

Создание отчетов CVP - устраняет неполадки ДВОЙНЫХ СПИСКОВ

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Списки копии устранения неполадок в создании отчетов CVP](#)

[Диагностировать](#)

[Обходной путь](#)

[Удалите двойной список](#)

[Удалите дублированные записи из таблиц](#)

[Перезагрузите последнее выполнение](#)

[Проверка](#)

[Дополнительные сведения](#)

Введение

Этот документ описывает процесс устранения неполадок двойных списков для Речевого портала Клиента Cisco (CVP) функция создания отчетов.

Внесенный Мохамедом Мохэссебом, Алексеем Янковским и Александром Левичевым, специалистами службы технической поддержки Cisco.

Предварительные условия

Требования

Компания Cisco рекомендует предварительно ознакомиться со следующими предметами:

- Microsoft Windows server
- CVP Cisco
- Программное средство Доступа DB informix

Используемые компоненты

Сведения в этом документе основываются на версии сервера 11.0 CVP, но это также применимо для предыдущих версий.

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

Списки копии устранения неполадок в создании отчетов CVP

В ciscoadmin базе данных **agg_schedule** табличные элементы управления, как часто выполнена каждая агрегация. В некоторых случаях после обновления таблица повторно загружена новым списком, однако, старый все еще существует. Это приводит к каждой агрегации, выполняемой дважды. Это неблагоприятно повлияет на точность сводных таблиц, поскольку каждый будет иметь дважды количество строк, и следовательно рассчитывает, вставленный в него.

Диагностировать

Проверьте, существуют ли копии в **agg_schedule** таблице.

Вход в систему к серверу Создания отчетов CVP с **cvp_dbadmin** пользователем.

Запустите программное средство **Windows CMD**. Введите **dbaccess** команду. Выберите вкладку **Connection> Подключение**. Выберите **cvp** сервер базы данных и нажмите Enter. Вам предложат ввести в учетных данных. Используйте учетную запись **cvp_dbadmin**.

Выберите **ciscoadmin@cvp** базу данных. Выберите **Query-language > New**. Выполните эту команду.

```
UNLOAD TO schedule.txt SELECT * FROM agg_schedule
```

Откройте **schedule.txt** файл. В зависимости от версии CVP это могло бы быть расположено в любом **C:\Users\Administrator** или **C:\db\Informix\etc\sysadmin** папки.

Это должно иметь уникальные записи списка данных CVP. Например, **call_15** или **applicationsummary_daily**, как изображено в этом изображении.

```
1 cvp_data|call_15|15|2016-09-07 07:33:57|dbdatetime|N|1|60 units day|
2 cvp_data|call_30|call_15|30|2000-01-01 00:00:00|dbdatetime|Y|2|60 units day|
3 cvp_data|call_hourly|call_15|60|2000-01-01 00:00:00|dbdatetime|Y|3|60 units day|
4 cvp_data|call_daily|call_15|DD|2016-09-07 07:33:57|dbdatetime|N|4|550 units day|
5 cvp_data|call_weekly|call_daily|WW|2016-09-07 07:33:57|dbdatetime|N|5|10 units year|
6 cvp_data|applicationsummary_15|15|2016-09-07 07:33:57|a.dbdatetime|N|6|60 units day|
7 cvp_data|applicationsummary_daily|applicationsummary_15|DD|2016-09-07 07:33:57|dbdatetime|N|7|550 units day|
8 cvp_data|applicationsummary_weekly|applicationsummary_daily|WW|2016-09-07 07:33:57|dbdatetime|N|8|10 units year|
9 cvp_data|call_monthly|call_daily|MM|2016-09-07 07:33:57|dbdatetime|N|9|40 units year|
10 cvp_data|applicationsummary_monthly|applicationsummary_daily|MM|2016-09-07 07:33:57|dbdatetime|N|10|40 units year|
11
```

Гарантируйте, что нет никаких двойных типов. Если копии найдены, тогда внедряют обходной путь.

Обходной путь

Удалите двойной список

Отключите итоговое задание (**CVPSummary**) в Windows Task Scheduler.

Откройте **schedule.txt** файл и удалите все двойные линии. Там должен остаться только первыми 10 записями.

Соединитесь с **ciscoadmin** базой данных, как описано прежде и выполните этот запрос. Эта команда удаляет все записи из **agg_schedule** таблицы.

- `DELETE FROM agg_schedule WHERE 1=1;`

Загрузите новые значения из **schedule.txt** файла, который не содержит копий к **agg_schedule** таблице.

- `LOAD FROM schedule.txt INSERT INTO agg_schedule;`

Гарантируйте, что в **agg_schedule** таблице нет никакого подарка копий. Выходные данные **schedule1.txt** файла должны содержать только 10 записей.

- `UNLOAD TO schedule1.txt SELECT * FROM agg_schedule;`

Удалите дублированные записи из таблиц

Так как 15-минутная таблица является ядром, от которого заполнены все другие таблицы, это должно быть исправлено сначала.

Соединитесь с **cvp_data** базой данных.

Выполните эти команды для **call_15** таблицы.

```
SELECT distinct * FROM call_15 into temp t1 with no log;
TRUNCATE table call_15;
INSERT into call_15 select * from t1;
DROP table t1;
```

Повторите те же шаги для **applicationsummary_15** таблицы.

```
SELECT distinct * from applicationsummary_15 into temp t1 with no log;
TRUNCATE table applicationsummary_15;
INSERT into applicationsummary_15 select * from t1;
DROP table t1;
```

Примечание: Если проблема запущена, больше чем 60 дней назад повторяют вышеупомянутые шаги для ежедневных, еженедельных и ежемесячных таблиц.

Перезагрузите недавнее выполнение

Сброс **lastrun** поле для 15-минутных таблиц.

Узнайте время, когда были обновлены 15-минутные таблицы. Выполните эти команды против **cvp_data** базы данных.

```
SELECT max(dbdatetime) FROM applicationsummary_15;
SELECT max(dbdatetime) FROM call_15;
```

Запишите число дней начиная с последнего обновления.

Выполните этот запрос для сброса последнего выполнения для 15-минутных таблиц на **ciscoadmin** DB. В данном примере 15-минутные таблицы были обновлены 17 дней назад прошлый раз.

```
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 17 units day) WHERE
dst_tabname LIKE 'call_15';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 17 units day) WHERE
dst_tabname LIKE 'applicationsummary_15';
```

Примечание: Замените "17" от команды с числом дней, которое вы получили от шага для каждой из этих двух таблиц.

После того шага будут исправлены 15-минутные таблицы.

Так как 15-минутные таблицы сохраняют данные в течение прошлых 60 дней, перезагружают значение `lastrun` для ежедневных, еженедельных и ежемесячных таблиц и удаляют все данные 60 дней назад для этих таблиц. Этот путь гарантирует, что в следующий раз, когда `aggregation.bat process start`, он заполнит данные в ежедневных, еженедельных, ежемесячных таблицах с правильными значениями.

Перезагрузите `lastrun` для ежедневных, еженедельных и ежемесячных таблиц с этими командами, выполняемыми против `ciscoadmin` базы данных.

```
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'call_daily';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'call_weekly';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'call_monthly';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'applicationssummary_daily';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'applicationssummary_weekly';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'applicationssummary_monthly';
```

Удалите все данные в ежедневных, еженедельных и ежемесячных таблицах в течение 60 дней назад.

```
DELETE FROM cvp_data:call_daily WHERE dbdatetime > (current - 60 units day);
DELETE FROM cvp_data:call_weekly WHERE dbdatetime > (current - 60 units day);
DELETE FROM cvp_data:call_monthly WHERE dbdatetime > (current - 60 units day);

DELETE FROM cvp_data:applicationssummary_daily WHERE dbdatetime > (current - 60 units day);
DELETE FROM cvp_data:applicationssummary_weekly WHERE dbdatetime > (current - 60 units
day);
DELETE FROM cvp_data:applicationssummary_monthly WHERE dbdatetime > (current - 60 units
day);
```

Включите итоговому заданию `CVPSummary` в планировщике задач для процесса агрегации для начала снова.

Проверка

Воспользуйтесь данным разделом для проверки правильности функционирования вашей конфигурации.

Гарантируйте, что сводные таблицы обновляются с этими командами, выполняемыми против `cvp_db` базы данных.

```
SELECT MAX(dbdatetime) FROM applicationssummary_15;
SELECT max(dbdatetime) FROM applicationssummary_daily;
SELECT max(dbdatetime) FROM applicationssummary_weekly;
SELECT max(dbdatetime) FROM applicationssummary_monthly;

SELECT MAX(dbdatetime) FROM call_15;
SELECT MAX(dbdatetime) FROM call_daily;
SELECT MAX(dbdatetime) FROM call_weekly;
SELECT MAX(dbdatetime) FROM call_monthly;
```

Дополнительные сведения

- [Интегрируйте клиента SQL белки с примером конфигурации создания отчетов CVP](#)
- Техническая поддержка и документация