

CVP レポート-重複したスケジュールを解決して下さい

目次

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[CVP レポートの重複したスケジュールを解決して下さい](#)

[診断して下さい](#)

[回避策](#)

[重複したスケジュールを取除いて下さい](#)

[表から重複した2つのエントリを削除して下さい](#)

[最近の実行をリセットして下さい](#)

[確認](#)

[関連情報](#)

概要

この資料は Cisco カスタマー 音声ポータル (CVP) レポート機能のための重複したスケジュールのトラブルシューティング プロセスを説明したものです。

モハメド Mohasseb、Aleksey Yankovskyy およびアレキサンダー Levichev によって貢献される、Cisco TAC エンジニア。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Microsoft Windows Server
- Cisco CVP
- Informix DB アクセス ツール

使用するコンポーネント

この文書に記載されている情報は CVP サーババージョン 11.0 に基づいていますが、前のバージョンにまた適当です。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、クリアな (デフォルト) 設定で作業を開始しています。ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

CVP レポートの重複したスケジュールを解決して下さい

ciscoadmin データベース `agg_schedule` テーブル コントロールではどの位の割り分で各集約が動作するか。場合によってはアップグレードの後で表は新しいスケジュールとしかしまだ存在する古いものリロードされます。これは二度実行された各集約という結果に終わります。これは各自に列番号が二度ある、およびそれ故に数逆効果をもたらしますのでそれに挿入された要約テーブルの正確さに。

診断して下さい

`agg_schedule` 表に重複があるかどうか確認して下さい。

`cvp_dbadmin` ユーザが付いている CVP レポート サーバーにログインして下さい。

Start ウィンドウ CMD ツール。型 `dbaccess` コマンド。タブ > 接続応答を『Connection』を選択して下さい。 `cvp` データベースサーバを選択し、『Enter』を押して下さい。資格情報を打ち込むためにプロンプト表示されます。 `cvp_dbadmin` アカウントを使用して下さい。

`ciscoadmin@cvp` データベースを選択して下さい。 `Query-language > New` を選択して下さい。このコマンドを実行して下さい。

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

`schedule.txt` ファイルを開いて下さい。CVP バージョンによってはそれは `C:\Users\Administrator` または `C:\db\Informix\etc\sysadmin` フォルダにあるかもしれません。

それはユニークな CVP データ Schedule エントリがあるはずですが。このピクチャに示すようにたとえば `call_15` または `applicationsummary_daily`。

```
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
```

重複型がないことを確認して下さい。重複があったら回避策を設定して下さい。

回避策

重複したスケジュールを取除いて下さい

Windows タスク スケジューラのサマリ ジョブ (`CVPSummary`) をディセーブルにして下さい。

`schedule.txt` ファイルを開き、すべての重複した行を削除して下さい。そこに最初の 10 のエントリだけ残るはずですが。

`ciscoadmin` データベースに前述のように接続し、このクエリを実行して下さい。このコマンドは `agg_schedule` 表からすべてのエントリを削除します。

- `DELETE FROM agg_schedule WHERE 1=1;`

schedule.txt ファイルから新しい値をロードして下さい agg_schedule 表に重複が含まれていない

。

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

そこにです agg_schedule 表で現在の重複確認しないで下さい。 schedule1.txt ファイルの出力は 10 のエントリだけ含まれているはずです。

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

表から重複した2つのエントリを削除して下さい

15分 表があるので他の表がすべてそれ読み込まれるベースは最初に修復される必要があります。

cvp_data データベースへの接続応答。

call_15 表のためのこれらのコマンドを実行して下さい。

```
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;
```

applicationsummary_15 表のための同じステップを繰り返して下さい。

```
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;
```

注: 問題が前に開始された 60 日以上リピート毎日の、週間および月例表のための上記のステップなら。

リセット最近の実行

15分 表のためのリセット lastrun フィールド。

15分 表が更新済だった時間を調べて下さい。 cvp_data データベースに対してこれらのコマンドを実行して下さい。

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

最新の更新プログラム以来の日数を書いて下さい。

ciscoadmin DB の 15分 表のための最後の実行をリセットするためにこのクエリを実行して下さい。 この例で 15分 表は前に更新済 17 日最後でした。

```
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';
```

注: それぞれの表のためのステップから得た日数によってコマンドからの "17" を取り替えて下さい。

後そのステップ 15分 表は訂正されます。

15分表が最後の60日のデータを保つので、毎日の、週間および月例表の lastrun 値を設定し直し、60日これらの表のためにすべてのデータを削除して下さい。この方法は aggregation.bat プロセス開始するが、それ正しい値との毎日、週間の、月例表のデータを読み込むその次の時間を確認します。

ciscoadmin データベースに対して実行されるこれらのコマンドで毎日の、週間および月例表のための lastrun をリセットして下さい。

```
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 'applicationsummary_daily';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'applicationsummary_weekly';
UPDATE ciscoadmin:agg_schedule SET lastrun = (current year to day - 60 units day) WHERE
dst_tabname LIKE 'applicationsummary_monthly';
```

毎日のすべてのデータを、週間削除すれば60日間月例表は支持します。

```
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:applicationsummary_daily WHERE dbdatetime > (current - 60 units day);
DELETE FROM cvp_data:applicationsummary_weekly WHERE dbdatetime > (current - 60 units
day);
DELETE FROM cvp_data:applicationsummary_monthly WHERE dbdatetime > (current - 60 units
day);
```

再度開始する集約プロセスのためのタスクスケジューラのサマリジョブ CVPSummary を有効にして下さい。

確認

このセクションでは、設定が正常に機能していることを確認します。

要約テーブルをアップデートされています cvp_db データベースに対して実行されるこれらのコマンドで確認して下さい。

```
SELECT MAX(dbdatetime) FROM applicationsummary_15;
SELECT max(dbdatetime) FROM applicationsummary_daily;
SELECT max(dbdatetime) FROM applicationsummary_weekly;
SELECT max(dbdatetime) FROM applicationsummary_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;
```

関連情報

- [CVP レポート設定例とリス SQL クライアントを統合](#)
- テクニカル サポートとドキュメント