

Часто задаваемые вопросы для Cisco MediaSense

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[1. Как вы коррелируете ссылочные идентификаторы вызова для других сценариев вызова при разветвлении Телефона диспетчера Унифицированной связи?](#)

[1.1. Поиск MediaSense и функция ассоциации вызова воспроизведения](#)

[1.2. Агент Держится/Возобновляет Сценарий](#)

[1.3. Клиент Держится/Возобновляет Сценарий](#)

[1.4. Передача агента в другой сценарий агента](#)

[1.5. Конференции агента с другим сценарием агента](#)

[2. Как мы коррелируем ссылочные идентификаторы вызова для других сеансов вызова при разветвлении Унифицированного элемента границы?](#)

[2.1. Середина изменения кодека вызова](#)

[2.2. Передачи консультации](#)

[2.3. Обнаружение консультационных вызовов](#)

[2.4. Обнаружение консультационных вызовов от Множественных Участников](#)

[2.5. Сводка](#)

[3. Как вы привязываете вызовы в Cisco MediaSense с их появлениями в других компонентах решения?](#)

[3.1. Таблица корреляции идентификатора](#)

[4. Как вы определяете, какая дорожка имеет вызывающую сторону и какая дорожка имеет вызываемую сторону?](#)

[4.1. Для вызовов, разветвленных CUBE](#)

[4.2. Для вызовов, разветвленных по унифицированным телефонам CM](#)

[5. Каковы возможные причины для состояния сеанса CLOSED_ERROR?](#)

[6. Каково различие между Сокращенными и Удаленными сеансами?](#)

[6.1. С Запросом getAllPrunedSessions](#)

[6.2. С Запросом getSessions](#)

[6.3. Почему различие в поведении в сокращенных и удаленных сеансах?](#)

[7. Как настроить TDM шлюз для разветвления сред?](#)

[8. Как перехватить фактический телефон получателя при использовании группы последовательного поиска?](#)

[9. Почему Менеджеру Унифицированной связи Сетевая Запись рекомендуют как предпочтительный механизм разветвления?](#)

[10. Почему узел занимает больше времени для обновления к MediaSense 10.5?](#)

[11. Каково влияние российских изменений часового пояса на приложении Поиска и Воспроизведения MediaSense?](#)

[12. Что языки поддерживаются MediaSense?](#)

[13. Как контролировать производительность системы MediaSense?](#)

[14. Как настроить браузер для выполнения проигрывателя в браузере в MediaSense?](#)

Введение

Этот документ описывает часто задаваемые вопросы для сервера Смысла Медиа Cisco.

Предварительные условия

Требования

Компания Cisco рекомендует предварительно ознакомиться со следующими предметами:

- Cisco MediaSense
- Cisco Unified Communication Manager (CUCM)

Используемые компоненты

Сведения, содержащиеся в данном документе, касаются следующих версий программного обеспечения и оборудования:

Cisco MediaSense 10.5

1. Как вы коррелируете ссылочные идентификаторы вызова для других сценариев вызова при разветвлении Телефона диспетчера Унифицированной связи?

В Cisco MediaSense метаданные для каждого вызова только предоставляют xRefCi (ссылочный идентификатор вызова) и устройство касательно (расширения) разветвляющегося устройства, и устройство на дальнем конце (может быть мост конференц-связи или любой другой телефон).

xRefCi параметр является идентификатором Менеджера Объединенных коммуникаций для потока particular medium. Они не всегда соответствуют 1:1 зарегистрированным дорожкам.

1.1. Поиск MediaSense и функция ассоциации вызова воспроизведения

MediaSense генерирует несколько сеансов для вызова, который зарегистрирован в случае, держатся/возобновляют или передают, который мешает определять все сеансы записи в вызове. Чтобы быть в состоянии привязать эти сеансы записи в одиночном вызове, MediaSense представляет новую характеристику, которую называют как Ассоциация Вызова. Через эту функцию группируются все строго связанные вызовы с общим значением xRefci. MediaSense 10.5 поддерживает функцию ассоциации вызова для Созданных в мосте записей.

1.2. Агент Держится/Возобновляет Сценарий

1. Агент (extn 1000) и абонент К (extn 2000) вызвал друг друга и в состоянии разговора.
2. Агент А помещает удержанный вызов.
3. Агент вызов резюме.

Существует два сеанса записи для этого сценария:

- Сеанс с открытым сеанс = S1 с этими двумя дорожками, для периода времени / сегмент, прежде чем агент помещает удержанный вызов:
trackNumber = 0 с участником (deviceRef = 1000, xRefCi = аааа)
trackNumber = 1 с участником Б (deviceRef = 2000, xRefCi = сссс)
- Сеанс с открытым сеанс = S2 с этими двумя дорожками, для периода времени / сегмент после того, как агент возобновляет вызов.
trackNumber = 0 с участником (deviceRef = 1000, xRefCi = аааа)
trackNumber = 1 с участником Б (deviceRef = 2000, xRefCi = сссс)

В то время как агент поместил удержанный вызов, MediaSense не делает запись сегмента вызова.

1.3. Клиент Держится/Возобновляет Сценарий

1. Агент (extn 1000) и абонент К (extn 2000) беспокоил друг друга и в состоянии разговора.
2. Клиент К помещает удержанный вызов.
3. Клиент К возобновляет вызов.

Весь вызов зарегистрирован на одном сеансе для этого сценария:

- Сеанс с открытым сеанс = S1 с этими двумя дорожками:
trackNumber = 0 с участником (deviceRef = 1000, xRefCi = аааа)
trackNumber = 1 с участником Б (deviceRef = 2000, xRefCi = сссс)

В то время как клиент поместил удержанный вызов, в этом сценарии MediaSense также делает запись сегмента вызова.

1.4. Передача агента в другой сценарий агента

1. Абонент К (extn 2000) вызывает агента (extn 1000)
2. Агент (extn 1000) консультируется с агентом Б (3000)
3. Агент (extn 1000) завершает передачу.
4. Агент Б (extn 3000) зависает.

С Менеджером Унифицированной связи 9.x и ранее, вот результаты:

1. Абонент К (extn 2000) вызывает Агента (extn 1000)

S1 сеанса ЗАПУСТИЛСЯ, Дорожка 0 (extn 1000), Дорожка 1 является С (extn 2000).

2. Агент (extn 1000) передает вызов другому агенту Б (extn 3000). И А и устройства В установлены для разветвления.

3. Абонент К (extn 2000) слышит музыку в режиме удержания (MoH).
4. Агент (extn 1000) говорит с В (extn 3000).

- S1 сеанса ЗАКОНЧИЛСЯ
- Откройте сеанс ЗАПУЩЕННЫЙ S2, Дорожка 0 (extn 1000), Дорожка 1 является В (extn 3000)
- S3 сеанса ЗАПУСТИЛСЯ, Дорожка 0 является В (extn 3000), Дорожка 1 (extn 1000)

Факторы:

- S1 сеанса заканчивается из-за Агента, телефон разместил Абонента К в ожидании.
- Открывает сеанс S2, и S3 существуют, потому что оба телефона установлены настройки для разветвления.
- Участники и xRefCi для этих двух участников S2 и S3 идентичны, но обратных позиций друг от друга.
- значения xRefCi S1 не отражены или в S2 или в S3, так как консультирование считают независимым вызовом.

5. Агент (extn 1000) завершает передачу.

6. С (extn 2000) говорящий В (extn 3000).

7. (extn 1000) разъединен.

Откройте сеанс ЗАКОНЧЕННЫЙ S2.

S3 сеанса, ОБНОВЛЕННЫЙ, и дорожка 0, является В (extn 3000), и дорожка 1 является С (extn 2000).

Факторы:

- Передача дальнего конца инициирует ОБНОВЛЕНИЕ существующего сеанса записи.
- Участник дальнего конца изменяется на тот из S1.
- S3 новый дальний конец xRefCi совпадает с дальним концом S1 xRefCi.
- Агент Б (extn 3000) зависает.
- С (extn 2000) и В (extn 3000) разъединены.
- S3 сеанса ЗАКОНЧИЛСЯ

Примечание: Передача дальнего конца приводит к обновлению существующего сеанса. Разветвляющийся телефон остается единственным участником Дорожки 0, но участником Дорожки 1 изменение новой партии.

В случае Менеджера Унифицированной связи 10.0 и позже, вот результат:

1. Абонент К (extn 2000) вызывает Агента (extn 1000).

С (extn 2000) говорит с Агентом (extn 1000). S1 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 (extn 1000), Дорожка 1 является С (2000)

2. Агент (extn 1000) консультируется с Агентом Б (extn 3000).

3. С (extn 2000) слышит МоН. (extn 1000) говорит с В (extn 3000).

- S1 сеанса - ЗАКОНЧИЛСЯ
- Сеанс, который S2 - НАЧАЛ - Дорожка 0, (extn 1000), Дорожка 1 является В (extn 3000)

- S3 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 является В (extn 3000), Дорожка 1 (extn 1000)

Факторы:

- S1 сеанса заканчивается, потому что Агент телефон разместил Абонента К в ожидании.
- Открывает сеанс S2 и S3, оба существуют, потому что оба телефона установлены настройки для разветвления.
- Участники, а также xRefCi для этих двух участников S2 и S3 идентичны, но обратных позиций друг от друга.
- Значения S1 xRefCi не отражены или в S2 или в S3, так как консультирование считают независимым вызовом

4. Агент (extn 1000) завершает передачу.

5. С (extn 2000) говорит с В (extn 3000).

6. (extn 1000) разъединен.

- S3 сеанса - ЗАКОНЧИЛСЯ
- Сеанс S2 - ЗАКОНЧИЛСЯ
- S4 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 является В (extn 3000), Дорожка 1 является С (extn 2000)

Факторы:

- Передача дальнего конца инициирует конец одного сеанса записи и запуск другого сеанса записи.
- Хотя новый session start, его значения xRefCi будут совпадать с предыдущими сеансами.
- Дальний конец S4 xRefCi совпадает с дальним концом S1 xRefCi и ближним концом S4 xRefCi ближний конец S3 соответствий xRefCi.

7. Агент Б (extn 3000) зависает.

8. С (extn 2000) и В (extn 3000) разъединен.

- S4 сеанса - ЗАКОНЧИЛСЯ

Примечание: Передача дальнего конца приводит к концу одного сеанса записи и запуску другого сеанса записи.

1.5. Конференции агента с другим сценарием агента

1. Абонент К (extn 2000) вызывает Агента (extn 1000).
2. Агент (extn 1000) консультируется с Агентом Б (extn 3000).
3. Агент (extn 1000) завершает конференцию.
4. Агент (extn 1000) понижается от конференции.
5. Агент Б (extn 3000) зависает.

В случае Менеджера Унифицированной связи 9.x и ранее, вот результат:

1. Абонент К (extn 2000) вызывает Агента (extn 1000).

2. С (extn 2000) говорит (extn 1000).

S1 сеанса ЗАПУСТИЛСЯ - Дорожка 0 (extn 1000), Дорожка 1 является С (extn 2000).

3. Агент (extn 1000) консультируется с Агентом Б (extn 3000).

4. С (extn 2000) слышит, что МоН (extn 1000) говорит с В (extn 3000)

- S1 сеанса ЗАКОНЧИЛСЯ
- Откройте сеанс ЗАПУЩЕННЫЙ S2 - Дорожка 0 (extn 1000), Дорожка 1 является В (extn 3000)
- S3 сеанса ЗАПУСТИЛСЯ - Дорожка 0 является В (extn 3000), Дорожка 1 (extn 1052)

Факторы:

- S1 сеанса заканчивается, потому что Агент телефон разместил Абонента К в ожидании.
- Открывает сеанс S2, и S3 существуют, потому что оба телефона установлены настройки для разветвления.
- Участники, а также xRefCi для этих двух участников S2 и S3 идентичны, но обратных позиций друг от друга.
- Значения S1 xRefCi не отражены или в S2 или в S3, так как консультирование считают независимым вызовом.

5. Агент (extn 1000) завершает конференцию.

6. С (extn 2000) говорящий (extn 1000) и В (extn 3000).

- Откройте сеанс ЗАКОНЧЕННЫЙ S2
- S3 сеанса ОБНОВИЛ - Дорожка 0 является В (расширение 3000), Дорожкой 1 является Мост конференц-связи
- S4 сеанса ЗАПУСТИЛСЯ - Дорожка 0 (extn 1000), Дорожкой 1 является Мост конференц-связи

Факторы:

Передача дальнего конца инициирует ОБНОВЛЕНИЕ существующего сеанса записи.

Завершение конференции внедрено:

Во время консультирования:

- Консультационный телефон имеет основной удержанный вызов и активный консультационный вызов.
- Телефон, с которым консультируются, только имеет один активный вызов (консультационный вызов).

Когда конференция завершена (все связанные стороны):

- Консультационный вызов консультационного телефона завершается.
- Основной вызов консультационного телефона получает передачу дальнего конца в мост конференц-связи.
- Телефон, с которым консультируются, получает передачу дальнего конца в мост конференц-связи.

В результате:

- S2 завершается, потому что он представляет консультационный вызов консультационного телефона, который также завершается.
- S4 запускается; это представляет продолжение и передачу дальнего конца А основной вызов, но исходный S1 не может быть ОБНОВЛЕН, потому что был ранее завершен должный ранее держаться.

- S3 ОБНОВЛЕН, потому что дальний конец В просто передается от до моста конференц-связи.
- Ближний конец S4 xRefCi значение будет совпадать с ближним концом S1 xRefCi значение.

7. Агент (extn 1000) понижается от конференции.

8. (extn 1000) разъединен. С (extn 2000) говорящий В (extn 3000) ОБНОВЛЕННЫЙ S3 Сеанса - Дорожка 0 является В (extn 3000), Дорожка 1 является С (extn 2000).

Факторы:

- Уменьшение конференции в обычный двухпартийный вызов внедрено как передача дальнего конца обоим остающимся телефонам друг другу
- Передача дальнего конца инициирует ОБНОВЛЕНИЕ существующего сеанса записи.
- S3 и S1 будут иметь соответствующий ближний конец xRefCi значениями. Обратите внимание на то, что только один сеанс остается активным, потому что у Абонента К нет разветвления включенным.

9. Агент Б (extn 3000) зависает.

10. С (extn 2000) и В (extn 3000) разъединен.
S4 сеанса ЗАКОНЧИЛСЯ.

Факторы:

- Передача дальнего конца приводит к обновлению существующего сеанса. Разветвляющийся телефон остается единственным участником Дорожки 0, но участником Дорожки 1 изменение новой партии.
- Конференция создана с передачей всех телефонов к мосту конференц-связи. Поэтому конференция действует точно так же, как ряд передач. Существующие сеансы обновлены на тех сеансах, разветвляющийся телефон остается единственным участником Дорожки 0, но участником Дорожки 1 изменение к мосту конференц-связи.
- Как только третья сторона понижается от конференции, стороны переданы друг другу. Это обновляет существующие сеансы снова, разветвляющийся телефон остается единственным участником Дорожки 0, но участником Дорожки 1 изменение другой стороне.
- Если четвертая сторона добавлена к мосту конференц-связи, в метаданных нет никакой индикации, пока четвертой стороне также не включили ее собственное разветвление.

В случае Менеджера Унифицированной связи 10.x и позже, вот результат:

1. Абонент К (extn 2000) вызывает Агента (extn 1000).

2. С (extn 2000) говорящий (extn 1000) S1 Сеанса - ЗАПУСТИЛСЯ - Дорожка 0 (extn 1000), Дорожка 1 является С (extn 2000).

3. Агент (extn 1000) консультируется с Агентом Б (extn 3000).

4. С (extn 2000) слушание МоН (extn 1000), говорящий В (extn 3000).

- S1 сеанса - ЗАКОНЧИЛСЯ
- Сеанс, который S2 - НАЧАЛ - Дорожка 0, (extn 1000), Дорожка 1 является В (extn 3000)

- S3 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 является В (extn 3000), Дорожка 1 (extn 1000)

Факторы:

- S1 сеанса заканчивается, потому что телефон Агента А разместил Абонента К в ожидании.
- Открывает сеанс S2, и S3 существуют, потому что оба телефона установлены настройки для разветвления.
- Участники, а также xRefCi для этих двух участников S2 и S3 идентичны, но обратных позиций друг от друга.
- Значения S1 xRefCi не отражены или в S2 или в S3, так как консультирование считают независимым вызовом

5. Агент (extn 1000) завершает конференцию.

6. С (extn 2000) говорящий (extn 1000) и В (extn 3000)

- Сеанс S2 - ЗАКОНЧИЛСЯ
- S3 сеанса - ЗАКОНЧИЛСЯ
- S4 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 (extn 1000), Дорожкой 1 является Мост конференц-связи
- S5 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 является В (extn 3000), Дорожкой 1 является Мост конференц-связи

Факторы:

Передача дальнего конца инициирует конец одного сеанса записи и запуск другой записи. Завершение конференции внедрено перечисленное здесь:

- Во время консультирования: Консультационный телефон имеет основной удержанный вызов и активный консультационный вызов. Телефон, с которым консультируются, только имеет один активный вызов (консультационный вызов)
- Когда конференция завершена (все связанные стороны):
 - Консультационный вызов консультационного телефона завершается
 - Основной вызов консультационного телефона получает передачу дальнего конца в мост конференц-связи
 - Телефон, с которым консультируются, получает передачу дальнего конца в мост конференц-связи
- В результате: Два новых сеанса созданы, потому что и Агенту А и Агенту Б включили разветвление. Ближний конец S4 xRefCi значение и ближний конец S1 xRefCi оценивает соответствия. Ближний конец S5 xRefCi значение и ближний конец S3 xRefCi оценивает соответствия. Дальний конец xRefCi значения для S4 и S5 не совпадает, даже при том, что оба связаны с тем же мостом конференц-связи

7. Агент (extn 1000) понижается от конференции

8. (extn 1000) разъединен. С (extn 2000) говорящий В (extn 3000)

- S4 сеанса - ЗАКОНЧИЛСЯ
- S5 сеанса - ЗАКОНЧИЛСЯ
- S6 сеанса - ЗАПУСТИЛСЯ - Дорожка 0 является В (extn 3000), Дорожка 1 является С

(extn 2000)

Факторы:

- Уменьшение конференции в обычный двухпартийный вызов внедрено как передача дальнего конца обоим остающимся телефонам друг другу
- Передача дальнего конца инициирует конец одного сеанса записи и запуск другого сеанса записи
- S6 и S5 будут иметь соответствующий ближний конец xRefCi значениями. Обратите внимание на то, что только один сеанс остается активным, потому что у Абонента К нет разветвления включенным

9. Агент Б (extn 3000) зависает

10. С (extn 2000) и В (extn 3000) разъединен

- S6 сеанса - ЗАКОНЧИЛСЯ

Факторы:

- Передача дальнего конца приводит к концу одного сеанса и запуску другого
- Конференция создана с передачей всех телефонов к мосту конференц-связи. Поэтому конференция действует точно так же, как ряд передач. Существующие сеансы закончены, и новые сеансы созданы между разветвляющимися телефонами и мостом конференц-связи
- Как только третья сторона понижается от конференции, стороны переданы друг другу. Данные стороны сеансы, которые включали мост конференц-связи и начинают новые сеансы между двумя остающимися оконечными точками
- Если четвертая сторона добавлена к мосту конференц-связи, в метаданных нет никакой индикации, пока четвертой стороне также не включили ее собственное разветвление

2. Как мы коррелируем ссылочные идентификаторы вызова для других сеансов вызова при разветвлении Унифицированного элемента границы?

С разветвлением Унифицированного элемента границы очень немногие ситуации заставляют вызов быть разделенным на множественные сеансы записи.

Держитесь/Возобновляйте, передача и операции конференции не начинают новые сеансы записи в большинстве случаев. В немногих случаях, где новые сеансы созданы, существует стандартное значение, CCID (ID Соотношения вызова). Это значение характерно для всех сеансов в вызове. CCID является десятичной формой GUID Cisco, уникальная кнопка вызова, которая генерируется голосовыми маршрутизаторами Cisco. Первый маршрутизатор, который принимает вызов, генерирует этот ключ и передает его по линии ко всем последующим устройствам включая Cisco MediaSense.

Сам Унифицированный элемент границы не генерирует значения xRefCi, но создать сходство с вызовами разветвления Телефона диспетчера Унифицированной связи, Cisco MediaSense также синтезирует пару значений xRefCi для каждого вызова Унифицированного элемента границы. Они могут быть замечены в метаданных на уровне дорожки, наряду с CCID, который появляется на сеансовом уровне.

Эти ситуации заставляют записи Унифицированного элемента границы быть разделенными на несколько сеансов:

2.1. Середина изменения кодека вызова

Если передача, конференция, отбрасывание конференции или другая операция заставляют стороны пересматривать свой кодек, Cisco MediaSense заканчивает текущий сеанс записи и запускает новый. Два сеанса совместно используют тот же CCID и ту же пару значений xRefCi.

2.2. Передачи консультации

Передача консультации является передачей от одного агента другому, в котором эти два агента говорят друг с другом, в то время как инициатор вызова ждет в ожидании. Консультируемый участок вызова отнесен в некотором роде к полному вызову, и возможно настроить Менеджера Унифицированной связи, таким образом, что консультационные вызовы действительно проходят через CUBE. Однако Унифицированный элемент границы и Cisco MediaSense не знают, что эти вызовы отнесены, и они создают новый CCID и новую пару значений xRefCi для этого сеанса.

Эти вызовы могут быть привязаны друг с другом, выдерживают сравнение участвующего deviceRef и полей timestamp. Рассмотрим следующий сценарий:

1. Абонент К (extn 2000) вызывает Агента (extn 1000) (открытый сеанс = S1, CCID = C1)
2. Агент А консультируется с Агентом Б (extn 3000) (открытый сеанс = S2, CCID = C2)
3. Агент отбрасывания и Абонент К говорит с Агентом Б (открытый сеанс = S1, CCID = C1)

Сигнал тревоги в этом сценарии находится в Шаг 2. В течение того периода Агент (deviceRef 1000) является участником двух сеансов записи сразу:

- Сеанс = S1 / CCID = C1 и
- Сеанс = S2 / CCID = C2

Поэтому S1 отнесен к S2, и C1 отнесен к C2.

2.3. Обнаружение консультационных вызовов

Во-первых, нам нужно ясное определение консультационного вызова:

Любой вторичный вызов, который выполнен текущим участником на существующем сеансе к окончательной точке, которая является вне того сеанса и это исключает других участников того сеанса.

В теории этот сценарий мог включать места агента абонент в ожидании для согласования с его боссом для разрыва крена, или даже агент приостановил абонента для приема вызова от его жены, но мы игнорируем те возможности на данный момент.

Для клиентского приложения возможно обнаружить консультационный вызов в режиме реального времени дорожкой потока событий Cisco MediaSense. Если клиент замечает, что событие STARTED сеанса содержит данный deviceRef с другим STARTED сеанса, событие содержит тот же deviceRef без прошедшего события ENDED сеанса, это может прийти к

заклучению, что привязаны sessionIds и CCIDs, найденный в ЗАПУЩЕННЫХ событиях сеанса этих двух.

Исторически, клиент может проверить для любых консультационных вызовов, которые привязаны к данному основному вызову к API Cisco MediaSense. Предположите, что клиент знает что Агент используемый extn 1000 в CCID <C1>. Это инструкции для обнаружения любых связанных консультационных вызовов:

Шаг 1. Получите метаданные сеанса для основного вызова путем запуска getSessionByCCID (<C1>).

Шаг 2. Извлеките sessionStartDate (вызовите его <Ta>), и sessionDuration.

Шаг 3. Вычислите sessionEndDate (вызовите его <TB>) путем добавления sessionDuration к <Ta>.

Шаг 4. . Выполните этот запрос API:

https://Mediasense IP address:8443/ora/queryService/query/getSessionsByDeviceRef?value=1000&minSessionStartDate = <Ta> &maxSessionStartDate = <TB>

Этот запрос может вернуть несколько сеансов. Если это делает, то все они, как может предполагаться, привязаны к тому жю самому вызов.

2.4. Обнаружение консультационных вызовов от Множественных Участников

Процедура, упомянутая в разделе обнаружения консультационного вызова, находит все консультационные вызовы сделанными из устройства, которое получило начальный телефонный звонок. Однако, что, если там консультационные вызовы сделаны от устройства, до которого был впоследствии передан вызов?

Рассмотрите эту процедуру:

1. Абонент вызывает Агента 1
2. Агент 1 консультируется с Агентом 2, затем понижается
3. Абонент говорит с Агентом 2
4. Агент 2 консультируется с Агентом 3, затем понижается
5. Абонент говорит с Агентом 3

Эта процедура не ловит консультационный вызов между Агентом 2 и Агентом 3.

Так как это - вызов Унифицированного элемента границы, мы можем использовать факт, что все соединения между абонентом и каждым из агентов включены в тот же сеанс записи и факт, что все вовлеченные агенты перечислены как участники того же сеанса в какой-то момент. Таким образом, от основных метаданных сеанса, мы можем собрать список всех deviceRefs, которые были включены. Для обнаружения тех сеансов мы можем сделать серию из вызовов к getSessionByDeviceRef, задает временной диапазон основного сеанса, вместе с одним deviceRef на запрос.

Также процесс может быть упрощен с одиночным запросом **getSessions**, таким как это:

```
{  
  "requestParameters": [  

```

```

{
  "fieldName": "deviceRef",
  "fieldConditions": [
    {
      "fieldOperator": "equals",
      "fieldValues": [
        "1000"
      ],
      "fieldConnector": "OR"
    },
    {
      "fieldOperator": "equals",
      "fieldValues": [
        "2000"
      ],
      "fieldConnector": "OR"
    },
    {
      "fieldOperator": "equals",
      "fieldValues": [
        "3000"
      ],
      "fieldConnector": "OR"
    },
    {
      "fieldOperator": "equals",
      "fieldValues": [
        "4000"
      ]
    }
  ],
  "paramConnector": "AND"
},
{
  "fieldName": "sessionStartDate",
  "fieldConditions": [
    {
      "fieldOperator": "between",
      "fieldValues": [
        <Ta>, // session start time
        <Tb> // session end time
      ]
    }
  ]
}
]
}
]
}

```

Этот запрос возвращает все консультационные вызовы, привязанные к исходному основному вызову и всем его передачам.

Эта процедура фактически передает сеть слишком широко. Если, например, агент в deviceRef 4000 провел и завершённый абсолютно независимый вызов, который, оказалось, запускался *после* <Ta> и *прежде чем* он был добавлен к рассматриваемому вызову, эта процедура включает тот независимый вызов в набор. Эта проблема может быть решена хотя с доступной информацией в метаданных основного сеанса. Информация каждого участника включает сдвиг времени, в котором он присоединился к сеансу и продолжительности его срока. Код клиента мог использовать ту информацию для простого удаления не связанных друг с другом сеансов из списка, который это получило выше. Или, это могло сформулировать серию прямого getSession или запросов getSessionByDeviceRef, которые правильно структурируют периоды времени, во время которых каждый агент был в основном вызове. Мы оставляем это как осуществление читателю.

2.5. Сводка

В предыдущем разделе мы представили procedures для извлечения всех сеансов, привязанных к данному сеансу записи Cisco MediaSense. Однако мы также видели, что данный вызов может быть разделен на несколько сеансов, как в случае середины изменения кодека вызова.

Как мы получаем все записи (а также консультируется), привязанный ко *всем* сеансам, связанным со взаимодействием абонента?

Ответ должен расширить это инструкции для обнаружения множественных консультационных вызовов. Во-первых, мы собираем все сеансы, которые совместно используют SSID основного рассматриваемого сеанса. Затем мы создаем наш список участников от *всех* тех записей сеанса. Затем, мы вычисляем временные диапазоны как `sessionStartDate` самого раннего сеанса через конец последнего сеанса. наконец, мы можем выполнить показанный запрос `getSessions`.

Как прежде, мы можем закончить с перехватом слишком многих записей, таким образом, мы могли выполнить шаг постобработки для удаления тех не связанных друг с другом сеансов из его списка.

3. Как вы привязываете вызовы в Cisco MediaSense с их появлениями в других компонентах решения?

3.1. Таблица корреляции идентификатора

Эти две таблицы — один для Вызовов диспетчера Унифицированной связи и один для вызовов Унифицированного элемента границы. Каждый столбец представляет компонент решения, или протокол, с первым столбцом представляет Cisco MediaSense. Каждая строка представляет определенный тип идентификатора.

Для чтения таблиц начните с ячейки, которая представляет элемент данных, который вы знаете, и затем смотрите горизонтально к столбцу, представляет компонент решения, в котором вы хотите найти вызов. Запись в той ячейке указывает, каким названием тот же самый элемент данных известен в целевом компоненте. Если целевой компонент имеет пустую ячейку в той строке, то тот элемент данных не известен тому компоненту. Можно вместо этого искать прошедший столбец, где можно пересечься вертикально в другую строку, где та ячейка не является пробелом в столбце целевого компонента.

Например, с Вызовом диспетчера Унифицированной связи, предположите знание GED-188 `CallReferenceID` и что вы хотите найти вызов в Cisco MediaSense. Проверьте оставленный от столбца GED-188, вы видите, что нет никакого значения в столбце MediaSense, таким образом, вы не можете сопоставить непосредственно с ним.

Однако существует столбец, где можно делать зигзаги через строки: Менеджер Унифицированной связи CDR. Клиент может выбрать надлежащего Менеджера Унифицированной связи Запись CDR путем поиска той, в которой `IncomingProtocolCallRef` совпадает с GED-188 `CallReferenceID`. Та запись содержит значение, названное `destLegCallIdentifier`, который совпадает с MediaSense `NearEnd xRefCi` и может поэтому использоваться для обнаружения соответствующей записи в Cisco MediaSense.

Менеджер Унифицированной связи Записи CDR не записаны до некоторого времени после конца вызова, завершает, однако, таким образом, этот метод может только использоваться исторически.

Также существует другой путь. Выглядите нисходящими от GED-188 CallReferenceID. Оказывается, что можно также использовать AlertingDevice и AnsweringDevice для соответствия с deviceRef полем в MediaSense. Этот метод также работает в режиме реального времени.

Call Correlation for Calls Forked by a Unified CM IP Phone

MediaSense	Ingress Gateway or CUBE	AAA RADIUS CDR	UCM CDR	TAPI/JTAPI field	UCCE Database	UCCE Script	CTIOS	GED-188
(1)	Cisco-GUID		IncomingProtocolCallRef	CiscoConnection.UniqueID	TCD.CallGUID	Call.CallGUID		CallReferenceID
NearEnd xRefCi			destLegCallIdentifier	Terminal.ConnectionID				
FarEnd xRefCi			origLegCallIdentifier	Terminal.ConnectionID				
			global_CallID_call-ManagerId + global_CallID_callId (A.K.A. UCM GCID)	CiscoCall.CallID	TCD.PeripheralCallKey			
deviceId							Agent.AgentInstrument	
deviceRef					TCD.InstrumentPortNumber (2)		Agent.AgentExtension or Agent.Extension	AlertingDevice or AnsweringDevice

Call Correlation for Calls Forked by CUBE

MediaSense	Ingress Gateway or CUBE	AAA RADIUS CDR	UCM CDR	TAPI/JTAPI field	UCCE Database	UCCE Script	CTIOS	GED-188
CCID (3)	Cisco-GUID	Cisco-GUID	IncomingProtocolCallRef		TCD.CallGUID	Call.CallGUID		CallReferenceID
deviceRef	Called or calling party extn				TCD.InstrumentPortNumber (2)		Agent.AgentExtension or Agent.Extension	AlertingDevice or AnsweringDevice

Факторы:

1. В записях Вызовов диспетчера Унифицированной связи Cisco MediaSense действительно фактически получает GUID Cisco от UCM, но это не тот же самый, который перехвачен другими устройствами решения. MediaSense поэтому даже не хранит это значение.
2. Для вызовов от агента к агенту TCD.InstrumentPortNumber является расширением целевого агента. Расширение агента вызова может быть найдено в TCD.ANI.
3. CCID является GUID Cisco в десятичной форме, которая является 4 разделенными от дефиса наборами 10-разрядных десятичных номеров. Они могут быть преобразованы в шестнадцатеричную форму с, просто преобразовывают каждого 10-разрядного десятичного номера в 8-разрядное шестнадцатеричное число и удаляют дефисы. Где GUID Cisco используется в UCCE, это находится в своей шестнадцатеричной форме.

4. Как вы определяете, какая дорожка имеет вызывающую сторону и какая дорожка имеет вызываемую сторону?

4.1. Для вызовов, разветвленных CUBE

Для вызовов CUBE, Дорожка 0 всегда карты к потоку мультимедиа участка Привязки.

Участок Привязки является точкой вызова, которая среды, делающие запись профиля, настроена. Вторая дорожка сопоставляет с участком непривязки.

Если вам включили среды, делающие запись профиля на *входящем dialpeer*, то участок привязки становится в участке. Другими словами, вызывающая сторона появляется в Дорожке 0, и вызываемая сторона появляется в Дорожке 1.

Если вам включили среды, делающие запись профиля на *исходящем dialpeer*, то участок привязки становится-участком. В этом случае вызывающая сторона появляется в Дорожке 1, и вызываемая сторона появляется в Дорожке 0.

4.2. Для вызовов, разветвленных по унифицированным телефонам СМ

Для Унифицированного разветвления СМ в сценариях простого вызова можно использовать xRefCi поля в метаданных для определения, какая сторона находится, в котором среды отслеживают. Численно меньший xRefCi обычно обращается к дорожке вызывающей стороны. Дорожка вызываемой стороны численно больше (обычно одним, но это могло быть больше под обоснованно загружаемая система). Однако эти значения xRefCi в конечном счете повторяются для обнуления. Так, если вы находите, что одно значение является большим числом, и другой небольшое число, вы предполагаете, что инвертированы их позиции.

В более сложных сценариях не всегда работает этот алгоритм. Если дополнительные сервисы вызваны, такие как передачи и конференции, и кластер Менеджера UC состоит из нескольких узлов, то значения xRefCi не обязательно генерируются последовательно, и вы не можете предположить, что их заказ имеет любое значение вообще. Прямой способ, чтобы определить, можно ли последовательности заказа определенной пары значений xRefCi доверять, должен посмотреть на первый байт значений xRefCi. Этот байт представляет Идентификатор узла Менеджера UC, на котором был создан тот определенный идентификатор. Если первые байты двух значений xRefCi являются тем же, то их заказ корректен. Если они являются другими, то заказ не мог бы быть корректным.

Для этих случаев единственный способ определить направление вызова в режиме реального времени состоит в том, чтобы получить информацию из любого другого источника, такого как канал События JTAPI. Как только вызов закончился, и несколько минут истекли, можно всегда определять направление вызова и проверять данные CDR Менеджера UC для вызова. В частности origLegCallIdentifier поле в Записи CDR всегда представляет абонента.

5. Каковы возможные причины для состояния сеанса CLOSED_ERROR?

Возможные причины для состояния сеанса CLOSED_ERROR включают:

1. Сервер управления вызовами получил ошибочный ответ от Сред (делающих запись) сервера для открытого или близкого запроса.
2. Сервер управления вызовами обнаружил SIP сигнальная ошибка, например недостающий ACK.
3. Сеанс был успешно закрыт, но дорожки ALL имеют нулевой размер.

Когда сеанс находится в Активном состоянии, это обычно, что нет никакой

продолжительности в метаданных, потому что продолжительность не известна, пока сеанс не закрыт.

Для сеанса , который находится в состоянии CLOSED_ERROR, если или сеанс или поля продолжительности дорожки не присутствуют или в событии или в getSessionс данных, то среды для этой дорожки не доступны.

6. Каково различие между Сокращенными и Удаленными сеансами?

Рассмотрите эти два запроса:

6.1. С Запросом getAllPrunedSessions

Этот запрос возвращает ряд сеансов, все чей состояния сеанса УДАЛЕННЫ:

```
https://Mediaserver IP address:8443/ora/queryService/query/getAllPrunedSessions?minSessionStartDate=1301788800000&maxSessionStartDate=1312329599000
```

6.2. С Запросом getSessionс

Этот запрос не возвращает сеансов:

```
https://MediaServer IP address:8443/ora/queryService/query/getSessionс
{
  "requestParameters":
  [
    {
      "fieldName" : "sessionState",
      "fieldConditions":
      [
        {
          "fieldOperator" : "equals",
          "fieldValues" : [ "DELETED" ]
        }
      ],
      "paramConnector" : "AND"
    },
    {
      "fieldName" : "sessionStartDate",
      "fieldConditions":
      [
        {
          "fieldOperator" : "between",
          "fieldValues" : [ "1301788800000", "1312329599000" ]
        }
      ]
    }
  ]
}
```

6.3. Почему различие в поведении в сокращенных и удаленных сеансах?

Различие в поведении дизайном. См. это разделяет в документации MediaSense:

- Описание параметров API: `getAllPrunedSessions` описание API:

Используйте этот API для поиска всех сокращенных записей... Сокращенный термин относится к записям, которые удалены системой Cisco MediaSense. Если вы явно удалили какую-либо запись с помощью `deleteSessions` API, то эти удаленные записи не рассматривают как сокращенные записи.

- MediaSense SRND под разделом Упреждающее Управление хранением:

Когда сеансы сокращены, метаданные, которые привязаны к этим сеансам, остаются в базе данных, даже после того, как эти сеансы отмечены, как 'сокращено'. Эти метаданные не берут большое количество пространства для хранения по сравнению с самими записями, но это действительно занимает некоторое место и должно периодически удаляться. Для способствования этому действию клиенты могут периодически выполнять запрос API сокращенных сеансов, или клиенты могут выбрать получать сокращенные события сеанса и явно удалять те события это, клиентам больше не нужно.

Для разъяснения два запроса являются совершенно другими. На самом деле второй запрос (`lwhich` ищет все сеансы, состояние которых УДАЛЕНО) всегда возвращает пустое множество. Обычные ежедневные запросы отфильтровывают сеансы с УДАЛЕННЫМИ состояниями, даже если, именно это запрашивают. Единственным исключением является **`getAllPrunedSessions`**. Это исключение предназначено, чтобы помочь приложению находить сокращенные сеансы так, чтобы приложение могло запросить, чтобы были удалены эти сеансы.

Как только вы используете **`deleteSessions` API** в списке сокращенных сеансов, которые вы получаете от **`getAllPrunedSessions`**, эти сеансы больше не появляются в результате `getAllPrunedSessions`. Такие сеансы полностью сразу удалены из метаданных.

Другой способ посмотреть на это состоит в том, что сокращенные сеансы не являются той же вещью как удаленные сеансы:

1. Сокращенные сеансы были отмечены для удаления алгоритмом в системе MediaSense. Никакой человек не был вовлечен в решение сократить эти сеансы. Таким образом даже при том, что в эти сеансы переходят УДАЛЕННОЕ состояние, эти сеансы фактически не удалены из метаданных. Человек (или Приложение) вмешательство требуется. Поскольку эти сеансы находятся в УДАЛЕННОМ состоянии, эти сеансы не видимы к большинству запросов. Однако эти сеансы видимы к API запроса `getAllPrunedSessions`. Кроме того, если какие-либо mp4 файлы генерировались для этих сеансов, эти mp4 файлы продолжают присутствовать на диске и продолжать быть доступными для скачивания, пока фактически не УДАЛЕНЫ сокращенные сеансы.
2. Удаленные сеансы отмечены путем явного вызова `deleteSessions` API. Эта маркировка может быть сделана к сеансам, которые уже сокращены или к сеансам, которые еще не были удалены. Как только сеанс был удален `deleteSessions` API, этот сеанс больше не видим ни к какому запросу. Это включает `getAllPrunedSessions` API. Эти удаленные сеансы сразу удалены из метаданных так, чтобы могло быть исправлено дисковое пространство.

7. Как настроить TDM шлюз для разветвления сред?

Когда у вас есть шлюз PSTN, через которые потоки вызовов, и вы хотите сделать запись тех вызовов. Этими вызовами является TDM К ВЫЗОВАМ SIP. Однако разветвление сред только доступно на SIP К ВЫЗОВАМ SIP.

Эти вызовы могут быть зарегистрированы. Эти вызовы могут направленный через маршрутизатор во второй раз. Руководство по конфигурации и другие подробные данные могут быть найдены в [этом Описании технологических решений](#).

8. Как перехватить фактический телефон получателя при использовании группы последовательного поиска?

При использовании сред, разветвляющихся от CUBE метаданные MediaSense обычно содержат расширение вызываемой стороны. Однако, если вызванный номер является контрольным номером группы последовательного поиска Диспетчера связи, то по умолчанию метаданные только содержат тот контрольный номер. Это не содержит расширение телефона, который фактически ответил на звонок.

Существует значение Диспетчера связи, которое может изменить это. На Странице конфигурации Поиска/Пилота найдите, что раздел назвал **Связанные Партийные Преобразования. DN Элемента группы Строки дисплея установки как Связанная Сторона** должен быть включен.

Эта возможность доступна в Диспетчере связи 9.0 (1) и позже.

9. Почему Менеджеру Унифицированной связи Сетевая Запись рекомендуют как предпочтительный механизм разветвления?

С Менеджером Унифицированной связи Сетевая запись (NBR) можно использовать шлюз для записи вызовов. NBR позволяет Менеджеру Унифицированной связи направлять вызовы записи, независимо от устройства, местоположения или географии. С NBR среды записи вызовов могут быть получены или от IP-телефона или от шлюза, который связан с Менеджером Унифицированной связи по магистрали SIP. Менеджер Унифицированной связи динамично выбирает правильный источник сред на основе участников вызова и потока вызовов.

NBR предлагает автоматическую нейтрализацию Созданному в мосте (ViB), когда Маршрутизаторы ISR (ISR) недоступны, поскольку не требуется никакая отдельная конфигурация записи. Это полезно в случаях, где клиенты хотят включать консультационные вызовы агента-агента в политику записи, поскольку Унифицированный элемент границы не может сделать запись консультационных вызовов, таким образом, ViB должен быть включен отдельно.

И NBR и вызовы ViB могут быть коррелированы с помощью xRefsi, который доступен от Менеджера Унифицированной связи JTAPI. GUID CISCO не необходим, что не означает ни CTI Server, ни соединений CTIOS, требуются. Как существует одиночный идентификатор корреляции, корреляция через компоненты более сильна и может быть сделана

универсальным способом, независимым от потока вызовов.

С NBR непосредственно набранные, а также инициируемые в номеронабирателя исходящие вызовы могут быть коррелированы с их появлением в других компонентах решения.

С NBR запись шлюза TDM автоматически используется без разделения емкости маршрутизатора. В настоящее время запись шлюза TDM не поддерживается с MediaSense 10.5.

10. Почему узел занимает больше времени для обновления к MediaSense 10.5?

Узел может занять несколько часов для обновления, зависит от номера и размера записей, которые это держит. Для MediaSense 10.5 при обновлении узла с очень большими наборами данных требуется приблизительно 90 дополнительных минут на 1 миллион записей.

11. Каково влияние российских изменений часового пояса на приложении Поиска и Воспроизведения MediaSense?

На пользователей приложения Поиска и Воспроизведения MediaSense влияют, если они расположены в каком-либо из затронутых часовых поясов или если они выбирают затронутый часовой пояс в условиях поиска. На партнерские продукты третьей стороны, которые взаимодействуют с MediaSense, влияют так же, пока они не обновляют свои соответствующие таблицы часового пояса.

Обходной путь должен выбрать часовой пояс, который совпадает с корректным смещением от GMT, даже если город больше не корректен.

12. Что языки поддерживаются MediaSense?

Вот языки, поддерживаемые MediaSense:

- Арабский язык
- Датский язык
- Нидерландский язык
- Английский язык (Соединенные Штаты)
- Финский язык
- Французский
- Немецкий язык
- Итальянский язык
- Японский язык
- Корейский язык
- Норвежский язык
- Польский язык
- Португальский язык (бразилец)
- Русский язык

- Упрощенный китайский
- Испанский язык
- Шведский язык
- Традиционные китайцы
- Турецкий язык

13. Как контролировать производительность системы MediaSense?

Для мониторинга производительности системы MediaSense проанализируйте значения этих ключевых показателей эффективности (KPIs) с программным средством RTMT или Cisco Главное программное средство Обеспечения Совместной работы.

Для получения дополнительной информации о программном средстве RTMT или Cisco Главное программное средство Обеспечения Совместной работы, обратитесь Унифицированное администрирование RTMT и Cisco Главные разделы администрирования Обеспечения Совместной работы [Руководства пользователя Cisco MediaSense](#).

KPIs и их Пороговые значения

Ключевой показатель эффективности	Счетчики RTMT	Предложенные пороговые значения
Процент успешных вызовов	Сервис Управления вызовами MediaSense> Количество записи сеансов без ошибок	99.99%
	Сервис Управления вызовами MediaSense> Количество записи сеансов с ошибками	Процент успешных вызов = количество записи сеансов без ошибок / (количество записи сеансов без ошибок + количество записи сеансов с ошибками) * 100
Среднее время ответа API для API	Сервис API Cisco MediaSense> Среднее значение сделал запрос времени отклика	60 secs
Запись запускает среднюю задержку установки	Сервис Управления вызовами Cisco MediaSense> Средняя задержка установки	3 secs
Среднее использование ЦП	Процессор> %CPU Время	90%
Среднее использование памяти	Память> %Mem Используемый	70%
Отбрасывание пакета RTP/UDP	Сетевой интерфейс> Rx, Отброшенный> eth0	0
	Сетевой интерфейс> Ошибки Rx> eth0	0

14. Как настроить браузер для выполнения проигрывателя в

браузере в MediaSense?

На основе браузера выполните эти шаги для выполнения проигрывателя в браузере:

Internet Explorer 9

1. В Поиске MediaSense и Воспроизведении, нажмите значок **Воспроизведения** сеанса записи.

2. Нажмите **Yes** для доверия сертификату.

Примечание: Проверьте, что предлагаемый подписанный сертификат имеет предназначенный узел MediaSense путем проверки FQDN в технических подробностях сертификата.

Internet Explorer 11

Предварительные условия: В случае новой установки MediaSense 11.0 гарантируйте, что узлы MediaSense добавлены к кластеру с помощью соответствующего Полного доменного имени (FQDN). В случае обновления к MediaSense 11.0 гарантируйте, что узлы MediaSense, которые были ранее добавлены с помощью имени хоста, должны теперь быть отображены соответствующим FQDN. Проверьте окно **MediaSense Server List in MediaSense Server Configuration** (администрирование Cisco MediaSense > Система > Конфигурация сервера MediaSense).

Выполните следующие шаги:

1. Заставьте *набор hostnameformediaurl* CLI как "истинный" заставляя MediaSense подготовить mp4url и отдых mediaurls, использующего FQDN только.

```
admin:set useHostNameForMediaURL  
admin:set useHostNameForMediaURL true
```

2. Перезапустите Сервис настройки для активации свойства.

```
admin:utils service restart Cisco  
MediaSense Configuration Service
```

Примечание: Если сервис не перезапустился должным образом, выполните ту же команду снова.

3. После перезапусков Сервиса настройки выпишитесь и регистрируйтесь к **Поиску MediaSense и Воспроизведению**.

Ограничение: В случае, если узлы MediaSense были ранее добавлены с помощью IPs, тогда узлы продолжают быть отображенными IPs только даже после обновления к MediaSense 11.0. Проигрыватель в браузере не работает на Internet Explorer 11, независимо от

Mozilla Firefox

1. Добавьте подписанный сертификат узла MediaSense порта 8446 mp4url в доверять полномочиях Mozilla Firefox.

2. Для добавления подписанного сертификата нажмите, значок **загрузки** записи открываются и выбирают mp4. Это **Соединение** является всплывающим окном **Untrusted** появляется.

значения *hostnameformediaurl* команды CLI. В этом случае рекомендуется, чтобы *hostnameformediaurl* команда CLI не была установлена как "истинная". Выполните следующие шаги, чтобы добавить, что подписанный сертификат MediaSense к Windows доверял полномочиям.

1. Откройте **поиск MediaSense и воспроизведение**.

Всплывающее окно сертификата безопасности появляется.

2. **Нажмите кнопку Continue.**

Окно **MediaSense Search и Play** появляется.

3. В Строке адреса нажмите **Certificate Error icon**.

4. Нажмите **сертификаты View**.

Всплывающее окно сертификата появляется.

5. **Нажмите кнопку Install Certificate (Установить сертификат).**

Мастер Импорта Сертификата появляется.

6. **Нажмите кнопку Next.**

7. В витрине Сертификата выберите **Place все сертификаты в следующей** кнопке с зависимой фиксацией **хранилища** и нажмите **Browse**.

Диалоговое окно **Select Certificate Store** появляется.

8. Проверьте, что **Show physical хранит** флажок, и выберите папку **Trusted Root Certificationhorities**.

9. Нажмите **OK** и **Next**.

10. Нажмите **Finish** для завершения импорта сертификата.

Всплывающее окно предупреждения системы безопасности, кажется, подтверждает установку сертификата.

11. **Нажмите кнопку YES.**

Следующее сообщение появляется.

`The import was successful.`

12. **Нажмите кнопку OK.**

13. Нажмите **OK** на всплывающем окне **Certificate**.

14. Закройте и откройте браузер.

15. Откройте **поиск MediaSense и воспроизведение**.

3. Нажмите значок **Воспроизведения**, соответствующий выбранной записи.

Игрок в браузере играет выбранный сеанс записи.

2. Для добавления подписанного сертификата нажмите, значок **загрузки** записи открываются и выбирают **mp4**.

Это Соединение является всплывающим окном **Untrusted** появляется. **Примечание:** Проверьте, что предлагаемый подписанный сертификат импортирован в предназначенный узел MediaSense путем проверки FQDN в технических подробностях сертификата.

Сертификат безопасности issxvi все еще сохраняется.

16. Перейдите к **Программным средствам > интернет-Опции > Усовершенствованный**, снимите флажок с **Тем, чтобы предупреждать о boxviiunder** Безопасности проверки несоответствия адреса сертификата.

17. Нажмите **Apply** и **ОК**.

18. Перезапустите браузер и откройте **Поиск MediaSense** и **Воспроизведение**.

Гарантируйте, что сервер MediaSense доступен через FQDN на браузере. В противном случае перейдите к C:\Windows\System32\drivers\etc, откройте **файл hosts** в Блокноте и добавьте IP-адрес сервера MediaSense и его FQDN у основания файла. Проигрыватель в браузере начинает работать на Internet Explorer 11.

Примечание: Если запись присутствует на другом узле MediaSense кластера, вам предлагают добавить сертификат что узел MediaSense в доверяемых полномочиях.

3. Щелкните по ссылке **I Understand the Risks**.

4. Нажмите **Add** исключение. Всплывающее окно **Add Security Exception** появляется.

5. Нажмите **Confirm Security Exception**.

Подписанный сертификат определенного узла MS 844 портов добавлен к доверяемым полномочиям браузера.