

新しい mon sub next-call が開始されるたびに mon sub next-call オプションが失われる問題の回避策

内容

[問題](#)

[解決方法](#)

[関連するシスコ サポート コミュニティ ディスカッション](#)

問題

この記事では、コールキャプチャプロセスの開始前に、モニタサブスクライバの次のコール設定を設定する方法について説明します。monitor subscriber next-callが実行されるたびに、そのコマンドの前の実行の設定が失われ、再度選択する必要があります。特に、トラブルシューティング中の特定のシナリオに対して複数のオプションを有効にする必要がある場合、特に、Speedy Gonzalesのようなオプションをすべて選択する前にコールの開始時にトラブルシューティングが行われている場合は、これは特に困難です。

解決方法

monitor subscriber menu option approachを使用し、オプションF) Next-Callを選択し、キャプチャを開始して目的のオプションを選択し、Escキーを1回押します。メニューに戻ってFを選択すると、以前に選択したオプションがすべて有効になり、必要なデータが何も失われずに収集され始めます。Escapeキーが2回連続して誤ってシステムプロンプトに戻る場合は、このプロセスを繰り返し、さらにキャプチャを設定します。

このアプローチは、キャプチャの確率が低い特定のタイプのコールをキャプチャする場合に特に便利です。たとえば、少量のEvolved High Rate Packet Data(eHRPD)トラフィックを伝送し、大量の3GモバイルIP(MIP)トラフィックを伝送するノードがあるとした場合、eHRPDコールの割合はMIPに比べて低く、eHRPDコールのキャプチャに時間がかかることがあります。また、これらのコールはMIPコールと同じように開始されるため、「o) Next-EVDO-RevA Call」オプションを使用すると、両方のコールタイプがキャプチャされます。最後のオプションを記憶しているシステムに関して、前述と同じアプローチを次のように使用できます。eHRPDコールが着信するコンテキストから開始し、monitor subscriberを実行し、next-callを選択し、次にLimit contextのオプション「L」を選択します。このコンテキストは、monサブが開始されたコンテキストです。次にEscキーを押し、[次のコール]を再度選択すると、eHRPDコールだけがキャプチャされます。必要な回数だけ繰り返します。

limit contextオプションの欠点は、他のコンテキストを通過するコールのパケット、つまりプロキシMIPv6がPGWと交換される出力コンテキストがキャプチャされないことです。これは、トラブルシューティングシナリオで重要な場合とそうでない場合があります。

このアプローチの要点は、plain monitor subscriber next-callとnext-callのバリエーションの制限を回避できることです。