

# Unity IP 팩스 아웃바운드 문제 해결

## 목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[표기 규칙](#)

[배경 정보](#)

[문제 해결 전](#)

[CslpFaxSvc 서비스가 설치, 구성 및 실행되는지 확인합니다.](#)

[팩스 계정을 모니터링하도록 메일 클라이언트 설정](#)

[레지스트리 설정](#)

[각 구성 요소 문제 해결](#)

[1단계: 최종 사용자가 팩스 계정으로 메시지를 보낼 수 있는지 확인합니다.](#)

[2단계: IP 팩스 서비스에서 메시지를 처리했는지 확인](#)

[3단계: 렌더링 작업 확인](#)

[4단계: 라우터가 메시지를 수신하고 전달하는지 확인](#)

[5단계: 라우터가 DSN 메시지를 전송하는지 확인](#)

[6단계: DSN 메시지가 수신되었는지 확인합니다.](#)

[주의 사항](#)

[부록](#)

[렌더링 엔진 추적 파일](#)

[Q.931 연결 해제 원인 코드](#)

[관련 정보](#)

## 소개

Cisco Unity 사용자에서 PSTN(Public Switched Telephone Network)의 팩스 장치로 보내는 아웃바운드 팩스는 Cisco Unity 4.0(4) 및 Cisco IOS® Software Release 12.3(7)T로 시작하는 Cisco Unity [IP Fax Configuration Wizard](#)를 통해 지원됩니다. 이 서비스를 통해 사용자는 간단한 전자 메일을 통해 아웃바운드 팩스를 보낼 수 있습니다. 대상 팩스 전화 번호는 이메일의 제목 줄에 포함되며 Cisco Unity IP Fax 서비스에서 확인하는 사서함으로 전송됩니다. 이 서비스는 메시지를 다시 포맷하고, 모든 첨부 파일을 적절한 형식으로 렌더링하고, 주소를 재지정하고, 메시지를 전송합니다. 이 메시지는 SMTP(Simple Mail Transfer Protocol)를 통해 T.37 OffRamp 기능으로 구성된 Cisco IOS 게이트웨이로 전송되어 대상 전화 번호로 팩스를 전송합니다.

이 문서에서는 트러블슈팅에 사용하는 일반적인 단계에 대해 설명합니다. Cisco IOS 게이트웨이의 초기 컨피그레이션 단계는 [Cisco Unity IP Fax Outbound Gateway 컨피그레이션 예](#)를 참조하십시오. Cisco Unity의 [IP 팩스 구성 마법사](#)를 참조하십시오.

## 사전 요구 사항

## 요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Cisco Unity IP Fax 아웃바운드 프로세스에 대한 일반 문서 [IP 팩스 컨피그레이션 마법사](#) 및 [Cisco Unity IP Fax 아웃바운드 게이트웨이 컨피그레이션 예를 참조하십시오](#).
- Microsoft Exchange 2000/2003을 통한 SMTP에 대한 기본 지식
- Cisco IOS Voice over IP(VoIP) 컨피그레이션

Cisco Unity IP Fax Configuration Wizard 버전 2.0.0.19 이상이 포함된 활성 Cisco Unity 서버는 이 문서에 정의된 게이트웨이 컨피그레이션과 함께 사용되어 완벽한 운영 체제를 갖추었습니다. IOS 게이트웨이 측면에서는 T.37 OffRamp 기능 및 Cisco IOS Software 릴리스 12.3(7)T 이상을 지원하는 Cisco IOS 라우터가 파트너 메시지 저장소로 Cisco Unity 버전 4.0(4) 이상과 함께 필요합니다.

**참고:** T.37 OffRamp는 MGCP(Media Gateway Control Protocol) 네트워크에서 지원되지 않습니다. T.37을 사용할 때 플랫폼 및 기타 제한 사항에 대한 자세한 내용은 [T.37 Store and Forward Fax 구성 설명서](#)를 참조하십시오.

## 사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- IOS 게이트웨이로 사용되는 Cisco 3725
- Cisco IOS Software 릴리스 12.3(8)T4

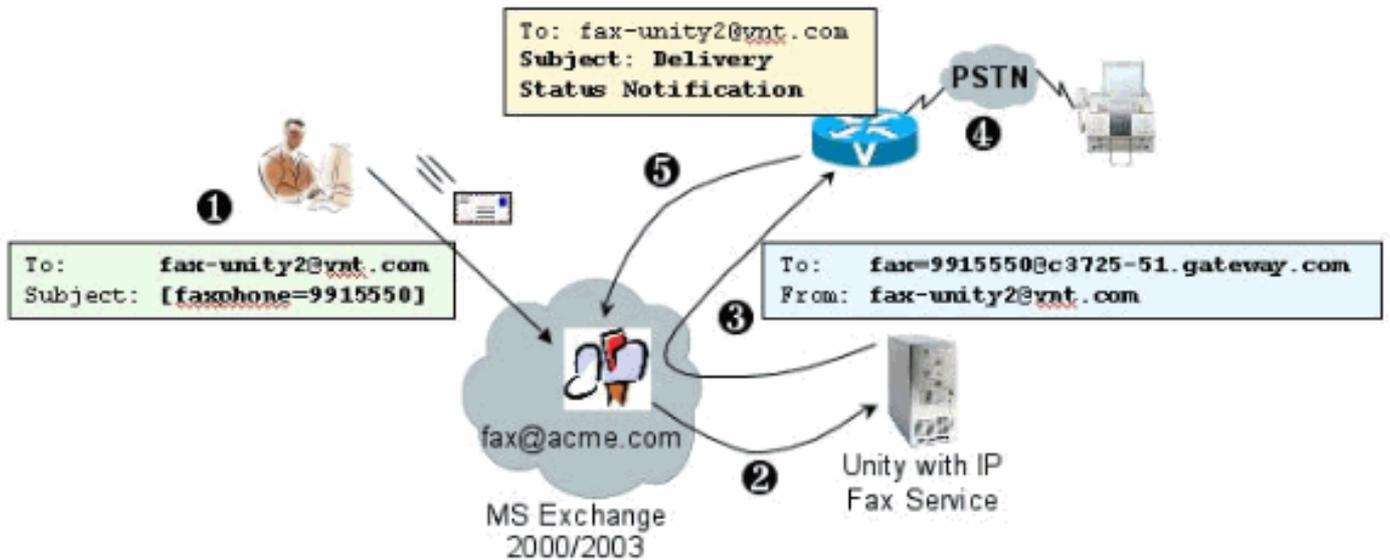
이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

## 표기 규칙

문서 규칙에 대한 자세한 내용은 [Cisco 기술 팁 표기 규칙을 참고하십시오](#).

## 배경 정보

트러블슈팅 절차를 설명하기 위해서는 이 기능의 기본 작동을 이해하는 것이 중요합니다. 그런 다음 문제가 특정 구성 요소로 좁혀지고 이에 대해 더 많은 정보가 수집됩니다.



1. 시스템의 최종 사용자가 팩스 사서함으로 전자 메일을 보냅니다. 이 메시지는 팩스 사서함(예: fax-unity2@vnt.com)의 제목에 [FAX=9915550]을 포함해야 합니다. 여기서 9915550은 대상 팩스 컴퓨터의 번호입니다. 게이트웨이에 설정된 다이얼 플랜에 따라 이 번호가 앞에 9가 있거나 특정 숫자 문자열만 허용됩니다. 메시지를 전송하면 Microsoft Exchange에서 이 사서함에 메시지를 전달합니다.참고: 전자 메일이 다른 도메인에서 시작된 경우 팩스 사서함이 있는 도메인 및 메일 계정으로 메일을 배달하도록 DNS(Domain Name System) 및 원격 메일 시스템이 구성됩니다.
2. Cisco Unity IP Fax 서비스는 정기적으로 팩스 사서함에 로그인하여 메시지를 검색합니다. 이제 첨부 파일을 읽습니다. 이 메시지는 Cisco Unity 서버에서 해당 크기의 첨부 파일을 처리할 수 있다는 가정 하에 TIF 파일로 변환된 상태로 렌더링됩니다.
3. 첨부 파일이 렌더링되면 메시지가 IOS 게이트웨이로 전송됩니다. 메시지를 보낼 수 있도록 Microsoft Exchange를 구성해야 합니다. 이 메시지는 수신자에게 보내집니다(예: fax=9915550@c3725-51.gateway.com). 이는 일반적으로 해당 호스트(c3725-51.gateway.com)에 대한 A 레코드 및 MX 레코드(메일 교환 레코드)가 DNS에 추가되고 메시지를 라우팅하기 위해 SMTP 커넥터가 Exchange에 추가됨을 의미합니다. 엔드 유저가 게이트웨이로 직접 이메일을 보내지 않으므로 여기에 사용된 이메일 주소와 호스트 이름은 사용자 친화적이지 않아도 됩니다.
4. IOS 게이트웨이는 T.37 OffRamp 기능에 대해 구성되어 있습니다. 이 기능을 사용하면 SMTP 메시지를 수신하고, fax=991550 주소를 해석하고, 991550 번호로 전화를 걸고(가능한 숫자 조작 후), TIF 파일을 수신자 팩스 장치에 전송할 수 있습니다.참고: 게이트웨이에서 MGCP를 사용하여 이 기능을 사용할 수 없습니다. 그러나 회로는 팩스에만 사용할 필요가 없습니다. 음성 및 팩스는 동일한 음성 회로에서 함께 사용됩니다. 라우터의 Dial-Peer 컨피그레이션에 따라 실제로 통화에서 수행되는 작업이 결정됩니다.
5. 팩스가 배달되면 게이트웨이가 팩스 사서함(fax-unity2@vnt.com)으로 DSN(배달 상태 알림) 메시지를 다시 보냅니다. 팩스가 성공적으로 전달되었는지 여부를 나타냅니다. IP 팩스 서비스가 로그인하면 DSN 메시지가 검색됩니다. 성공적으로 전달되지 않은 경우 원인 코드가 검사됩니다. 이 원인 코드에 따라 메시지가 재전송되거나(원격 팩스 사용 중) 최종 사용자에게 메시지를 보내 사용자에게 문제를 알립니다. 이렇게 하지 않으면 IP 팩스 서비스는 팩스가 성공적으로 전달되었는지 알 수 없습니다.

이 아키텍처의 장점 중 하나는 모든 문제가 일반적으로 관련 구성 요소로 분리된다는 것입니다.

## 문제 해결 전

## CslpFaxSvc 서비스가 설치, 구성 및 실행되는지 확인합니다.

시작하기 전에 설정을 확인하십시오.

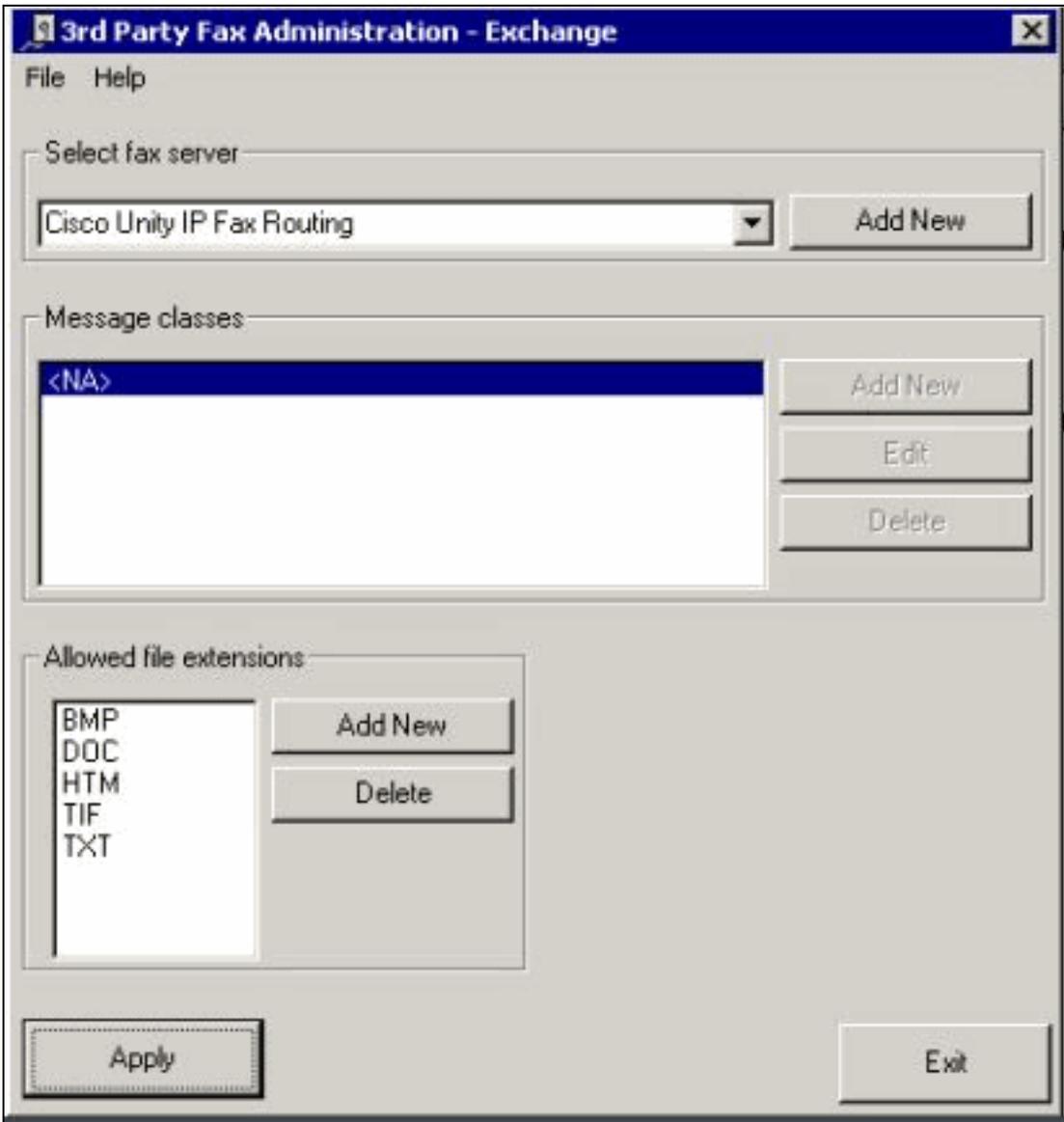
게이트웨이를 올바르게 설정합니다. Cisco Unity [IP Fax Outbound Configuration Example](#) 및 [IP Fax Configuration Wizard](#)를 참조하십시오.

메일이 최종 사용자의 메일 저장소에서 팩스 사서함이 있는 사서함으로 배달되도록 Exchange를 설정합니다. 또한 상태 메시지를 소스로 다시 보낼 수 있도록 역방향 방향으로 전달됩니다. 메일은 팩스 어카운트에서 게이트웨이로 전송하고 다시 전송할 수 있어야 합니다. 여기에는 DNS 및 Exchange의 컨피그레이션이 포함됩니다.

IP 팩스 서비스가 제대로 설치되었는지 확인합니다. 설정 프로그램이 활성화되면 Cisco Unity에서 컨피그레이션 마법사를 실행하고 시작 > 프로그램 > Unity > Cisco Unity IP Fax 컨피그레이션 마법사를 선택합니다.

IP 팩스 서비스에서 보내는 항목을 렌더링할 수 없습니다. IP 팩스 서비스는 전자 메일에 첨부된 파일을 받으면 먼저 해당 파일이 확장명을 인식할 수 있는지 확인합니다. 이러한 정보는 [서드파티 팩스 관리 도구](#)에서 [설정됩니다](#). 기본 확장명은 BMP, HTM, TIF 및 TXT입니다. DOC 확장명을 원하는 경우 이 예제에서 보여주는 것처럼 도구를 추가해야 합니다.

**참고:** 일부 릴리스에서는 서드파티 팩스 관리 도구 목록의 마지막 확장이 제대로 작동하지 않는 문제가 있습니다. 자세한 내용은 [주의](#) 섹션을 참조하십시오.



선택한 팩스 서버는 Cisco Unity IP Fax Routing입니다. DOC가 Allowed file extensions 영역에 추가됩니다. 즉, IP 팩스 서비스가 렌더링 엔진에 첨부 파일을 전달합니다. 이렇게 하면 렌더링 엔진은 확장자 유형과 연결된 응용 프로그램을 사용하여 문서를 열고 인쇄 드라이버가 있는 파일로 인쇄할 수 있습니다. DOC 확장은 일반적으로 WordPad 애플리케이션이 기본적으로 Cisco Unity 서버에 설치되어 있으며 대부분의 Microsoft Word 문서를 열기 때문에 허용됩니다. Acrobat Reader와 같은 응용 프로그램을 사용하면 PDF 파일을 팩스로 보낼 수 있습니다. 자세한 내용은 이 문서의 주의 섹션을 참조하십시오.

**참고:** Cisco에서 .doc 확장명에 대해 공식적으로 지원하는 애플리케이션은 WordPad뿐입니다. Wordview 또는 Microsoft Word와 같은 응용 프로그램을 문서 확장자와 함께 사용하면 렌더링 엔진이 실패할 수 있습니다.

**참고:** Cisco Unity 서버에 Microsoft Office를 설치하지 마십시오. 따라서 Cisco Unity의 적절한 운영이 중단됩니다. 지원되는 타사 애플리케이션만 Cisco Unity 서버에 설치해야 합니다. Cisco Unity 서버에 타사 소프트웨어를 설치할 때는 주의해야 합니다.

Start(시작) > Administrative Tools(관리 툴) > Services(서비스)에서 CSlpFaxSvc가 활성 상태이고 Log On As(다음으로 로그인) 필드가 AvCsMgr 서비스와 동일한지 확인합니다. AvCsMgr 서비스가 시작되지 않으면 문제가 있을 수 있습니다. 이는 CslpFaxSvc에도 영향을 줍니다. [Cisco Unity 3.1\(x\) 및 4.x](#)를 참조하십시오. [AvCsMgr 서비스가](#) 자세한 내용을 [시작하지 못했습니다.](#)

## 팩스 계정을 모니터링하도록 메일 클라이언트 설정

IP 팩스 서비스 및 게이트웨이를 설치한 후에도 작동하지 않는 경우 첫 번째 단계는 Microsoft Outlook Express(IMAP(Internet Mail Access Protocol) 또는 Outlook(또는 다른 메일 프로그램)을 사용하여 IP 팩스 사서함을 검사하는 것입니다. 최종 사용자가 이 사서함으로 메시지를 보내고 IP 팩스 서비스는 이 계정에서 메시지를 확인 및 전송하므로 메시지가 도착하고 올바르게 전송되었는지 확인하십시오. 여기에 표시된 모든 내용은 로그 파일과 Exchange Message Tracking Center를 사용하여 진단할 수 있습니다. 그러나 이는 제품의 작동을 나타내는 데에도 도움이 됩니다.

Outlook Express는 Cisco Unity 서버에 자동으로 설치됩니다. Outlook은 Cisco Unity를 위한 통합 메시징 구축에서 선택하는 메어입니다.

**주의:** Cisco Unity Server에 Microsoft Outlook을 설치하지 마십시오. 이로 인해 Cisco Unity가 작동을 중단하게 됩니다. Exchange 서버에는 Outlook을 설치하지 않아야 합니다. 클라이언트 시스템에 설치한 다음 Exchange 서버에 연결하려면 구성해야 합니다.

Outlook Express에 메일 계정을 하나 이상 추가할 때 IMAP을 지정해야 합니다. POP를 선택하면 모든 메일 메시지가 메일 저장소에서 즉시 다운로드됩니다. 그러면 IP 팩스 서비스가 새 메시지를 탐지하지 않습니다. IP 팩스 서비스 계정을 지정합니다. 암호를 모르는 경우 Active Directory 사용자 및 컴퓨터에 액세스하고 IP 팩스 계정을 찾은 다음 암호를 재설정하는 것이 안전합니다.

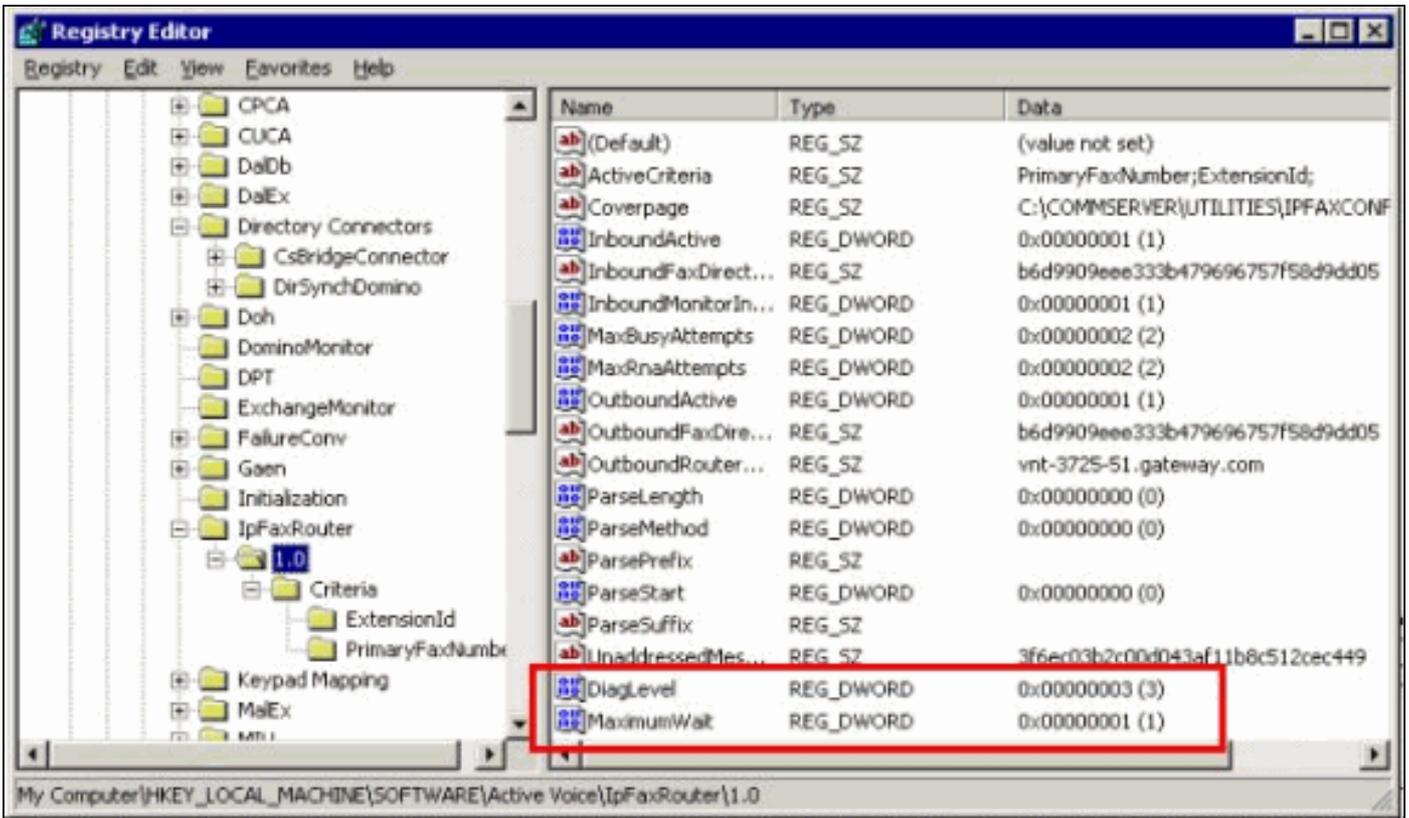
이 문서에서 제공하는 문제 해결 방법만이 사용할 수 있는 방법이 아닙니다. 예를 들어 IP 팩스 계정에 액세스할 수 있고 Outlook 또는 Outlook Express를 통해 로그인한 경우 해당 위치에서 최종 사용자에게 직접 테스트 전자 메일을 보낼 수 있습니다. 게이트웨이로 직접 이메일을 보낼 수도 있습니다. 이것은 한 요소나 다른 요소가 제대로 작동한다는 것을 증명하기 때문에 문제를 줄입니다.

## 레지스트리 설정

문제 해결을 지원하려면 추가 출력이 기록되도록 진단 수준을 변경하십시오. CSIPFaxSvc가 실행되는 Cisco Unity 서버에서 이 기능을 활성화하려면 **Start > Run**을 선택하고 명령 상자에 regedit를 입력합니다. 레지스트리 편집기 도구가 나타나면 HKEY\_LOCAL\_MACHINE\SOFTWARE\Active Voice\IPFaxRouter\1.0을 선택합니다.

편집 > 새로 만들기 > DWORD 값을 선택합니다. 지정된 이름은 DiagLevel입니다. 생성된 후 이 키를 두 번 클릭합니다. Value Data 필드에 3을 입력합니다. 최대 진단 수준입니다. 나중에 0(기본값)으로 변경할 수 있습니다. 이 변경 사항으로 서비스를 다시 시작할 필요가 없습니다.

**주의:** 잘못된 레지스트리 키를 변경하거나 잘못된 값을 입력하면 서버가 작동하지 않습니다. 레지스트리 키를 편집하기 전에 문제가 발생할 경우 복원하는 방법을 알고 있는지 확인합니다. 자세한 내용은 레지스트리 편집기 도움말의 복원 항목을 참조하십시오. Cisco Unity 서버의 일반적인 백업은 레지스트리를 백업하지 않습니다. Cisco Unity 장애 조치 시스템의 경우 레지스트리 변경 사항이 복제되지 않으므로 한 Cisco Unity 서버의 레지스트리 변경 사항은 다른 Cisco Unity 서버에서 수동으로 수행됩니다. 레지스트리 키 설정을 변경하는 방법에 대한 자세한 내용은 Cisco 기술 지원부에 문의하십시오.



경우에 따라 MaximumWait 키가 유용한 또 다른 키입니다. 이 키가 없을 경우 기본값은 24(시간)입니다. 최종 사용자에게 배달 못 함 메시지를 보내기 전에 메시지를 보류하는 최대 시간을 지정합니다. 이 키는 트러블슈팅에 직접적으로 도움이 되지는 않지만 자주 요청됩니다.

OutboundRouterDomain 키는 아웃바운드 팩스 전송을 위해 Cisco IOS 게이트웨이로 배달될 주소로 메일 메시지를 보낼 때 메일 접미어로 사용되는 이메일 도메인을 포함합니다. 이 작업을 수행하려면 이 예에서 vnt-3725-51.gateway.com의 값이 DNS를 통해 확인되어야 합니다. 이 항목을 ping하는 것이 도움이 될 수 있습니다. 이 항목이 올바르지 않으면 발신자에게 이메일이 반환되고 게이트웨이 항목을 확인할 수 없음을 알립니다.

**팁:** 이 팩트는 문제 해결에 유용합니다. Cisco IOS 게이트웨이를 제외한 모든 항목이 설치된 경우 이 레지스트리 항목을 일시적으로 DNS에 없는 것과 같은 가짜 값으로 변경할 수 있습니다. 그런 다음 팩스 사서함으로 팩스 메시지를 보냅니다. CspFaxSvc 서비스는 메시지를 검색하고 첨부 파일을 렌더링한 다음 게이트웨이를 찾지 못합니다. NDR(Non-Delivery Receipt) 알림이 게이트웨이로 전송되는 팩스 메시지의 렌더링된 TIF 첨부 파일과 함께 발신자에게 다시 전송됩니다. 이를 수행할 경우 다음을 확인했습니다.

- 최종 사용자로부터 팩스 계정으로 전자 메일을 보낼 수 있습니다.
- 팩스를 성공적으로 렌더링할 수 있습니다.
- 팩스 계정에서 최종 사용자에게 전자 메일을 다시 보낼 수 있습니다.

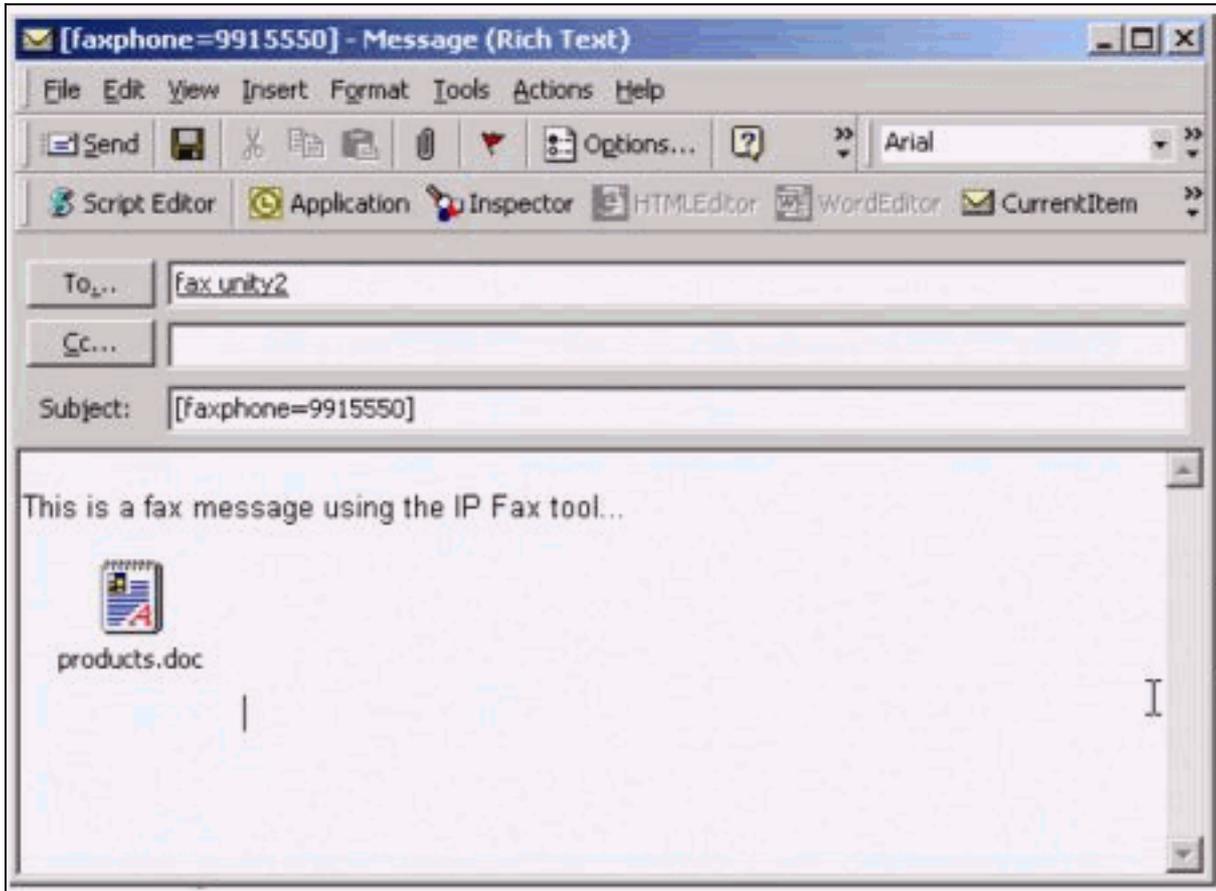
팩스 계정에서 게이트웨이로, 게이트웨이에서 팩스 계정으로 메시지를 보내는 경우를 제외하고 모든 것이 제거되었습니다.

Coverpage 키에는 렌더링된 커버 페이지의 위치가 포함됩니다. 전송된 모든 팩스에는 페이지 수와 같은 일부 정보가 포함된 표지가 포함되어 있습니다. 이 페이지를 편집하고 사용자 지정 정보 또는 회사 로고를 삽입할 수 있습니다. 키가 존재하지 않는 파일로 변경되면 표지 페이지가 전송되지 않습니다.

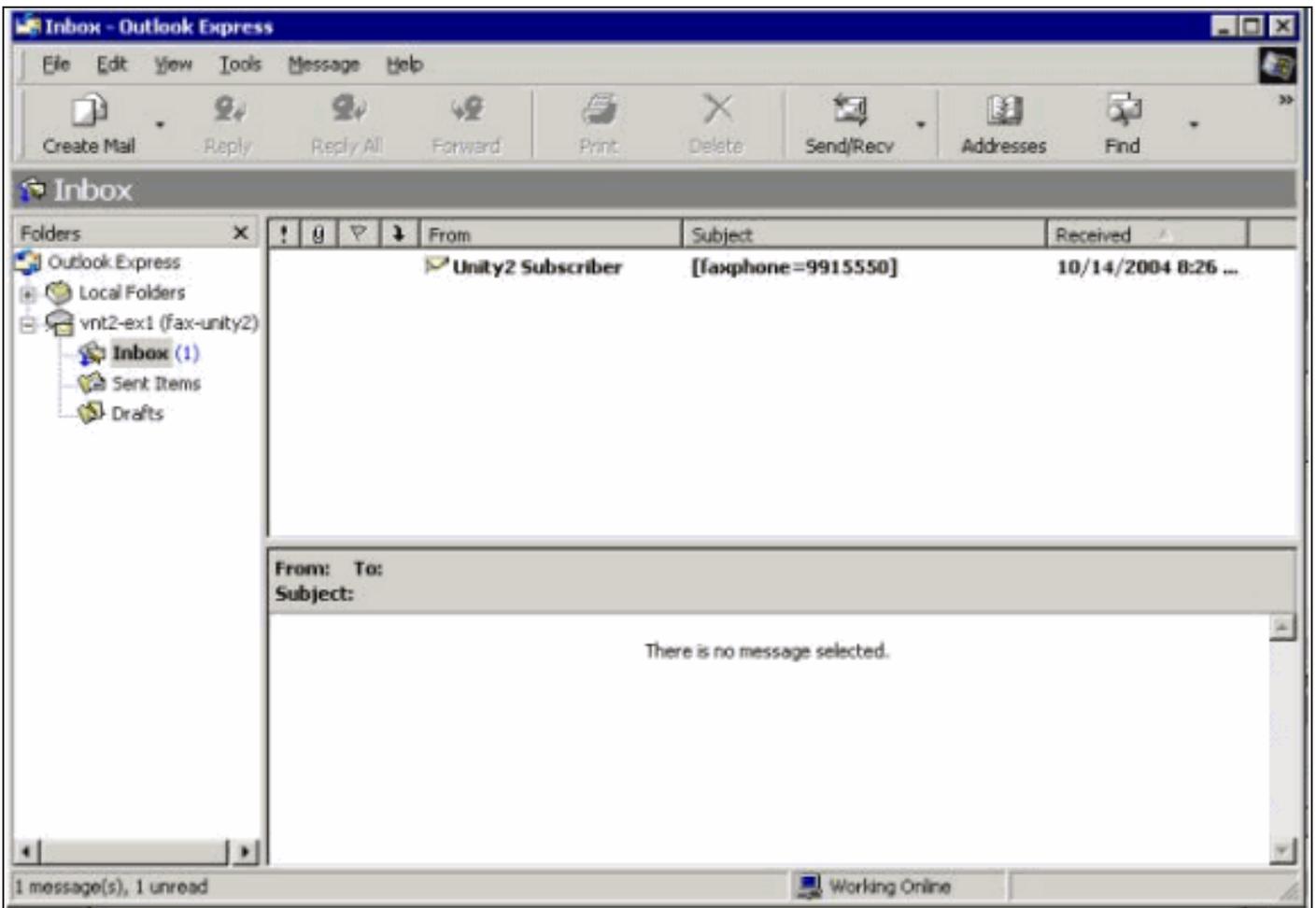
## [각 구성 요소 문제 해결](#)

## 1단계: 최종 사용자가 팩스 계정으로 메시지를 보낼 수 있는지 확인합니다.

사용자의 메시지를 팩스 계정으로 보냅니다. 이 다이어그램에서 메시지는 팩스 사서함인 fax-unity2로 전달됩니다. 사용자가 통화하려는 번호는 제목 줄에 지정됩니다(이 경우 9915550). 사용자가 팩스로 보내려는 문서가 첨부됩니다.

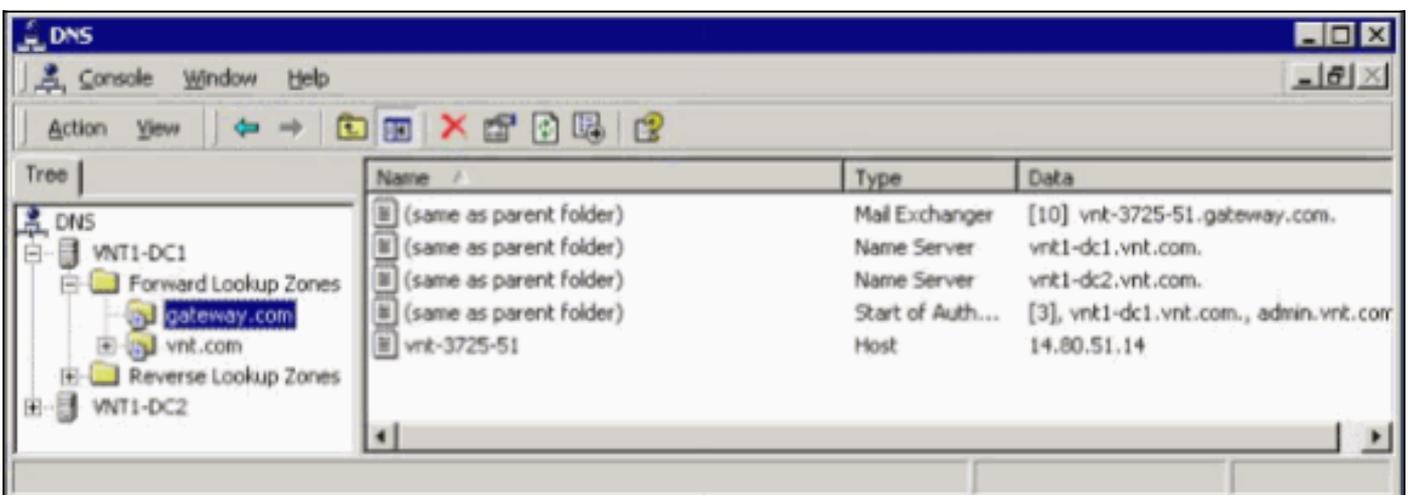


이 메시지가 전송되면 처음에 설정된 팩스 사서함에 도착합니다. 이 그림은 사서함의 Outlook Express 보기를 보여줍니다.

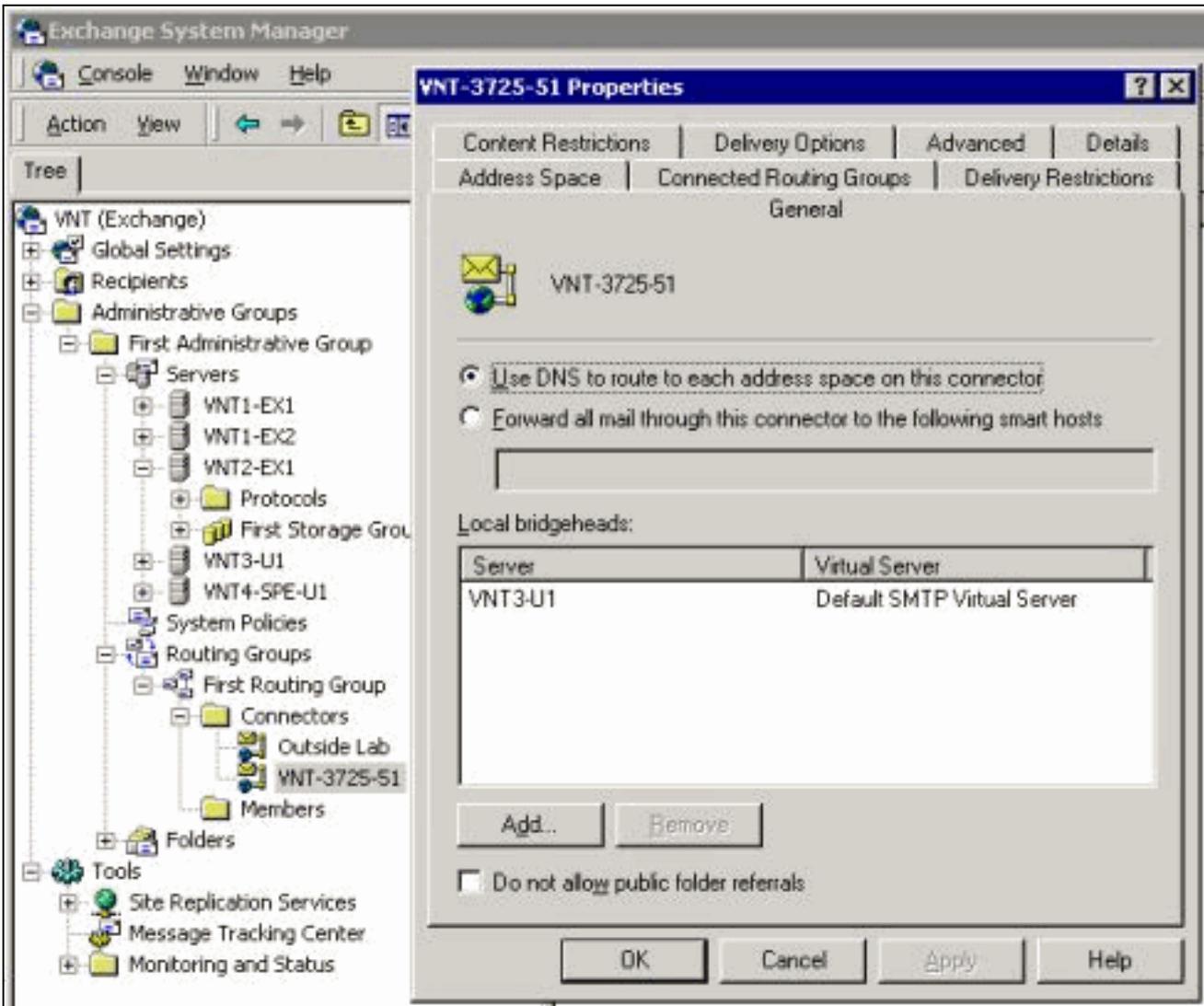


메시지가 도착하지 않으면 기본적인 이메일 전달 문제가 발생합니다. 발신자가 일종의 배달 실패 메시지를 받았는지 확인해야 합니다. Microsoft Exchange 2000/2003에는 Exchange 메시지 추적 기능이 있어 메일 전달에 대한 추가 세부 정보를 볼 수 있습니다. 발신자와 팩스 계정이 두 개의 개별 도메인(예: a.com 및 b.com)에 있는 경우, 전송 메일 시스템이 대상 메일 서버로 메일을 전달하도록 올바르게 구성되어 있는지 확인합니다. 여기에는 DNS에서 A 및 MX 레코드를 설정한 다음 SMTP 커넥터를 추가하여 메일을 라우팅하는 작업이 포함됩니다. 또한 배달 못 함(non-delivery) 메시지가 최종 사용자에게 다시 전송되도록 반대 방향으로 메일을 보낼 수 있어야 합니다.

이 그림은 A 및 MX 레코드가 있는 IOS 게이트웨이에 대한 DNS 항목을 보여줍니다.



해당 메일 호스트에 대한 SMTP 커넥터가 Exchange에 필요합니다.



DNS는 주소 공간으로 라우팅하는 데 사용됩니다. 주소 공간은 c3725-51.gateway.com으로 구성됩니다. 이는 여러 컨피그레이션 중 하나뿐입니다. 이는 테스트용 샘플 환경 설정입니다. 일반적인 기업 Exchange 환경은 훨씬 복잡합니다. 요점은 사용자로부터 팩스 사서함으로 메일을 가져오도록 하는 것입니다. 이러한 항목에 대한 자세한 내용은 Microsoft 웹 사이트를 참조하십시오.

메일이 최종 사용자로부터 팩스 계정으로 전송되고 팩스 계정에서 최종 최종 사용자에게 전송되는지 확인합니다.

## 2단계: IP 팩스 서비스에서 메시지를 처리했는지 확인

IP 팩스 서비스 구성 중에 폴링 간격이 지정됩니다. 이는 IP 팩스 서비스가 팩스 사서함에 로그인하여 새 메시지를 확인하는 빈도를 나타냅니다. Cisco에서는 제품의 올바른 작동을 확인할 때까지 이 값을 낮게 설정하는 것이 좋습니다.

최상의 진단 로그 파일은 IPFaxSvc<date/time>.log입니다. \Commserver\logs directory폴더에 있습니다.

**참고:** 이 추적 샘플에서는 일부 정보가 제거되었습니다.

각 간격으로 로그 파일이 로그인하는 사서함을 볼 수 있습니다. 아웃바운드 팩스를 처리하는 것을 볼 수도 있습니다.

**참고:** 모든 추적은 앞서 언급한 DiagLevel = 3 레지스트리 설정을 반영합니다.

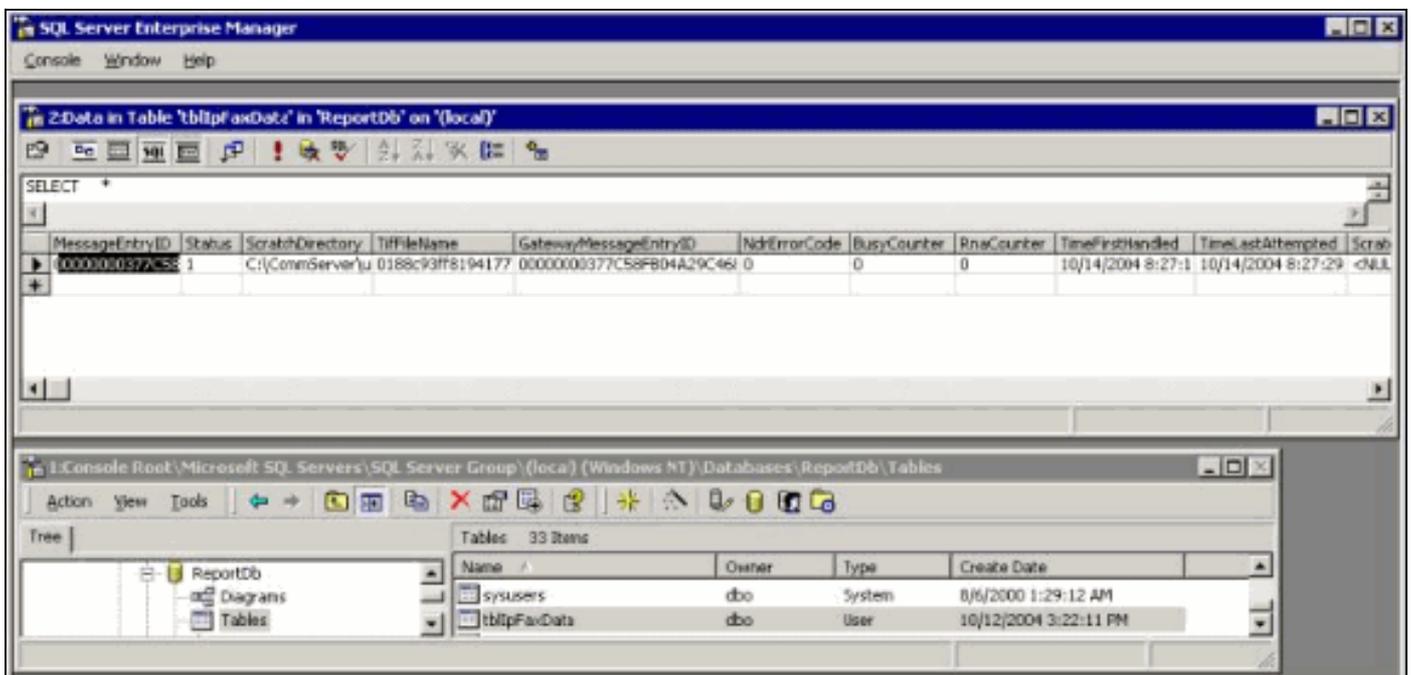
```
"Logging on to mailbox [VNT2-EX1\fax-unity2]"
"CipFaxCdoHelper::LogonToMailbox() name returned from Logon is [fax unity2]"
"Processing outbox, message count= 1"
"CipFaxCdoHelper::ProcessFaxOutbox() processing message receipts"
"CipFaxCdoHelper::DetermineMessageDisposition() current message
subject=[[faxphone=9915550]]"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

아웃바운드 프로세스가 올바른 사서함에 성공적으로 로그인하고 메시지가 있음을 인식하고 제목을 검색하며 아웃바운드 팩스로 간주해야 합니다.

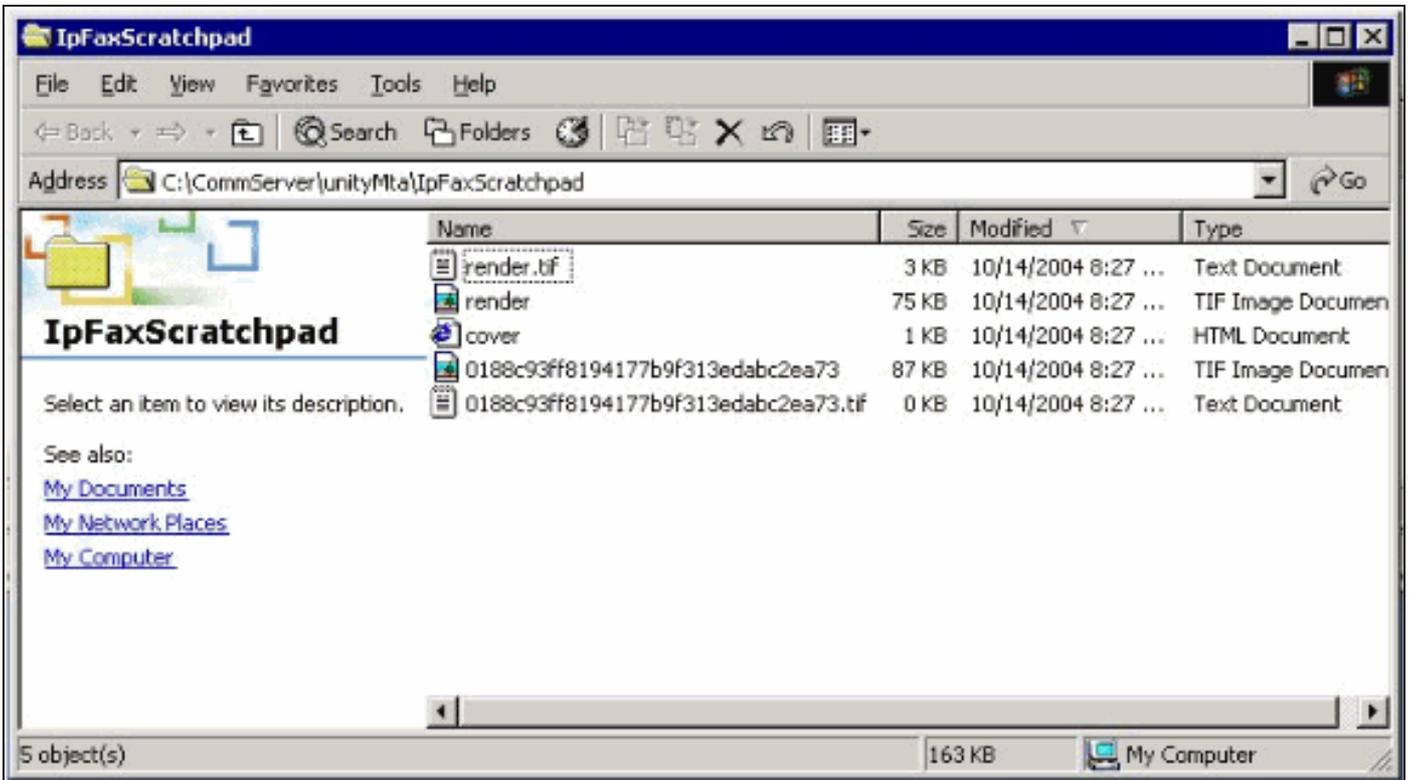
```
["_RecordsetPtr::Open(Select * from tblIpFaxData where MessageEntryID =
'00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C512000000
05E5B20000E7E65E238D0DEF46B5F8EAF983C6C51200000005F55A0000') returned [0] rows -
Code: 0x80004005. Method: CipFaxAdoHelper::GetRecordSet. [IpFaxAdoHelper.cpp::403]"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

```
"CipFaxAdoHelper::Execute() Cmd=[INSERT INTO tblIpFaxData (MessageEntryID, Status,
ScratchDirectory, TiffFileName, 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')]"
```

이제 메시지를 읽습니다. 첫 번째 문은 서비스가 Cisco Unity 서버의 Reports 데이터베이스에 있는 tblIpFaxData에 대해 SQL 쿼리를 실행하여 해당 MessageEntryID(메시지의 MAPI ID)가 있는 메시지가 있는지 여부를 확인합니다. 해당 항목을 찾을 수 없으므로(0개의 행이 반환됨) 이 정보가 포함된 새 행이 추가됩니다. 이 MessageEntryID는 게이트웨이로 전송된 전자 메일의 제목 필드로 사용됩니다. SQL을 직접 살펴보면 다음과 같습니다.



SQL은 각 메시지를 추적하는 데 사용됩니다. 팩스 사서함이 비어 있지 않은 경우 SQL은 각 간격으로 팩스 사서함을 통과하고 이 메시지 ID를 기반으로 각 메시지를 계산했는지 확인합니다. 첨부 파일은 스크래치 디렉토리 C:\CommServer\unityMta\IpFaxScratchpad in this case에 복사됩니다. 이 메시지에 대한 DSN이 있는 경우 BusyCounter 또는 유사한 내용만 업데이트해야 합니다. 그러나 이는 서비스가 각 메시지를 추적하는 방법입니다. 파일이 스크래치 디렉토리에 복사되었으므로 렌더링이 수행됩니다.



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

```
"CIpFaxCdoHelper::RenderCurrentMessage() outputting message body to filename=[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"
```

렌더링 프로세스가 시작됩니다. 모든 팩스에는 커버 페이지, 메시지 본문이 있는 페이지 및 모든 첨부 파일이 포함됩니다. 표지 페이지에는 팩스로 처리된 페이지 수가 표시됩니다. 본문은 항상 먼저 보내지므로 먼저 렌더링됩니다. 한 번에 하나의 메시지만 렌더링하므로 항상 메시지 본문 MsgBody.txt와 임시 출력 파일 render.tif를 호출합니다. 렌더링은 ImageMaker 소프트웨어에서 수행합니다. 각 렌더링 프로세스가 완료되면 렌더링된 페이지 수를 반환합니다. 이 경우에는 한 페이지입니다.

### 3단계: 렌더링 작업 확인

ImageMaker 렌더링 엔진은 인쇄 드라이버입니다. 입력 파일을 가져오고 연결된 기본 응용 프로그램(워드패드, 메모장 등)을 열고 ImageMaker 드라이버를 사용하여 파일을 파일로 인쇄합니다. 이렇게 하면 TIF 파일이 만들어집니다.

전체 진단 파일은 [부록](#) 섹션에 있습니다. 렌더링 엔진과 관련된 문제를 Cisco 기술 지원부에 제출할 때 다음 파일이 필요합니다.

- C:\imgMaker\Conversion\_Toolkit\log\ODCDAEMON.log - 배치 파일을 통해 변환을 수행하는 데몬입니다.
- C:\log\imgMakerDocCnvert Driver.txt—드라이버 수준 정보

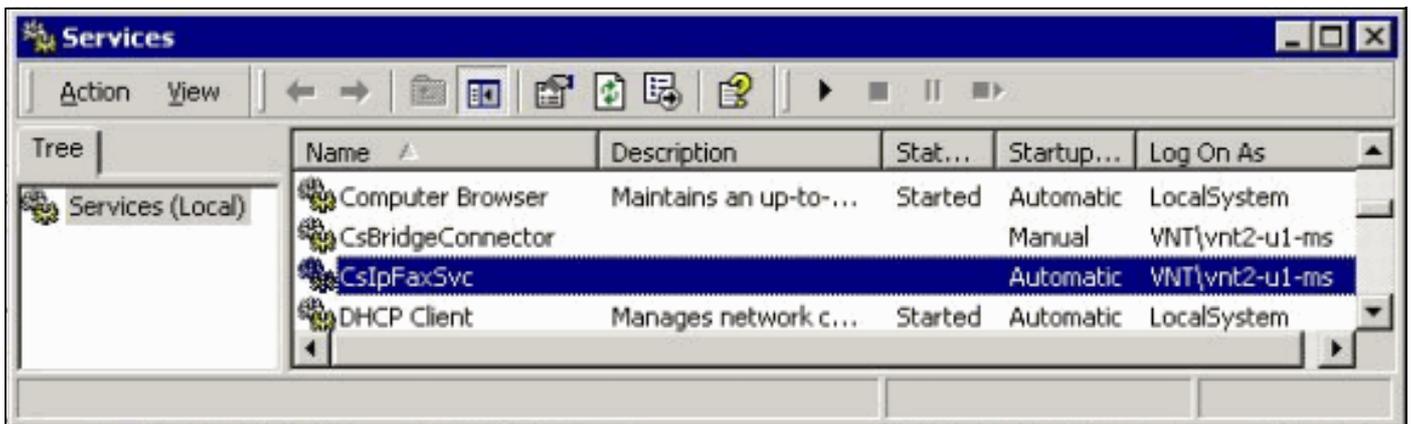
가장 유용한 정보는 ODCDAEMON.log 파일에 있습니다. 렌더링할 각 파일에 대해 다음을 볼 수 있습니다.

```

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

```

살펴볼 첫 번째 요소는 **사용자로 로그인됩니다**. CslpFaxSvc 서비스에서 로그인하는 데 사용하는 계정과 동일합니다.



이 사용자는 모든 애플리케이션을 실행하고 이를 렌더링하므로 이 점을 유의해야 합니다. 다음 요소는 **입력 및 출력 파일 이름**입니다. 세 번째 요소인 **response**는 렌더링된 페이지 수를 나타냅니다. 이 경우에는 1입니다. 응답이 음수이면 오류를 나타냅니다. 오류가 있으면 바로 앞에 오는 선을 확인해야 합니다. 대부분의 경우 특정 파일 유형에 대해서만 오류가 발생합니다. ODCDAEMON 로그 파일은 모든 렌더링에 대해 오류가 발생하는지 또는 특정 확장 유형에만 오류가 발생하는지를 표시합니다.

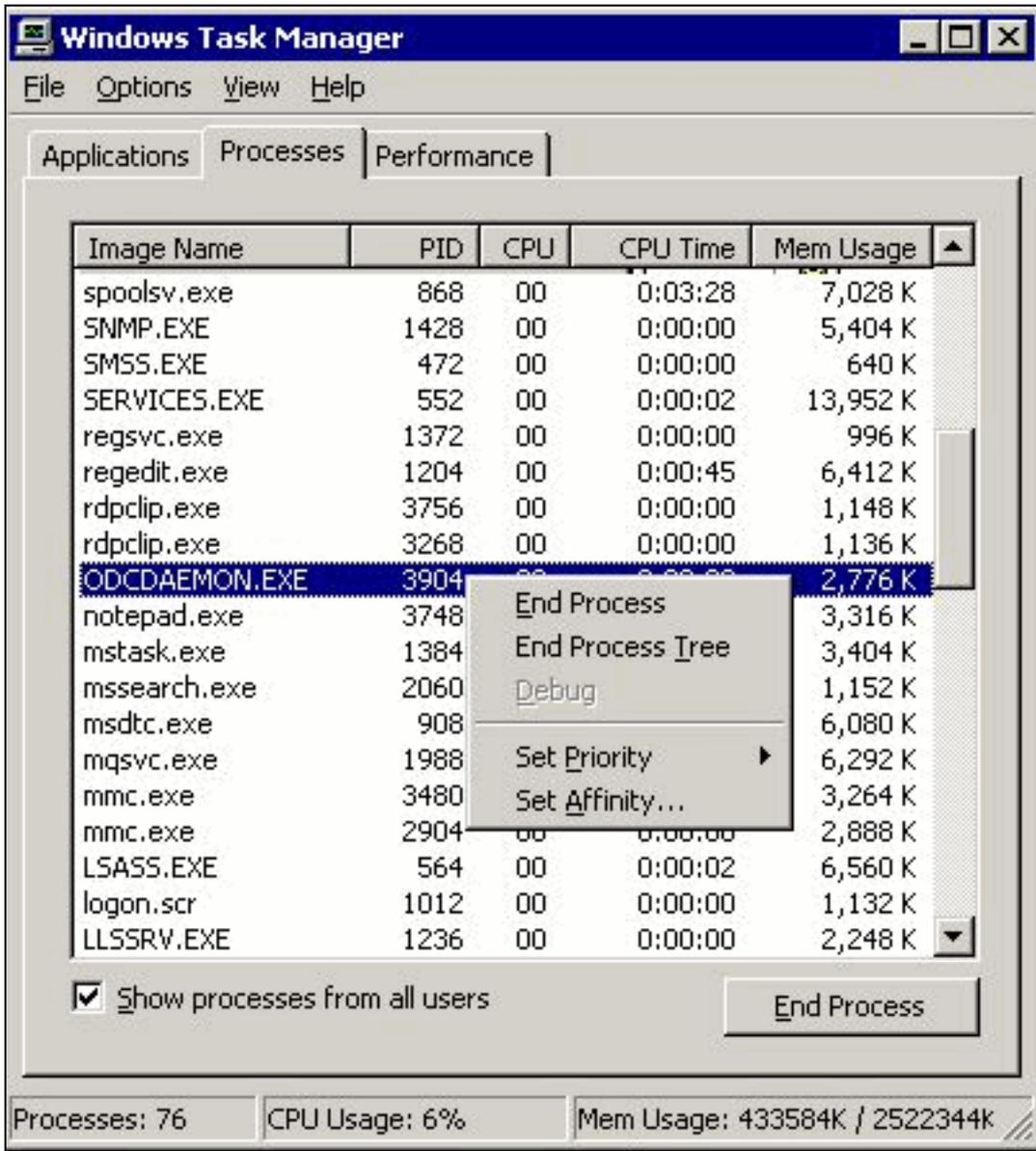
렌더링 엔진을 수동으로 실행하여 이를 추가로 테스트하는 것이 좋습니다. CslpFaxSvc 서비스를 실행하는 동일한 사용자로 로그인합니다. 서비스에서 렌더링 엔진을 시작할 때 동일한 오류가 발생하는지 여부를 확인하기 위한 것입니다.

허용되는 각 확장 유형의 파일이 있는 경우 파일을 두 번 클릭하여 각 파일을 엽니다. 연결된 애플리케이션이 열리고 업데이트(또는 유사한 프로세스)가 있는지 확인합니다. 예를 들어 Acrobat의 경우 이 기능을 비활성화할 수 있습니다.

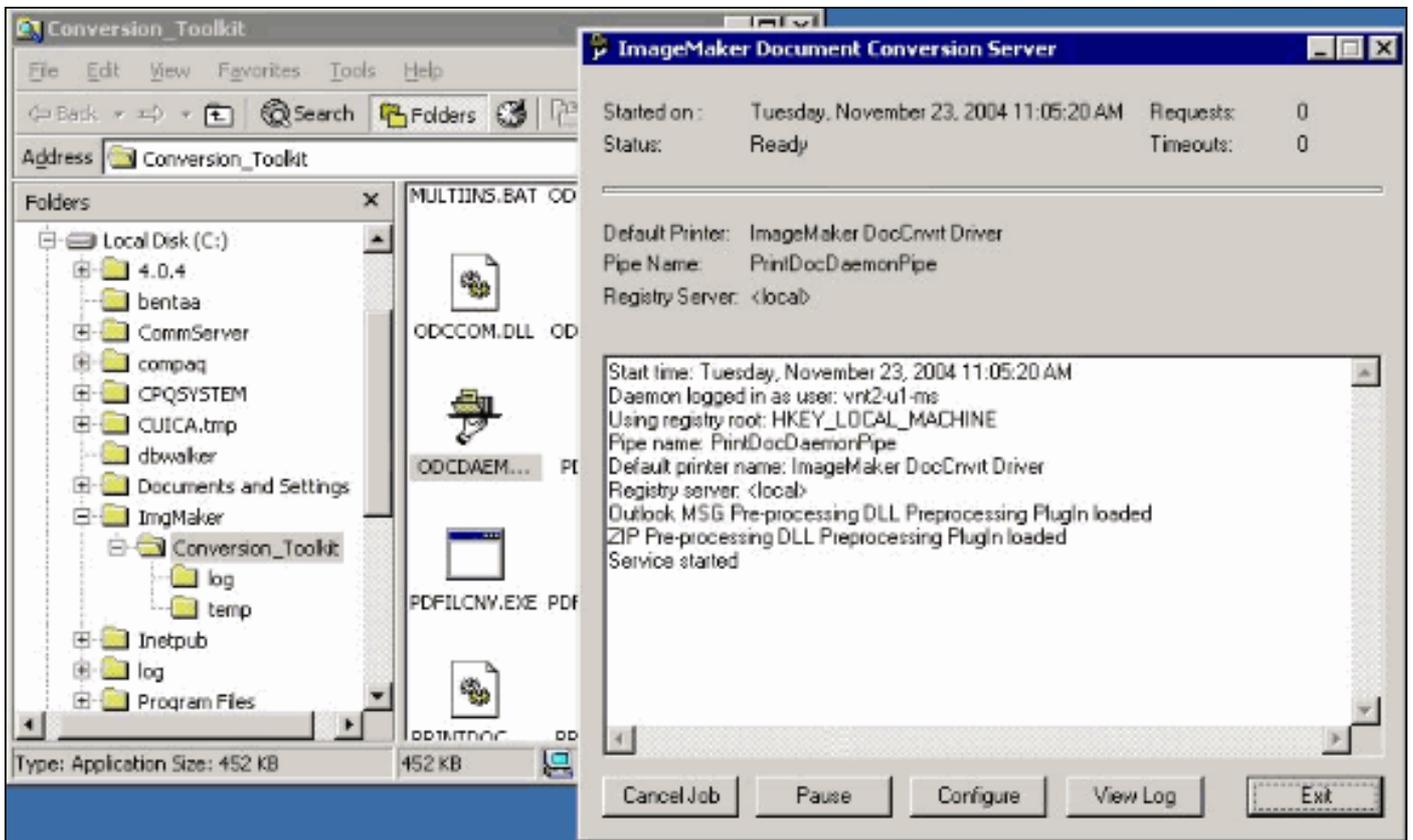
CslpFaxSvc를 중지합니다. ODCDAEMON 프로세스가 계속 실행되는 경우 작업 관리자로 중지합니다.

**참고:** 추가 참조를 위해 ODCDAEMON 로그 파일을 저장해야 하는 경우 CslpFaxSvc를 ping하기 전에 파일을 저장해야 합니다. 서비스를 다시 시작하면 ODCDAEMON 로그 파일이 지워지고 출력

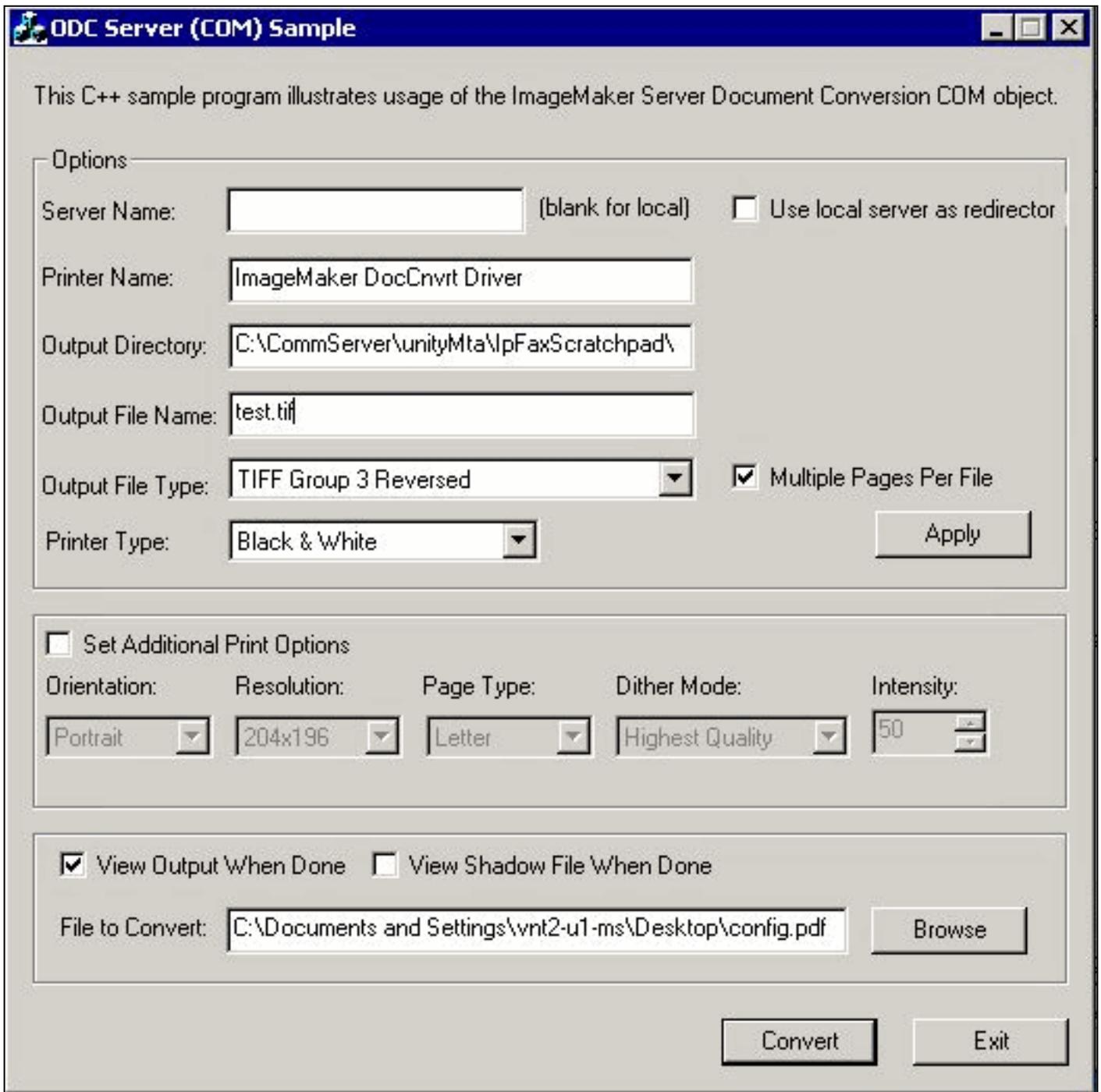
이 손실됩니다.



C:\ImgMaker\Conversion\_Toolkit 폴더에서 ODCDAEMON.EXE 프로세스를 시작합니다. 다음 그림과 같이 창이 표시됩니다.



동일한 폴더에서 CPCOM.EXE 파일을 실행합니다.



대부분의 기본 설정은 정상입니다. 변환할 파일 필드에서 원하는 입력 파일이 선택됩니다. 모든 확장 유형을 별도로 테스트할 것을 권장합니다. 출력 파일이 변경되지 않으면 입력 파일이 출력 파일에 추가됩니다.

변환(Convert)을 **클릭합니다**. 애플리케이션이 인쇄되기 시작하면 이 대화 상자가 나타납니다.



확인을 클릭합니다. 완료 시 출력 보기 옵션을 선택합니다. TIF 뷰어가 시작되고 출력 파일이 표시됩니다. 오류가 있는 경우 이 예와 같이 로그 파일에 나타난 것과 동일한 경우가 있습니다.

```
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
```

입력 파일은 TIF 첨부 파일입니다. 그러나 파일 유형은 DOC(MSPaper.Document)입니다. 이 경우 Word 및 Excel과 같은 Microsoft 응용 프로그램이 서버에 설치되고 레지스트리에서 파일 연결이 변경된다는 문제가 발생합니다. 이 오류를 수정하려면 HKEY\_CLASSES\_ROOT\tif 및 TIF를 편집하고 Default 키를 MSPaper.Document 대신 TIFImage.Document로 변경합니다.

각 파일 유형이 확인되면 CPCOM 및 ODCDAEMON 창을 닫고 CslpFaxSvc 서비스를 다시 시작합니다.

모든 첨부 파일이 실패하거나 초기화 오류가 발생한 경우 렌더링 엔진을 다시 설치합니다. IPFaxConfigWizard가 있는 ImageMaker 폴더에서 instfax.exe -z 명령을 사용하여 제거합니다. 그런 다음 다시 설치하고 재부팅해야 하며 시작 > 프로그램 > Unity에서 Cisco Unity IP 팩스 컨피그레이션 마법사를 다시 실행합니다.

오류를 테스트하는 마지막 옵션은 데몬 프로세스를 건너뛰고 명령줄에서 파일을 렌더링하는 것입니다. 대부분의 경우 전에 실패했다면 여기에서도 실패하게 됩니다. 그러나 경우에 따라 추가 정보를 제공합니다. 이는 PDFILCNV 실행 파일의 도움을 받아 수행됩니다. 이 프로그램은 IPFaxConfigWizard 설치 경로의 ImageMaker 하위 디렉터리에 있습니다. 예를 들면 다음과 같습니다.

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

표시되는 파일을 열 수 있는지 확인합니다.

다음 예는 IpFaxSvc 로그 파일에서 가져온 것입니다.

```
"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"
```

DOC 첨부 파일이 렌더링됩니다. 이 파일은 이전에 만든 render.tif 파일에 추가됩니다. 2페이지로 되어 있어서 총 3페이지 수가 됩니다.

```
"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] "
```

커버 페이지는 cover.htm 파일입니다. 몇 가지 변수가 있는 간단한 HTML 템플릿입니다. 메시지를 보내는 가입자의 이름, 호출된 번호 및 전송된 페이지 수를 입력합니다.

custom cover.htm 파일이 생성된 후 messageID.tif 파일로 렌더링됩니다. 이전에 만든 render.tif 파일은 messageID.tif 파일에 추가됩니다(예: 추가된 파일). 이때 messageID.tif 파일은 전자 메일 또는 팩스로 전송되는 실제 메시지입니다.

성공적으로 렌더링됩니다. SQL의 일부 필드가 업데이트되고 메시지가 전송됩니다.

```
"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']";
```

메시지의 상태가 1로 설정되고 마지막 업데이트 시간이 채워집니다. 게이트웨이로 전송됩니다.

## 4단계: 라우터가 메시지를 수신하고 전달하는지 확인

IP 팩스 계정에서 게이트웨이로 보낸 메시지는 fax#####@router.gateway.com 형식이어야 합니다. 일반적으로 DNS A 및 MX 레코드는 DNS에 필요하며, 메시지를 해당 호스트로 라우팅하려면 SMTP 커넥터를 Exchange에 추가해야 합니다. 이 작업이 완료되면 게이트웨이 트러블슈팅을 진행할 수 있습니다.

라우터를 설정한 경우에만 다음 예와 같이 텔넷을 사용하여 SMTP 포트에 액세스하고 메시지를 수동으로 보낼 경우 라우터가 제대로 작동하는지 확인할 수 있습니다.

```
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 ESMTTP 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.
```

이 경우 알 수 없는 오류가 있습니다. 아웃바운드 팩스를 전달하는 데 사용되는 PRI(Primary Rate Interface)가 다운되었기 때문입니다. 다음은 또 다른 예입니다.

```
telnet vnt-3725-51.gateway.com 25
Trying 14.80.51.14, 25 ... Open
220 vnt-3725-51.gateway.com Cisco NetWorks ESMTTP 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
```

이 경우 Unable to remote host 오류 메시지 Cisco IOS 컨피그레이션 mta 목록에 error.com 구성되지 않았음을 의미합니다.

라우터에 텔넷하거나 콘솔을 통해 액세스하여 게이트웨이 문제를 직접 해결합니다. Telnet인 경우 **terminal monitor** 명령을 실행합니다. 디버그 출력이 표시됩니다. 모두 디버그 팩스 데이터를 설정합니다. 아웃바운드 음성 회로가 PRI인 경우 debug isdn q931도 관련 명령입니다.

**경고:** 라우터에서 추적을 활성화하면 성능에 영향을 줄 수 있습니다. 완료되면 no debug all 명령을 사용하여 추적을 끕니다.

**참고:** 공간을 절약하기 위해 이러한 로그가 잘렸습니다.

이 디버그 출력은 활성화 예제의 것입니다.

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  
5F8EAF983C6C5120000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C5120000005FB680000`**

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: 0000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0  
DEF46B5F8EAF983C6C5120000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C5120000005  
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: `0000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C51200000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C51200000005FB680000`**

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

```

라우터의 메일 서버가 메시지를 수신합니다. SMTP 대화의 모든 필드는 번호를 호출할 때까지 라우터 코드의 여러 레벨에서 구문 분석됩니다. 몇 가지 컨피그레이션 단계에 중요한 요소가 나열되어 있습니다.

show running-config 출력에서 다음을 확인할 수 있습니다.

- **팩스 인터페이스 유형 팩스 메일**

- OffRamp 소프트웨어는 Flash 또는 FTP/TFTP 서버에 로드해야 합니다. 그런 다음 애플리케이션이 설정됩니다. 예를 들어, 애플리케이션 음성 오프램프 플래시를 호출합니다. **app\_faxmail\_offramp.2.0.1.1.tcl**. 참고: OnRamp OffRamp 기능에 사용되는 TCL(Tool Command Language) 소프트웨어는 [Cisco 다운로드\(등록된 고객만 해당\)](#)에서 다운로드됩니다.

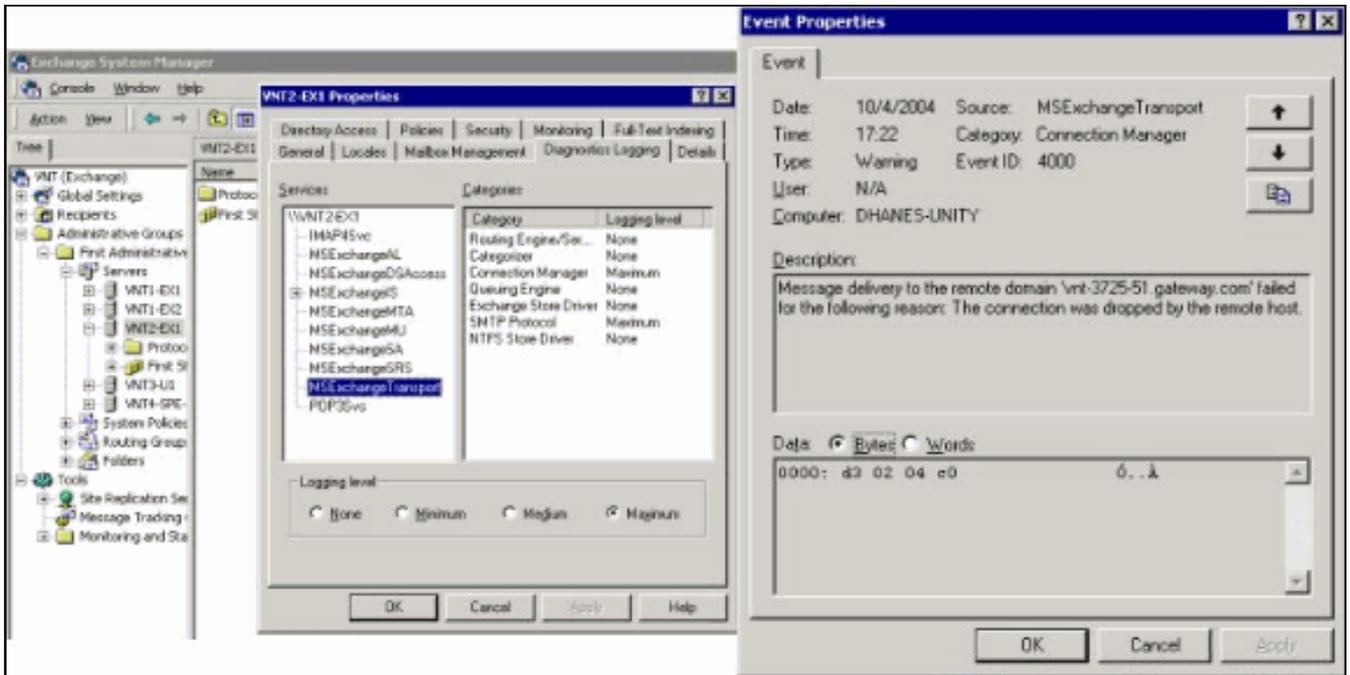
- **mta receive maximum <recipients>** - 0보다 큰 숫자
- **mta receive aliases <string>** - DNS 접미사와 일치해야 합니다. 예를 들어, Cisco Unity의 전자 메일이 fax=912345@myrouter.com으로 전송되면 별칭은 myrouter.com이어야 합니다. 여러 **mta 수신 별칭**을 가질 수 있습니다.
- **mta send server <IP address of Exchange server>** - DSN을 보내려면 게이트웨이가 메일 서버에 연결하여 메시지를 전달해야 합니다. 여러 Exchange 서버를 구성할 수는 있지만 Exchange 서버를 하나만 구성해야 합니다. 유효하지 않은 모든 **mta 송신 서버** 주소가 제거되었는지 확인합니다. 그렇지 않으면 목록의 첫 번째 항목이 항상 사용됩니다.
- **mta receive generate permanent-error** - 라우터가 DSN을 Cisco Unity로 올바르게 다시 전송하려면 이 작업이 필요합니다.
- **다이얼 피어 구성.** 예를 들면 다음과 같습니다.

```

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
!

```

메시지가 게이트웨이에 도착하지 않으면 Exchange 문제를 해결해야 합니다. Exchange 2000 내의 진단 기능은 이 작업에 매우 유용합니다. 예를 들면 다음과 같습니다



범주 SMTP 프로토콜 및 연결 관리자의 MSExchangeTransport 진단이 사용하도록 설정되었습니다. Exchange System Manager의 개별 서버 속성 아래에 있습니다. 메시지를 보내면 Application Event Viewer 로그에 오류가 표시됩니다. 가장 일반적인 컨피그레이션 중 하나는 라우터의 mta **receive aliases** 매개변수가 이메일 접미사와 일치하지 않을 때입니다. 예를 들어, 팩스 서비스가 fax=91234@router.abc.com에 메시지를 보내려고 하면 게이트웨이에 mta receive aliases router.abc.com이 구성되어 있어야 합니다.

## 5단계: 라우터가 DSN 메시지를 전송하는지 확인

생성된 알림입니다. 공간을 절약하기 위해 추적의 일부가 생략됩니다.

```
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/SMTPLC/esmtplib_client_engine_open:
from=, to=fax-unity2@vnt.com
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPLC/esmtplib_client_engine_open:exit@1866
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPLC/esmtplib_client_engine_add_headers:
from_comment=Fax Daemon
Oct 15 08:30:41.739: //-1/xxxxxxxxxxxxx/SMTPLC/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/SMTPLC/esmtplib_client_engine_work_routine:
socket 0 readable for first time
Oct 15 08:30:41.983: //-1/xxxxxxxxxxxxx/SMTPLC/esmtplib_client_engine_work_routine:
socket 0 readable for first time
Oct 15 08:30:41.983: //-1/xxxxxxxxxxxxx/SMTPLC/esmtplib_client_engine_getln:
(C)R: 220 vnt2-ex1.vnt.com Microsoft ESMTPL 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/SMTPLC/esmtplib_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:

```

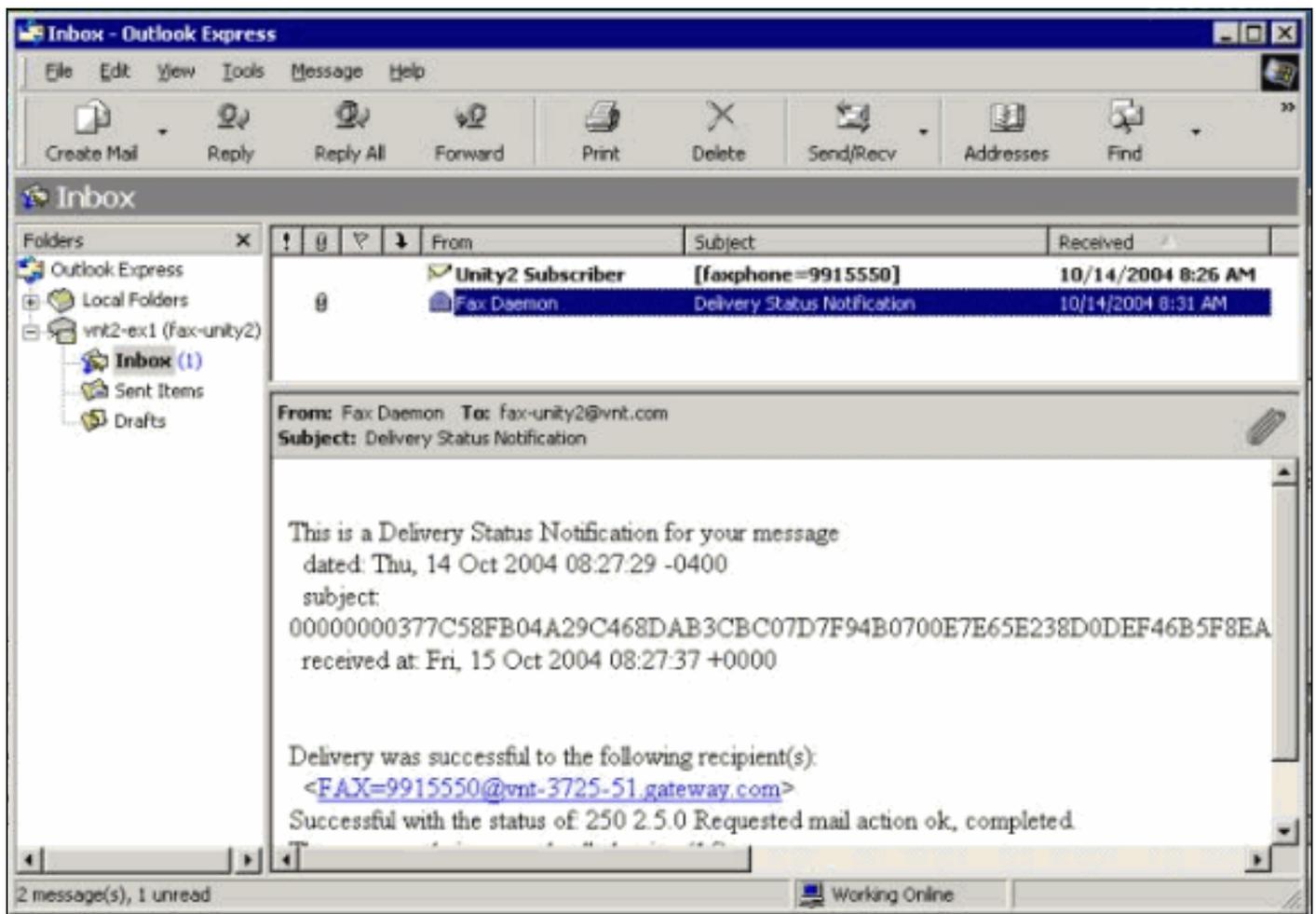
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Message-ID: <00042004083044107@vnt-3725-51.gateway.com>
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Date: Fri, 15 Oct 2004 08:30:44 +0000
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxx/SMTPC/esmtp_client_engine_write:
return code=0
Oct 15 08:30:44.107: //-1/xxxxxxxxxxxx/SMTPC/esmtp_client_engine_writeln:
(C)S: Subject: Delivery Status Notification
...

```

작동하지 않으면 컨피그레이션의 mta **send server** 명령을 올바른 Exchange 서버 및 포트(기본적으로 25)로 설정해야 합니다. mmoip 다이얼 피어에서 팩스 메시지가 전송될 때 매칭하는 DSN을 구성해야 하며 mta receive **generate permanent-error** 명령이 있습니다. 명령이 없거나 잘못된 메일 서버를 가리키는 경우가 있습니다.

**참고:** Cisco IOS 컨피그레이션을 사용하면 여러 mta **send server** 명령을 구성할 수 있습니다. 그러나 디바이스에 완전히 연결할 수 없는 한 첫 번째 디바이스만 사용됩니다. 따라서 mta **송신 서버**를 변경할 때 이전 서버를 제거해야 합니다.

Outlook 또는 Outlook Express가 있고 팩스 사서함을 모니터링하는 경우 DNS가 다음 간격으로 처리되기 전에 IP 팩스 서비스에 의해 전달됩니다.



메시지 본문에는 메시지 ID가 포함되어 있습니다. 이전에는 주제였습니다. 오류는 메시지 본문 아래쪽에 나타납니다. ISDN Q.931 오류 코드의 추상화입니다. 자세한 내용은 [부록](#)을 참조하십시오.

Exchange Message Tracking Center도 마찬가지로 유용합니다. Exchange 2000/2003 System

Manager에 내장되어 있습니다. 다음은 메일 메시지입니다.

The screenshot shows the Message Tracking Center interface. It includes search fields for Message ID, Sender, Server, and Recipients. The Server field is populated with 'vnt2-ex1.vnt.com'. The Logged Between section shows a date range from 10/14/2004 8:20:00 AM to 11:16:55 AM. On the right, there are buttons for 'Find Now', 'Stop', 'New Search', and 'Help'. Below the search area is a table with columns: Origination Time, Sender, Recipients, and Subject.

Origination Time	Sender	Recipients	Subject
10/14/2004 8:26 AM	Unity2 Subscriber	Fax unity2	[faxphone=9915550]
10/14/2004 8:27 AM	Fax unity2	FAX=9915550@vnt-3725-51.gateway.com	00000000377C58FB04A29C468DAB3CBC07D7F94B0700E7E65E238D0DEF46B5F8EAF983C6C51200000005E5B30000E7E65E238D0DEF46B5F8EAF983C6C5120000005FB680000]
10/14/2004 8:31 AM	<>	fax-unity2@vnt.com	Delivery Status Notification

### 6단계: DSN 메시지가 수신되었는지 확인합니다.

DSN 메시지가 IP 팩스 서비스의 사서함에 있으면 메시지를 처리하는지 확인합니다. IpFaxSvc 로그 파일이 도움이 됩니다.

```
"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"
메시지가 처리됩니다.
```

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

원본 제목은 메시지 본문에서 구문 분석됩니다. 이렇게 하면 데이터베이스에서 메시지를 찾을 수 있습니다. 원인 코드도 16으로 구문 분석됩니다. 이는 정상적인 통화 지우기입니다. 이러한 원인 코드는 Q.931 원인 코드에 해당합니다. 원인 코드에 대한 자세한 내용은 [Q.931 연결 끊기 원인 코드](#)를 참조하십시오.

항목이 데이터베이스에서 안전하게 제거되고 상태 0으로 다시 추가됩니다.

```
"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')]"
```

두 메일 메시지가 모두 삭제됩니다.아웃바운드 팩스 메시지를 확인할 때 아무 것도 없습니다.

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

## 주의 사항

현재 테스트된 버전에 있는 몇 가지 현재 주의 사항입니다. 이러한 문제를 해결하는 최신 릴리스는 [IP 팩스 구성 마법사](#) 또는 도움말 파일을 참조하십시오.

- Cisco 버그 ID [CSCef61241](#)([등록된](#) 고객만 해당) - 아웃바운드 IP 팩스는 PDF를 렌더링할 수 없습니다.
- Cisco 버그 ID [CSCef61256](#)([등록된](#) 고객만 해당) —아웃바운드 IP 팩스는 Exchange 5.5의 게이트웨이 DSN을 처리할 수 없습니다.
- Cisco 버그 ID [CSCef96036](#)([등록된](#) 고객만 해당) - 제목 형식이 잘못된 경우 IP 팩스 서비스가 NDR을 보내야 합니다.
- Cisco 버그 ID [CSCef96051](#)([등록된](#) 고객만 해당) —아웃바운드 IP 팩스 서비스 커버 페이지에서 콜백 번호를 허용해야 합니다.
- Cisco 버그 ID [CSCeg04158](#)([등록된](#) 고객만 해당) - IP 팩스 서비스는 알 수 없는 첨부 파일이 있는 메시지를 NDR해야 합니다.
- Cisco 버그 ID [CSCeg04180](#)([등록된](#) 고객만 해당) —IP 팩스 서비스는 서드파티 팩스 툴의 마지막 내선 번호를 인식하지 않습니다.
- Cisco 버그 ID [CSCeg39033](#)([등록된](#) 고객만 해당) —IP 팩스 서비스를 통해 이메일 메시지 본문을 렌더링할 수 없습니다.
- Cisco 버그 ID [CSCeg39046](#)([등록된](#) 고객만 해당) —IP 팩스 렌더링 엔진 및 로그 디렉토리는 항상 C: 드라이브.
- Cisco 버그 ID [CSCeg39100](#)([등록된](#) 고객만 해당) —IP 팩스 서비스는 여러 렌더링 실패 후 메시지를 NDR해야 합니다.
- Cisco 버그 ID [CSCeg45439](#)([등록된](#) 고객만 해당) - IP 팩스 서비스는 발신자에게 팩스 전송을 확인하는 것을 허용해야 합니다.
- Cisco 버그 ID [CSCeg45666](#)([등록된](#) 고객만 해당) —IP 팩스 서비스는 시작 시 애플리케이션 종속성을 확인해야 합니다.
- Cisco 버그 ID [CSCeg46458](#)([등록된](#) 고객만 해당) —IP 팩스 서비스를 중지하면 ODCDAEMON 프로세스가 중단됩니다.
- Cisco 버그 ID [CSCeg52736](#)([등록된](#) 고객만 해당) —아웃바운드 IP 팩스 서비스에서 MaximumWait 키가 작동하지 않습니다.

## 부록

## 렌더링 엔진 추적 파일

다음은 이 문서에서 설명한 예제의 추적입니다. 두 개의 파일이 있습니다. 한 파일은 데몬용 파일이고 다른 파일은 인쇄 드라이버용 파일입니다. 일반적으로 이러한 기능은 필요하지 않습니다. 그러나 각 샘플의 비교는 비교 목적으로 포함되어 있습니다. 렌더링 엔진에 문제를 제출하면 이러한 파일은 일반 IP 팩스 로그와 함께 필요합니다.

### ODCDAEMON.log

C:\ImgMaker\Conversion\_Toolkit\log\ODCDAEMON.log

이 파일은 자주 사용되며 읽기 쉽습니다. 입력 및 출력 파일은 별도의 섹션에 나열되고 응답 코드가 나열됩니다. 응답 코드가 양수이면 출력 TIF 파일에 렌더링되는 페이지 수를 나타냅니다. 음수 응답 코드는 오류를 나타냅니다. 프로세스를 실행할 사용자를 확인합니다. 이는 개별 응용 프로그램을 열어 TIF로 변환할 수 있어야 하는 사용자이므로 매우 중요합니다.

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

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 DocCnvr 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 DocCnvr 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 DocCnvr 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

## 드라이버 로그

C:\log\ImageMaker DocCnvr Driver.log

이 추적은 렌더링 엔진과 관련된 문제를 Cisco 기술 지원부에 제출할 때 필요합니다.

----->  
Log session started: 04-10-14, 08:27:19 AM  
Platform: WinNT 4.0  
Release Compilation  
pszPrinterName ImageMaker DocCnvr 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 DocCnvr Driver.FWD"  
to transfer data (printdoc.dll -> printdoc.exe dialog)  
data flushed!  
MS: 18808737  
deleted file: "C:\ImgMaker\Conversion\_Toolkit\ImageMaker DocCnvr Driver.BWD"  
used to transfer data from dialog back to here!  
MS: 18808737  
  
Leaving pdcf\_WriteInfoForControlDialog()  
  
verb:[printto] params:["ImageMaker DocCnvr 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 DocCnvrt Driver
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 DocCnvrt Driver.FWD"
to transfer data (printdoc.dll -> printdoc.exe dialog)
data flushed!
MS: 18810518
deleted file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrt Driver.BWD"
used to transfer data from dialog back to here!
MS: 18810518

Leaving pdcf_WriteInfoForControlDialog()

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

Calling ShellExecuteEx()
MS: 18810518
ShellExecute() succeeded

Calling pdcf_WaitForDoneFromControlDialog()
MS: 18810561
```

```
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: 18811581
!DONE: timeout = 1798
reading file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrt Driver.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 DocCnvrt Driver
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 DocCnvrt Driver.FWD"
to transfer data (printdoc.dll -> printdoc.exe dialog)
data flushed!
MS: 18812890
deleted file: "C:\ImgMaker\Conversion_Toolkit\ImageMaker DocCnvrt Driver.BWD"
used to transfer data from dialog back to here!
MS: 18812890

Leaving pdcf_WriteInfoForControlDialog()

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

Calling ShellExecuteEx()
MS: 18812894
ShellExecute() succeeded
```

Calling pdcf\_WaitForDoneFromControlDialog()  
MS: 18812897

Entering pdcf\_WaitForDoneFromControlDialog()  
Monitoring cancel event: ImageMaker DocCnvr Driver.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 DocCnvr Driver.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 DocCnvr Driver

**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 DocCnvr Driver.FWD"  
to transfer data (printdoc.dll -> printdoc.exe dialog)  
data flushed!

MS: 18815632

deleted file: "C:\ImgMaker\Conversion\_Toolkit\ImageMaker DocCnvr Driver.BWD"  
used to transfer data from dialog back to here!

MS: 18815632

Leaving pdcf\_WriteInfoForControlDialog()

verb:[printto] params:["ImageMaker DocCnvr Driver" "winspool" "Ne00:"]  
lpFile:[C:\CommServer\unityMta\IpFaxScratchpad\render.tif]

Calling ShellExecuteEx()  
MS: 18815632

```

ShellExecute() 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 연결 해제 원인 코드

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

## 관련 정보

- [음성 기술 지원](#)
- [음성 및 통합 커뮤니케이션 제품 지원](#)
- [Cisco IP 텔레포니 문제 해결](#)
- [기술 지원 및 문서 - Cisco Systems](#)