"]

lpFile:[C:\CommServer\unityMta\IpFaxScratchpad\render.tif]

Calling ShellExecuteEx()

MS: 18815632

ShellExecuteEx() succeeded

Calling pdcf_WaitForDoneFromControlDialog()

MS: 18815659

Entering pdcf_WaitForDoneFromControlDialog()

Monitoring cancel event: ImageMaker DocCnvrtdriver.printdoc.cancel

Event already exists [this is good]

fopen() failed

fread() failed

!DONE: timeout = 1799

reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"
to get 'done' signal from dialog

MS: 18816667

!DONE: timeout = 1798

reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrtdriver.BWD"
to get 'done' signal from dialog

MS: 18817671

```

ppd_trans_feedback.done = TRUE

Calling pdcf_WriteAckToControlDialog()
MS: 18817671

Entering pdcf_WriteAckToControlDialog
flushing 'done ack' signal to dialog
MS: 18817671
Have to wait

Leaving pdcf_WriteAckToControlDialog

Leaving pdcf_WaitForDoneFromControlDialog()

pdcf_WaitForDoneFromControlDialog() returned
MS: 18818319

Entering pdcf_CleanUpControlDialog()
MS: 18818319
Log session stopped: 04-10-14, 08:27:29 AM
----->

```

Q.931 接続解除原因コード

Code	Cause
0	Valid cause code not yet received
1	Unallocated (unassigned) number
2	No route to specified transit network (WAN)
3	No route to destination
4	send special information tone
5	misdialed trunk prefix.
6	Channel unacceptable
7	Call awarded and being delivered in an established channel
8	Prefix 0 dialed but not allowed
9	Prefix 1 dialed but not allowed
10	Prefix 1 dialed but not required
11	More digits received than allowed, call is proceeding
16	Normal call clearing
17	User busy
18	No user responding
19	no answer from user
21	Call rejected
22	Number changed
23	Reverse charging rejected
24	Call suspended
25	Call resumed
26	Non-selected user clearing
27	Destination out of order
28	Invalid number format (incomplete number)
29	Facility rejected
30	Response to STATUS ENQUIRY
31	Normal, unspecified
33	Circuit out of order
34	No circuit/channel available
35	Destination unattainable
37	Degraded service
38	Network (WAN) out of order
39	Transit delay range cannot be achieved
40	Throughput range cannot be achieved
41	Temporary failure
42	Switching equipment congestion

43 Access information discarded
44 Requested circuit channel not available
45 Pre-empted
46 Precedence call blocked
47 Resource unavailable - unspecified
49 Quality of service unavailable
50 Requested facility not subscribed
51 Reverse charging not allowed
52 Outgoing calls barred
53 Outgoing calls barred within CUG
54 Incoming calls barred
55 Incoming calls barred within CUG
56 Call waiting not subscribed
57 Bearer capability not authorized
58 Bearer capability not presently available
63 Service or option not available, unspecified
65 Bearer service not implemented
66 Channel type not implemented
67 Transit network selection not implemented
68 Message not implemented
69 Requested facility not implemented
70 Only restricted digital information bearer capability
is available
79 Service or option not implemented, unspecified
81 Invalid call reference value
82 Identified channel does not exist
83 A suspended call exists, but this call identity does not
84 Call identity in use
85 No call suspended
86 Call having the requested call identity has been cleared
87 Called user not member of CUG
88 Incompatible destination
89 Non-existent abbreviated address entry
90 Destination address missing, and direct call not subscribed
91 Invalid transit network selection (national use)
92 Invalid facility parameter
93 Mandatory information element is missing
95 Invalid message, unspecified
96 Mandatory information element is missing
97 Message type non-existent or not implemented
98 Message not compatible with call state or message type
non-existent or not implemented
99 information element non-existent or not implemented
100 Invalid information element contents
101 Message not compatible with call state
102 Recovery on timer expiry
103 parameter non-existent or not implemented - passed on
111 Protocol error unspecified
127 Internetworking, unspecified

[関連情報](#)

- [音声に関する技術サポート](#)
- [音声とユニファイド コミュニケーションに関する製品サポート](#)
- [Cisco IP Telephony のトラブルシューティング](#)
- [テクニカル サポートとドキュメント - Cisco Systems](#)