

# UCCX의 JTAPI 아키텍처 및 통화 흐름 이해

## 목차

---

### [소개](#)

### [사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

### [배경 정보](#)

[JTAPI 개요](#)

[JTAPI 및 UCCX](#)

[JTAPI 아키텍처](#)

[JTAPI 관찰자 모델](#)

[JTAPI 공급자 모델](#)

[JTAPI 사용자](#)

[P1 및 P2 핸들](#)

[통화 흐름](#)

### [문제 해결](#)

[JTAPI 로그 활성화 및 수집](#)

[JTAPI 디버깅 사용](#)

[JTAPI 디버그 수집](#)

[RTMT에서](#)

[CLI에서](#)

[JTAPI 로그 위치](#)

### [참조 링크](#)

---

## 소개

이 문서에서는 JTAPI의 기본 기능, 아키텍처 및 UCCX와 관련된 통화 흐름에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

Cisco에서는 다음 톨 및 기능에 대한 지식을 권장합니다.

- JTAPI - Java 텔레포니 API
- API - 애플리케이션 프로그래밍 인터페이스
- UCCX - Unified Contact Center Express
- CUCM - Cisco Unified Communications Manager

- CTI - 컴퓨터 텔레포니 통합

Cisco에서는 다음 항목에 대한 지식을 권장합니다.

- Cisco Unified Communications Manager(CUCM) 컨피그레이션
- UCCX(Unified Contact Center Express) 컨피그레이션

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 버전을 기반으로 합니다.

- UCCX 버전 12.5.1.11002-481
- CUCM 버전 12.5.1.11900-146

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

배경 정보

이 문서에서는 JTAPI 아키텍처, 통화 흐름, 디버깅 활성화 및 로그 수집에 대해 설명합니다.

JTAPI 개요

Cisco Unified JTAPI는 Sun Microsystems에서 Java 기반 컴퓨터 전화 응용 프로그램과 함께 사용하기 위해 개발한 프로그래밍 인터페이스 표준으로 사용됩니다. Cisco JTAPI는 추가 Cisco 확장과 함께 Sun JTAPI 1.2 사양을 구현합니다. Cisco JTAPI를 사용하여 다음과 같은 애플리케이션을 개발해야 합니다.

- Cisco Unified Communications Manager 폰을 제어하고 확인합니다.
- CTI(Computer-Telephony Integration) 포트 및 경로 포인트(가상 디바이스)를 사용하여 통화를 라우팅합니다.

JTAPI 및 UCCX

컨택 센터에서 Cisco Unified JTAPI를 사용하여 디바이스 상태를 모니터링하고 라우팅 지침을 실행하여 적시에 적소에 통화를 전송합니다. 또한 JTAPI는 분석을 위해 통화 통계를 검색하는 동안 명령 기록을 시작 및 중지하고 CRM 애플리케이션으로의 화면 팝업 통화, 자동화된 스크립팅, 원격 통화 제어 등을 수행하는 데 사용됩니다.

## JTAPI 아키텍처

엔터프라이즈 환경에서 사용되는 Cisco Unified JTAPI는 프레즌스 기반 라우팅을 위해 사용자 가용성, 위치 및 환경 설정을 고유한 맞춤형 환경으로 통합합니다.

다음은 JTAPI의 애플리케이션입니다.

- JTAPI는 둘 이상의 전화 및 회선 조합을 모니터링하거나 알림을 받을 수 있습니다.
- 컨택 센터 애플리케이션은 JTAPI에 대해 전체 통화 모델을 사용합니다.
- JTAPI는 다음 기능을 제공합니다.
  - 통화 제어
  - 미디어 컨트롤
  - 미디어 협상

## JTAPI 관찰자 모델

다음 다이어그램에서는 JTAPI가 작동하는 제공자-관찰자 모델에 대해 설명합니다.

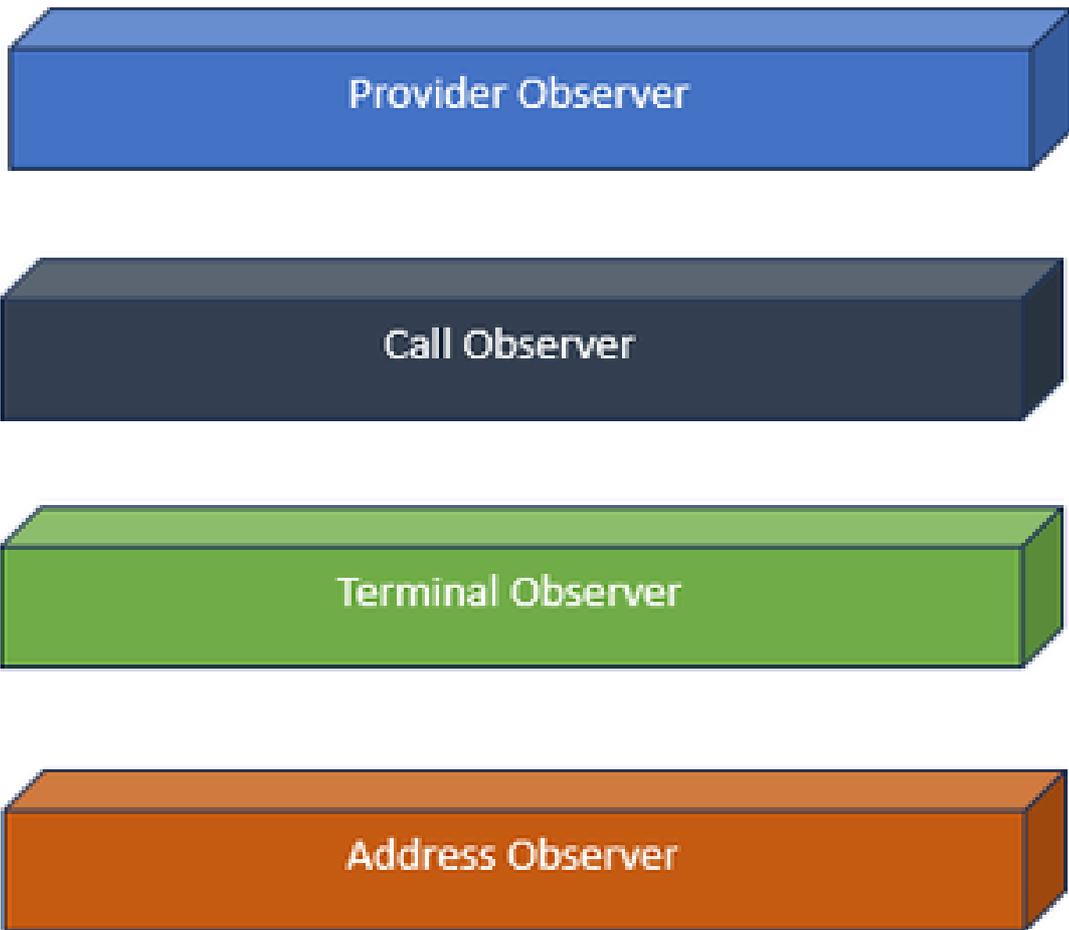
Provider

Call

Terminal

Address

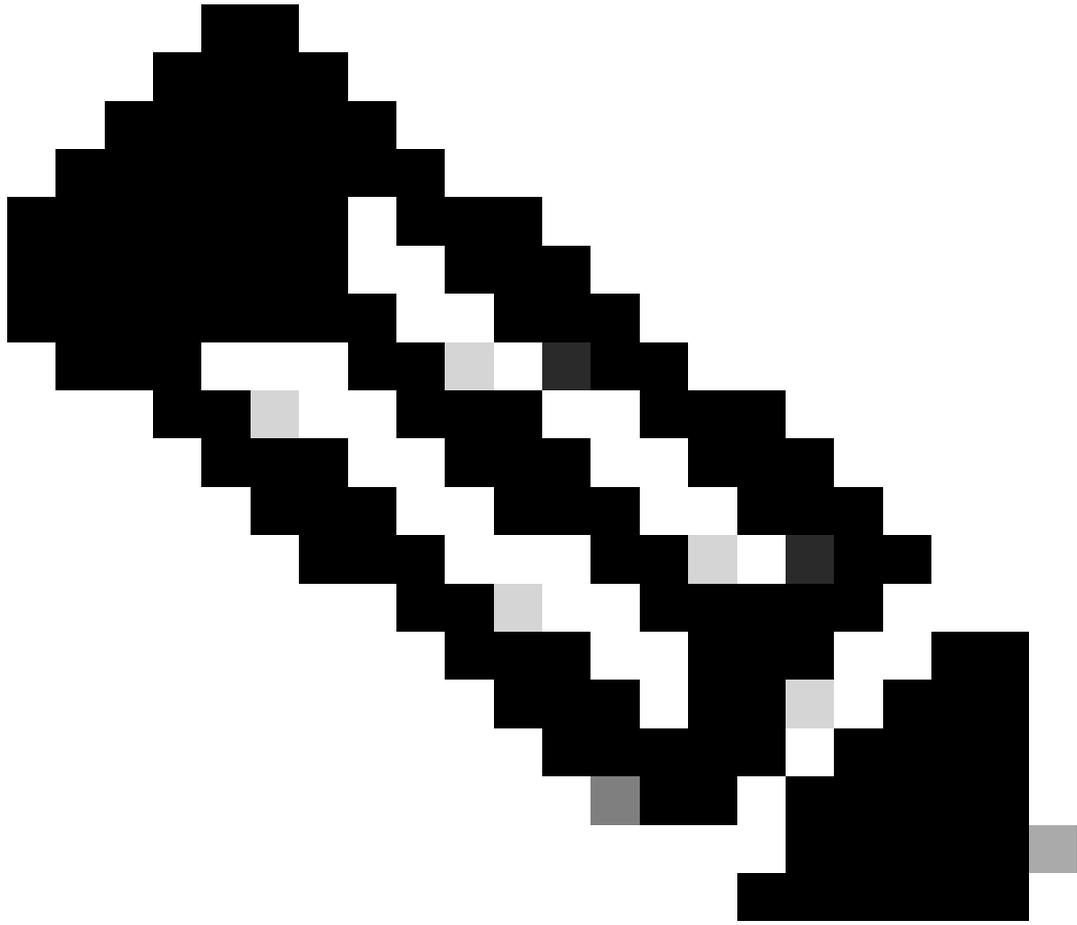
관찰자



#### 관찰자 인터페이스

JTAPI는 관찰자의 아이디어를 바탕으로 합니다. 관찰자에 의해, 그것은 사건을 관찰하는 사람의 생각을 언급한다. 시스템 내에서 여러 관찰자를 서로 다른 대상에 배치할 수 있습니다. JTAPI는 관찰자를 사용하여 객체의 상태 변화에 대해 알 수 있습니다.

예를 들어 회선, 전화, 터미널, 주소 등에 관찰자를 두고 상태 변경에 대해 알아볼 수 있습니다.

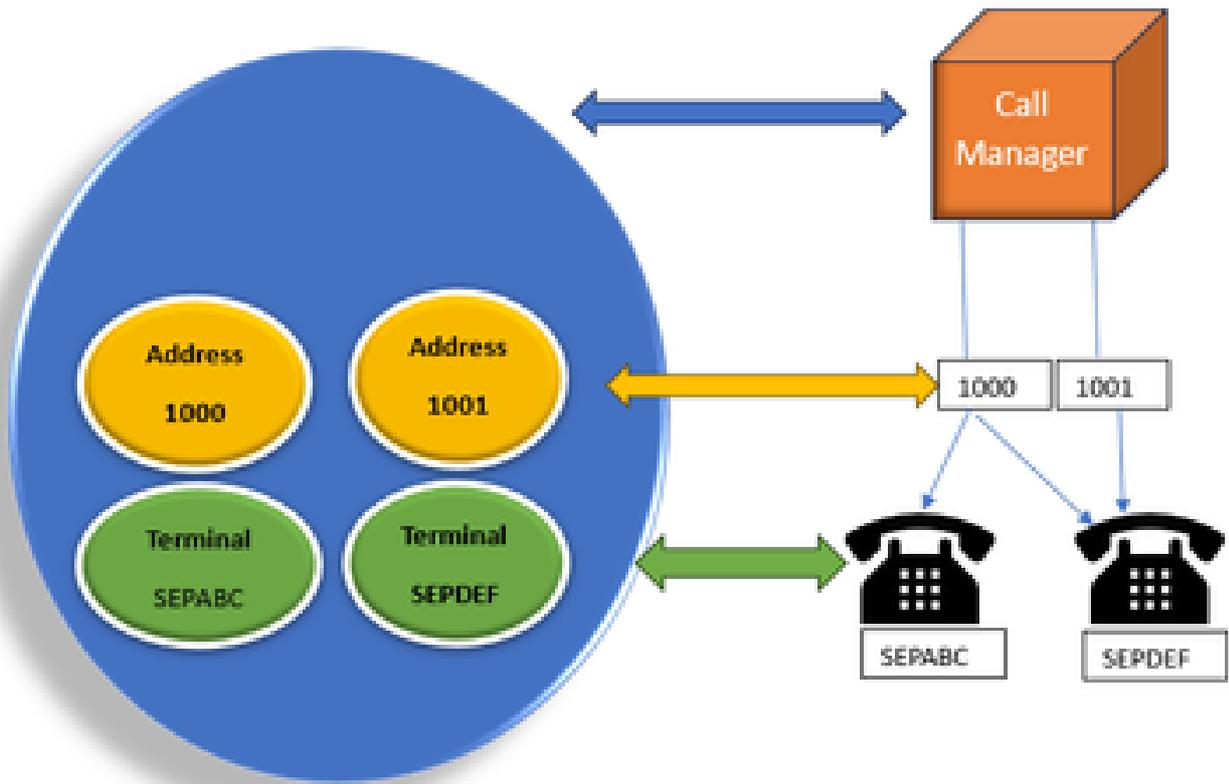


**참고:** 객체에 관찰자를 배치하지 않은 경우 관찰자에 대해 수행된 작업에 대한 알림을 받을 수 없습니다.

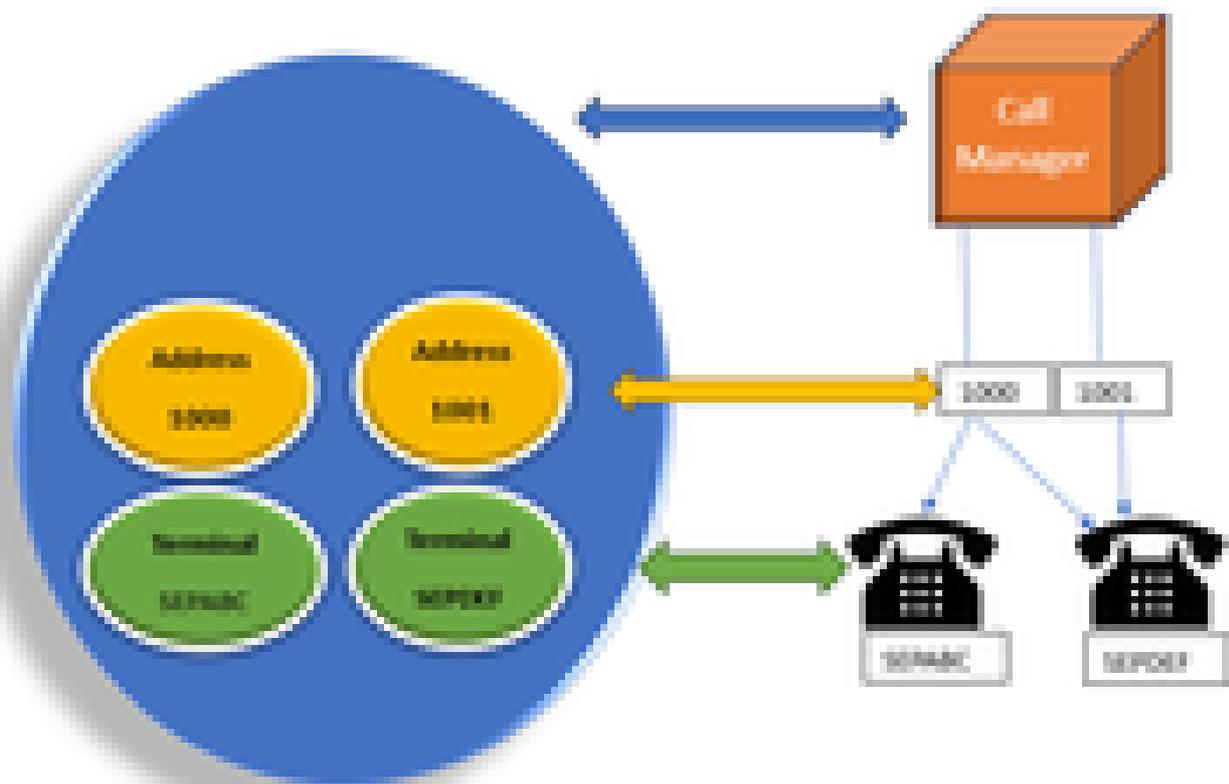
---

## JTAPI 공급자 모델

다음 다이어그램에서는 JTAPI가 작동하는 공급자 모델에 대해 설명합니다.



JTAPI 공급자

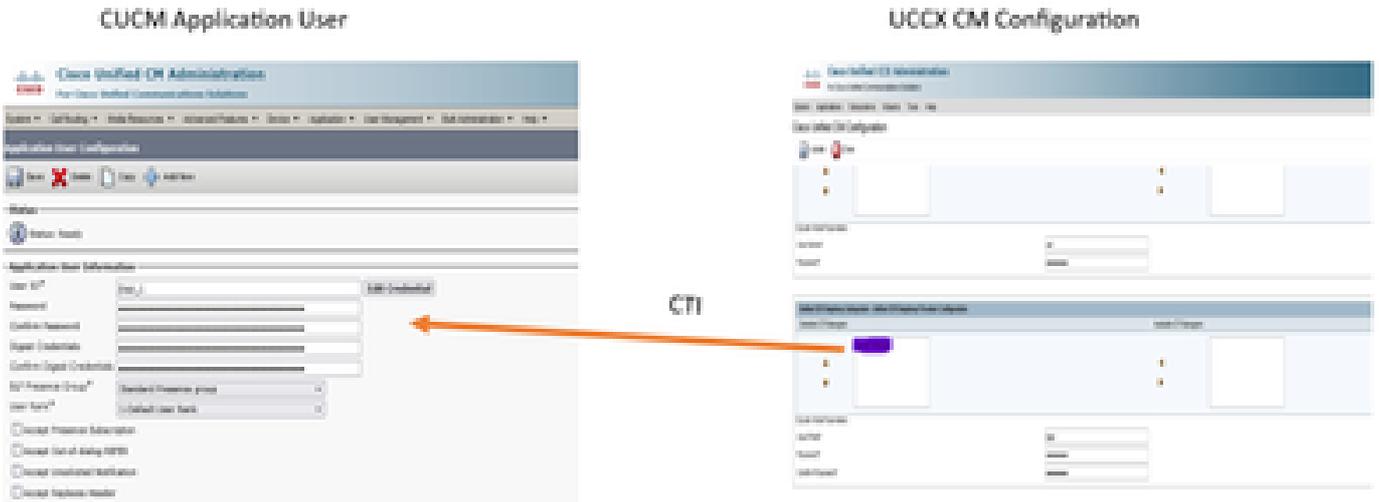


모델공급자 모델

JTAPI 제공자는 애플리케이션에서 Call Manager 또는 CTI Manager로의 연결입니다. 제공자는 객체에 관찰자를 연결하는 데 사용됩니다. 제공자에 관찰자를 연결할 수도 있습니다. 제공자는 객체에 대해 수행된 작업에 대한 알림을 받는 데 사용됩니다. (관찰자가 어태치된 디바이스를 해당 관찰자를 어태치한 제공자로부터 제어할 수도 있습니다.)

### JTAPI 사용자

다음 스크린샷은 JTAPI 애플리케이션 사용자에 대해 설명하는 CUCM(왼쪽) 및 UCCX(오른쪽)에서 찍은 것입니다.



- 응용 프로그램을 시작하고 CTI 관리자에 연결하려는 경우 공급자를 엽니다.
- 제공 기관을 여는 이유는 장치, 터미널 등에서 수행되는 작업을 모니터링할 수 있기 때문입니다.
- CUCM에서 구현되는 방식은 애플리케이션 사용자를 통해 이루어집니다.
- 먼저 CTI를 통해 CUCM에 인증할 수 있도록 이 사용자를 생성하고 자격 증명을 제공합니다.
- 따라서 CUCM에서 생성된 JTAPI 애플리케이션 사용자는 제공자입니다.
- 모니터링만 하는 것이 아니라, 디바이스를 제어할 수 있는지 확인하려면 해당 **디바이스**가 Associated Devices 아래에 있는지 확인해야 합니다.

---

**참고:** JTAPI 사용자는 cucm에서 생성하지 않습니다. JTAPI 사용자는 CUCM에서 AXL을 통해 애플리케이션(UCCX)에서 생성됩니다.

---

#### P1 및 P2 핸들

P1 및 P2는 제공자 핸들입니다. 이는 동일한 애플리케이션에서 여러 제공자를 열 때 중요합니다. UCCX에서 2개의 공급자를 생성합니다. 그중 하나는 CTI 포트 및 경로 포인트를 제어하고 다른 하나는 RM-JTAPI 공급자라고 하는 에이전트 폰을 제어합니다. UCCX에서 CTI 포트 및 경로 포인트를 제어하는 제공자를 생성할 때 먼저 JTAPI P1 제공자가 됩니다. P1 뒤에 생성하는 다른 제공자는 에이전트 디바이스를 처리하는 P2 제공자 또는 RmCm 제공자입니다.

P1 새 통화 이벤트가 표시되면 CTI 포트 또는 CTI 경로 포인트의 통화 이벤트임을 알 수 있습니다. P2 새 통화 이벤트가 표시되면 실제 전화기의 통화 이벤트임을 알 수 있습니다. CCX 측에서 RM-JTAPI 사용자 1명과 JTAPI 사용자 1명을 생성하지만 CUCM 측에서

2명의 JTAPI 사용자가 생성되었습니다. 이는 각 CCX 노드가 자체 JTAPI 사용자를 갖지만 하나의 RM-JTAPI 사용자를 공유하기 때문입니다. UCCX에서 트리거를 생성하면 두 JTAPI 사용자 모두에 추가됩니다. 두 노드 모두 제공자를 개별적으로 엽니다. 따라서 경로 포인트에서 수행되는 모든 작업은 두 CCX 노드에 모두 알림을 받습니다.

그 외에 보조 노드는 그냥 거기 앉아서 그것이 여전히 보조 노드라는 것을 계속 알립니다. 각 노드에는 별도의 CTI 포트 그룹이 있습니다. 노드 1의 CTI 포트는 jtapi\_1에 의해 제어됩니다. 노드 2의 CTI 포트는 jtapi\_2에 의해 제어됩니다.

통화가 CTI 경로 포인트에 도착하면 두 CCX 노드 모두 새 통화 이벤트에 대한 알림을 받지만 활성 CCX 노드는 자신이 제어하는 CTI 포트 중 하나에 대한 리디렉션 요청과 함께 Call Manager에 응답합니다.

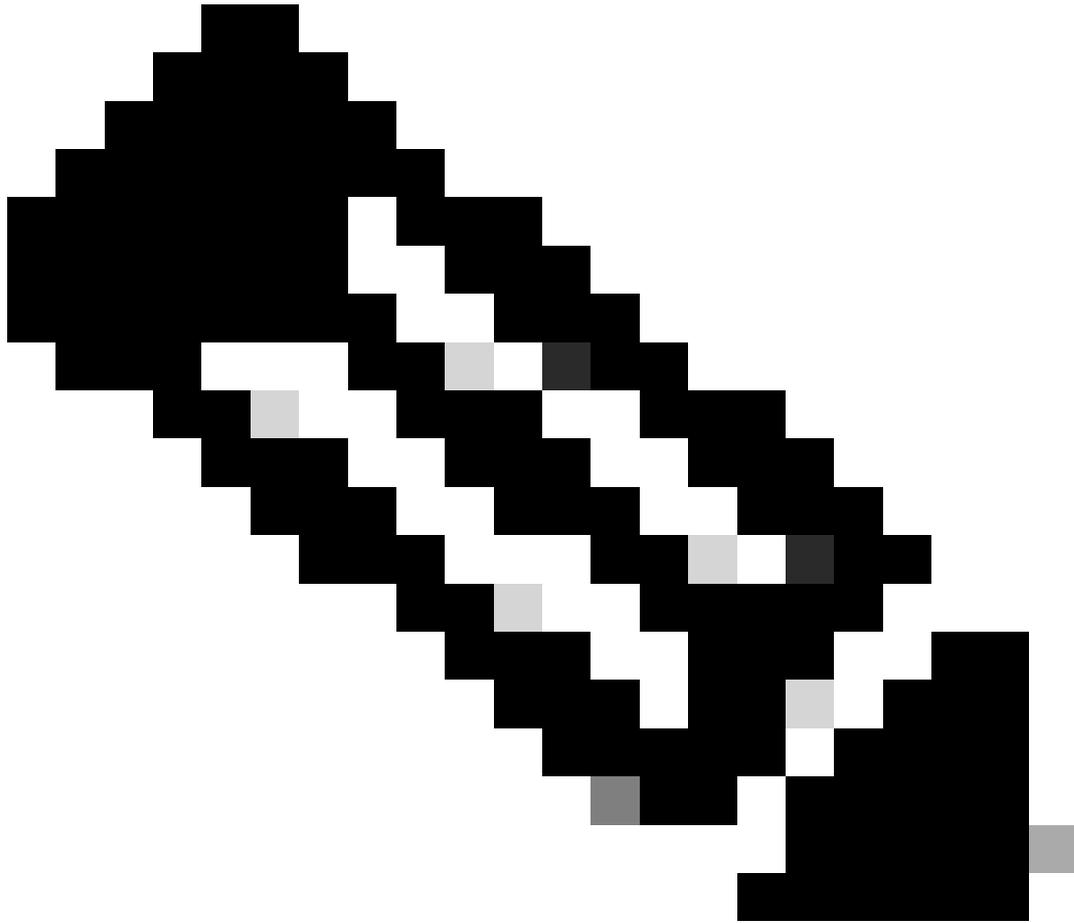
CUCM에서 CTI 경로 포인트에 대한 IP가 표시되는 경우 CCX의 IP 중 하나이지만, 특정 노드가 통화를 라우팅한다는 의미는 아닙니다.

한 가지 일반적인 방법은 JTAPI 사용자로부터 디바이스의 연결을 해제하고 다시 추가하는 것입니다. 그 이유는 연결을 해제하면 제공자에게 알림이 전달되고 제공자에 대한 모든 연결이 제거되며, 다시 추가되면 관찰자가 다시 추가되고 제공자는 열려 있는 제공자 요청을 사용하여 다시 모니터링하기 시작합니다. 제어 대상 디바이스 목록에 추가된 디바이스 외에도 개별 디바이스에서 CTI에서 디바이스 제어 허용 확인란이 있습니다. 이는 추가적인 수준의 세분화를 제공하기 위한 것입니다. 따라서 제어 목록에 경로 포인트가 추가되었지만 CTI 확인란이 선택되지 않은 경우 해당 경로 포인트의 이벤트에 대한 알림을 받을 수 있지만 통화 제어에 대한 작업은 수행할 수 없습니다.

## 통화 흐름

다음은 UCCX 통화 흐름과 관련된 이벤트의 시퀀스입니다.

- 통화가 CTI 경로 포인트에 도착하면 CTI 포트에 리디렉션됩니다.
- CTI 포트는 uccx 상자의 ipvms 드라이버를 사용하여 미디어 채널을 엽니다.
- 상담원이 통화를 받아야 한다고 결정하면 상담 호전환이 포트에서 상담원에게 수행됩니다.
- 새 통화 이벤트가 CTI 경로 포인트로 전송됩니다.
- 리디렉션 요청이 CTI 포트에 이동합니다.
- 통화 ID의 상태가 유효 상태가 됩니다.
- 그러면 CTI 포트에 통화에 대해 또 다른 새 통화 이벤트가 발생합니다.
- 리디렉션 응답이 0이면 양호하고, 0이 아니면 실패했음을 의미합니다.
- 그런 다음 통화 수락 요청을 보냅니다(응답되지 않음, 포트에서 수락됨).
- 그런 다음 스크립트에서 단계 적중을 수락합니다. Jtapi에서 통화 응답 요청이 올 때입니다.

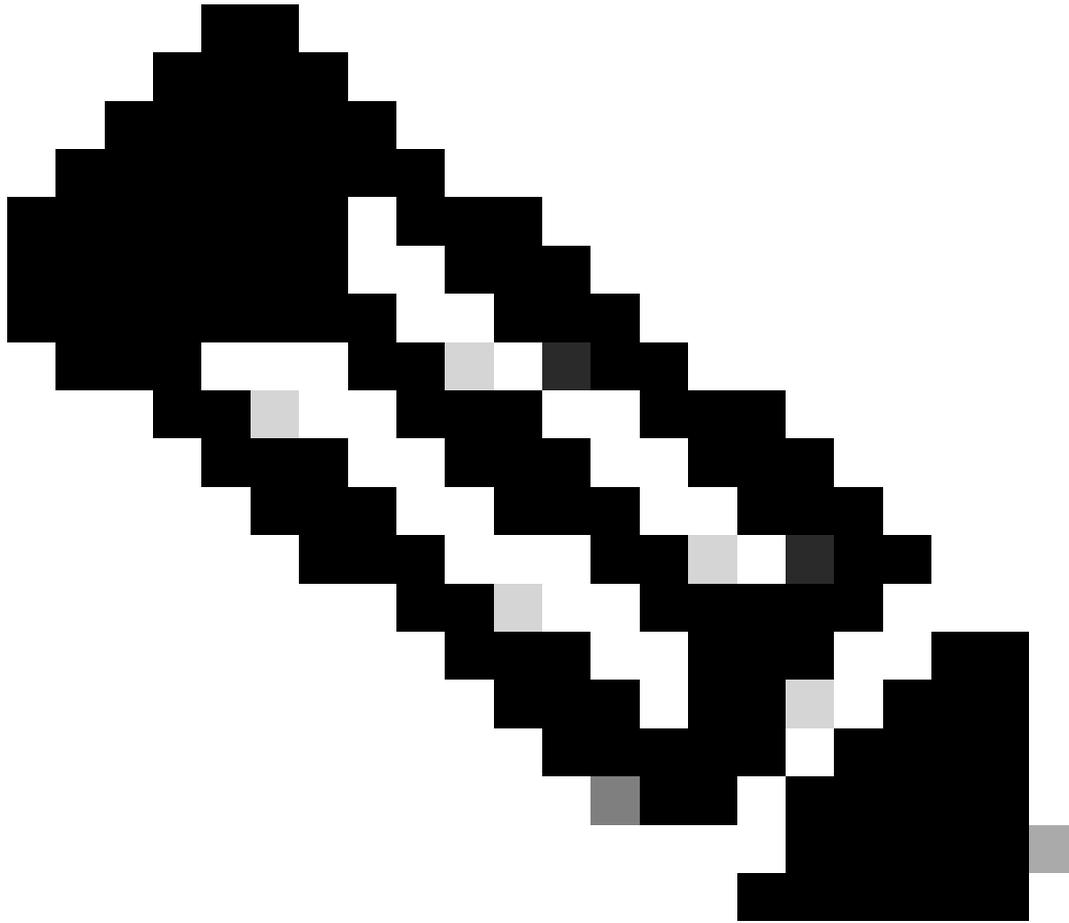


**참고:** 이 상황은 매우 빠르게 발생하며, Cisco Unity Connection에서 걸려오는 전화 또는 CUCM에서 시간이 걸리는 호전환과 같이 여러 이벤트가 동시에 발생하는 경우도 있습니다. 이 경우 수락 단계에서 RACE 조건이 발생할 수 있습니다. 따라서 이 속도를 늦추고자 합니다. 승인 단계 전에 지연 단계를 추가하여 이 작업을 수행합니다.

---

11. 그런 다음 포트에서 응답 메시지를 받습니다.

12. 통화 상태가 연결됨으로 변경됩니다.

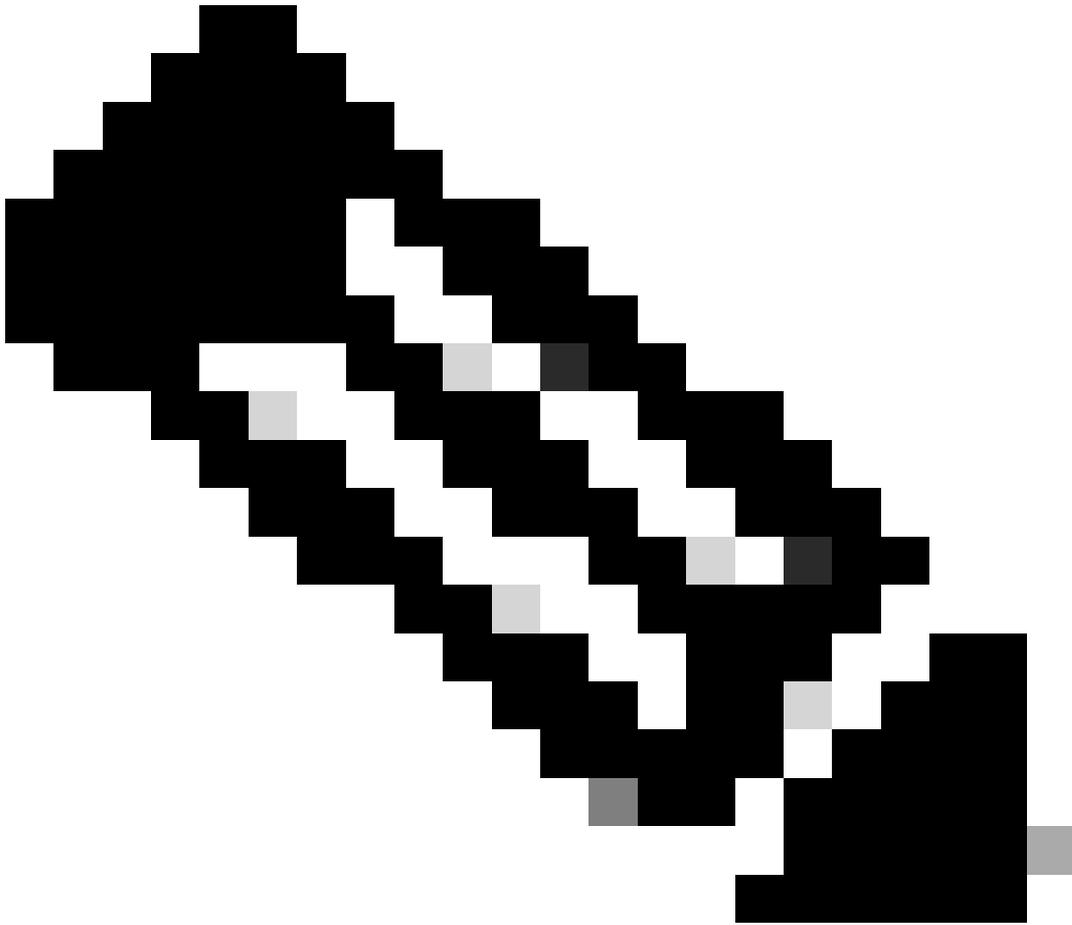


**참고:** CTI는 새 요청을 보내기 전에 응답을 기다리는 sip 또는 skinny 전화와 달리 비동기적이므로 JTAPI CTI 메시지의 메시지 순서가 뒤바뀔 수 있습니다.

- 
13. 포트에서 응답 받으면 매체 협상이 이루어집니다.
  14. 포트가 open\_logical\_channel 요청을 전송하고, 여기서 원격 상대방이 RTP를 전송할 포트와 ip를 공유합니다.
  15. 논리 채널을 열기 전에 UCCX 상자의 IPVMS 드라이버에 연결을 만들고 RTP 스트림을 엽니다.
  16. 다음 이벤트는 원단 정보(즉, 발신 장치의 ip 및 포트)를 알려주는 start\_reception 이벤트입니다.
  17. 그런 다음 다른 미디어 신호와 마찬가지로 start\_transmission 이벤트를 가져옵니다.

18. 이제 IVR과 프롬프트를 듣고 있습니다.

---



**참고:** 여기서의 통화 설정이 완료됩니다.

---

19. 이제 상담원에게 통화를 호전환할 수 있는 스크립트의 단계에 도달하면 CallSetupTransferRequest가 표시됩니다.

20. 첫 번째 메시지와 달리 이 메시지는 상담 호전환이며 리디렉션이 아닙니다.

21. 상담 호전환인 이유는 상담원이 준비되었지만 자리에 있지 않은 경우 통화를 재전송하면 응답하지 않은 상태로 남아 있지만 상담 호전환을 하면 상담원이 없는 경우 통화가 다시 대기열에 배치되기 때문입니다.

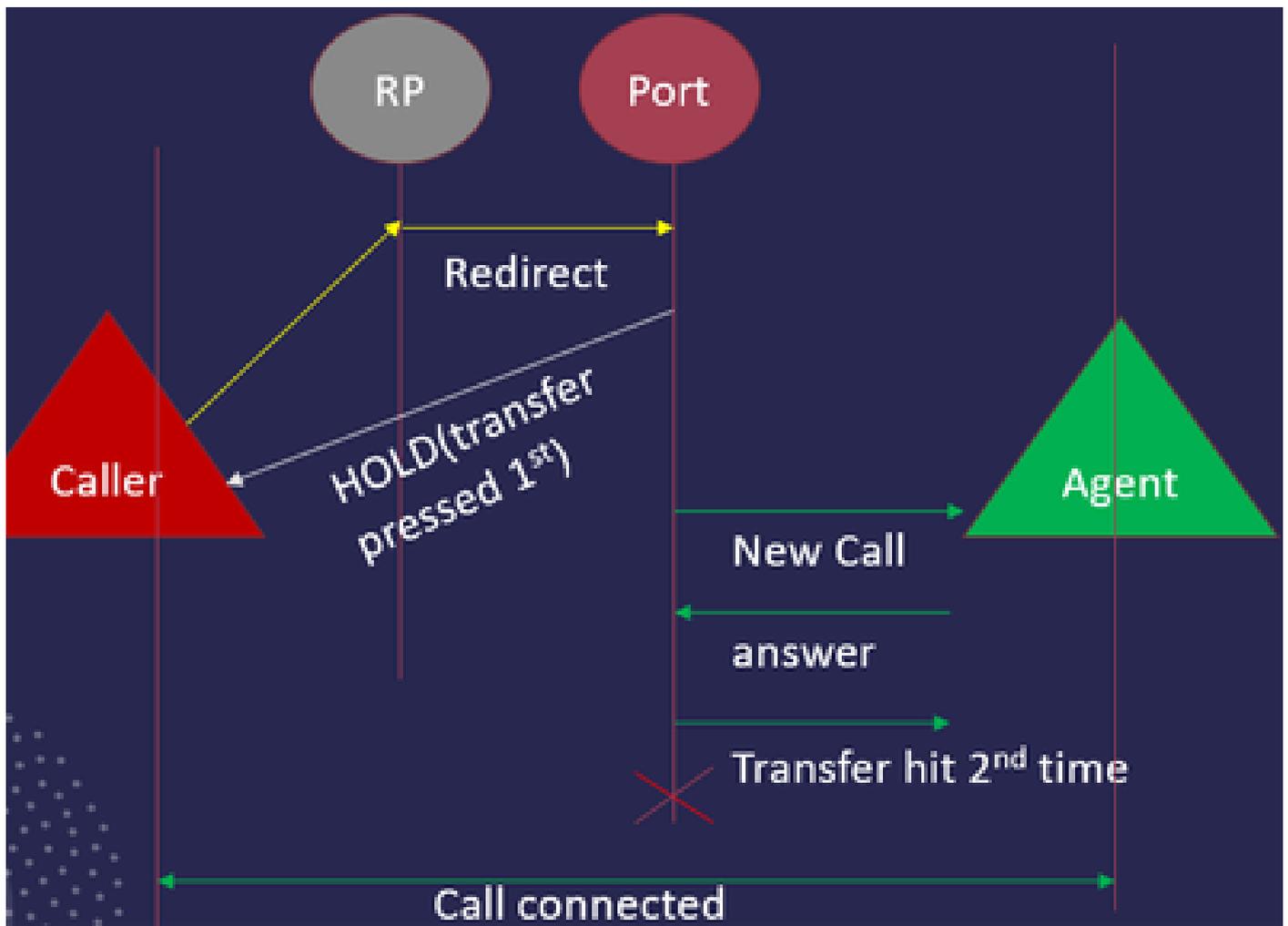
22. 다른 상담 호전환과 마찬가지로, 이것은 CTI 포트로서 전화기에서 호전환 버튼을 처음 누르는 것입니다.

---

**참고:** ConsultWithoutMedia 상담 호전환을 위한 API입니다.

- 
23. 일반 상담 호전환에서는 A와 C 사이에 미디어 경로가 있지만 UCCX와 에이전트 간에 미디어를 설정한다는 의미가 없으므로 이 작업을 수행하지 않도록 CUCM에 지시합니다.
  24. 따라서 상담원과 CTI 포트 간에 미디어 컷오버를 수행하지 않고 상담 호전환을 수행합니다.
  25. 이 시점에서 CTI 포트는 발신자를 보류 상태로 설정하고 통화 상태가 변경된 event=HOLD를 확인합니다.
  26. 이제 CTI 포트에서 에이전트 디바이스로의 새 통화 이벤트가 표시됩니다.(원래 통화 ID를 사용하되 하위 집합 및 P2 이벤트를 사용합니다.)

27. P2 이벤트 통화 id를 검색하는 경우 다음 메시지가 CallAnswer 요청으로 표시됩니다. 이는 상담원이 통화를 수신했음을 의미합니다.
  28. 상담원이 통화를 수신(P2 사용)하고 통화가 CTI 포트측에서도 연결된 상태(P1 사용)인 것을 알게 되면 경로 포인트에 CallDirectTransferRequest 전송 버튼이 두 번째로 눌러졌다는 의미인 가 표시됩니다.
  29. 이제 CTI 포트는 에이전트가 통화를 수신했음을 알므로 대기 지점이 없으므로 앞서 설명한 대로 즉시CallDirectTransferRequest CTI 포트인 CTI 포트가 전송 버튼을 두 번째로 누릅니다.
  30. 이제 원래 통화 구간(대기 중인 통화 구간)이 남아 있는 통화 구간입니다.
  31. 다른 통화 레그가 파괴됩니다(포트와 에이전트 간).
  32. 이 시점에서 CUCM 및 UCCX는 정상화되며 RTP는 게이트웨이를 통해 발신자와 상담원 간에 설정됩니다.
- 다음 다이어그램은 앞서 언급한 통화 흐름을 요약하여 설명합니다.



JTAPI 통화 흐름 요약

문제 해결

JTAPI 로그 활성화 및 수집

## JTAPI 디버깅 사용

JTAPI 디버그 레벨을 활성화하려면 다음 단계를 확인하십시오.

- UCCX에 로그인합니다.
- CCX Serviceability(CCX 서비스 가용성)로 이동합니다.
- Trace(추적)로 이동합니다.
- 구성으로 이동합니다.
- 서비스 선택 드롭다운에서 Cisco Unified CM Telephony Client를 선택합니다.
- 디버깅 확인란을 선택합니다.
- MISC\_DEBUGGING을 제외한 모든 확인란을 선택합니다.

## JTAPI 디버그 수집

RTMT 또는 CLI에서 JTAPI 디버그 레벨을 활성화하려면 다음 단계를 확인하십시오.

### RTMT에서

- CCX 실시간 모니터링 톨에 로그인합니다.
- **Trace & Log Central**을 클릭합니다.
- Collect Files(파일 수집)를 클릭합니다.
- 모든 서버에 대해 JTAPI 클라이언트를 선택합니다.
- **Next(다음)**를 클릭합니다.
- 로그 파일을 저장할 타임스탬프와 위치를 선택합니다.

### CLI에서

#### JTAPI 로그 위치

```
activelog /uccx/log/JTAPI
```

JTAPI 로그를 수집하는 명령:

```
file get activelog /uccx/log/JTAPI/* recurs compress
```

## 구문:

file get { activelog|inactivelog|install } file-spec [options]

file-spec 전송할 필수 파일

옵션 선택적 reltime months|weeks|days|hours|minutes timevalue

요약 시간 hh:mm:MM/DD/YY HH:mm:MM/DD/YY

regex 일치

재귀

압착기

타임스탬프를 기반으로 로그를 다운로드하는 5가지 방법

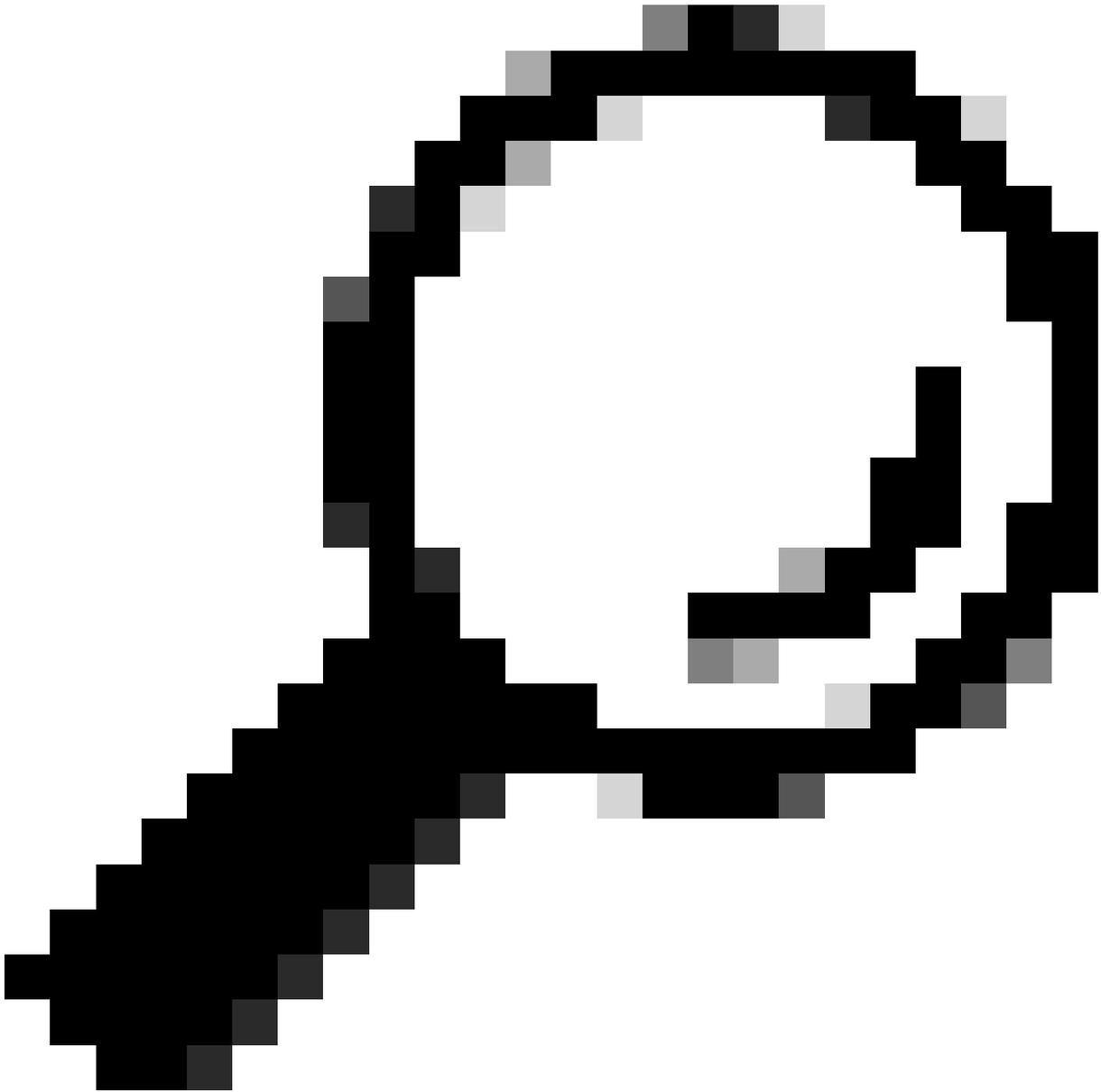
**reltime** - 분 단위로 지정된 상대 기간 | 시간 | 일 | 주 | 월 값

**abstime** - hh:mm:MM/DD/YY hh:mm:MM/DD/YY로 지정된 절대 기간입니다.

**match** - 파일 이름에서 문자열 값으로 지정된 특정 문자열을 확인합니다.

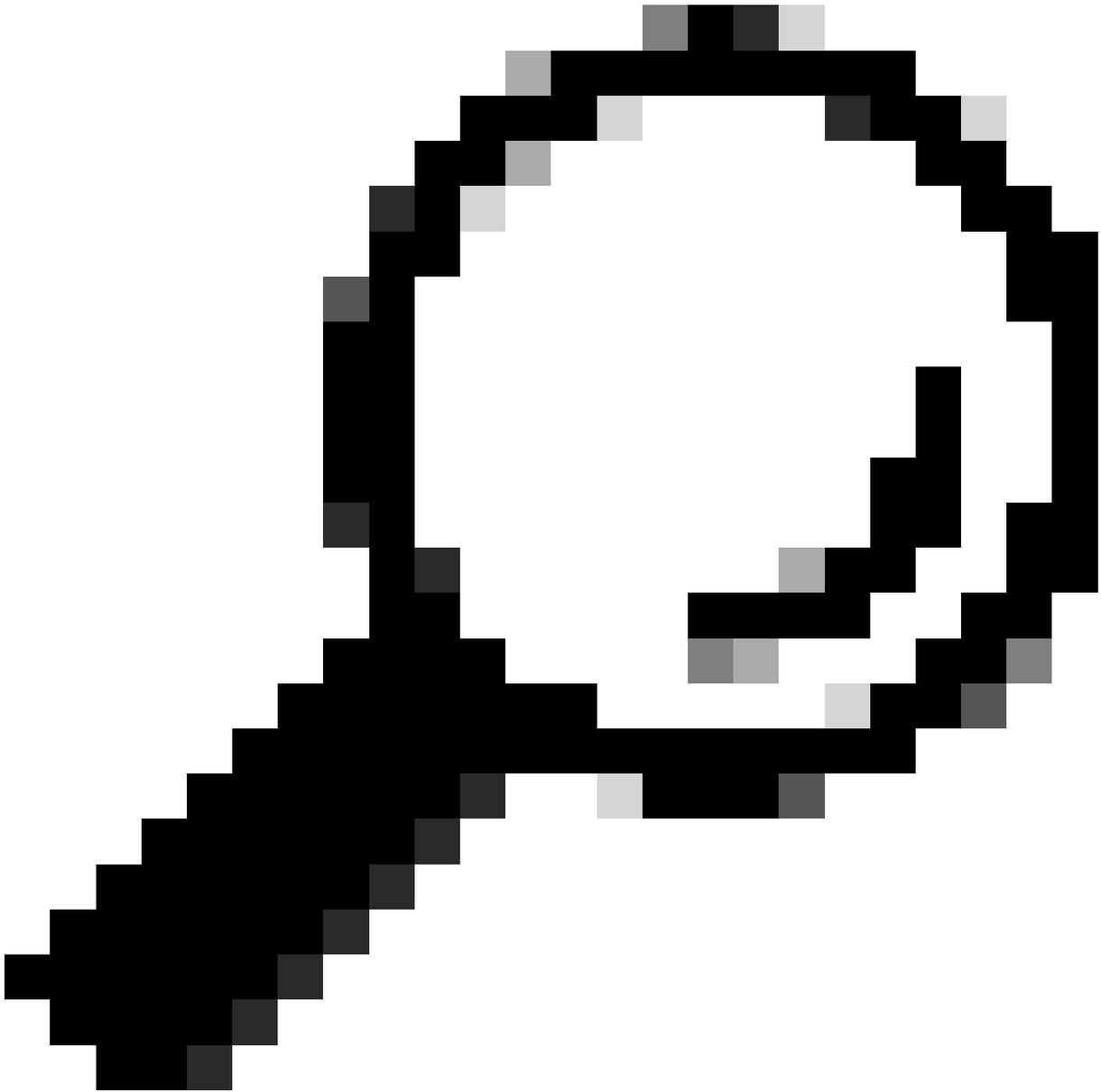
**recurs** - 모든 파일을 가져오고 하위 디렉터리를 포함합니다.

**compress** 옵션을 사용하면 압축된 형식으로 파일을 다운로드할 수 있습니다.



팁: 파일을 다운로드하려면 외부 SFTP(Secure File Transfer Protocol) 서버가 구성되어 있고 액세스할 수 있는지 확인하십시오.

---



팁: `recur` 옵션을 사용하면 모든 하위 디렉토리 및 파일의 디렉토리를 이동할 수 있습니다. 이는 디렉토리에서 모든 로그를 가져오려는 경우에 사용됩니다.

---

#### 참조 링크

- [JTAPI 개발자 가이드](#)

- [UCCX 추적 레벨](#)

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.