

# Cable voor Cisco IP-telefoonconsole op maat

## Inhoud

[Inleiding](#)

[Achtergrondinformatie](#)

[Typen troostpoorten](#)

[Hoe u de console-kabel maakt](#)

[USB-console-poortkabel](#)

[Zes pin aux console poortkabel](#)

[Stappen om de console poortkabel te gebruiken](#)

[Gerelateerde Cisco Support Community-discussies](#)

## Inleiding

Deze documenten beschrijft hoe u aan op maat gemaakte Cisco IP-telefoonconsole kabel kunt maken die wordt gebruikt om telefoonlogs rechtstreeks vanaf de console-poort te verzamelen.

## Achtergrondinformatie

De telefoon debug-logbestanden worden verzameld via drie verschillende methoden:

- Logbestanden van console vanaf web toegang
- Debug Mode door Secure Shell (SSH)-toegang
- Poortpoort

Gebruik methodes één en twee, wanneer de telefoon over een IP bereikbaar is. Soms is toegang tot het web of toegang tot SSH niet beschikbaar vanwege netwerkbeveiliging, bijvoorbeeld firewallbeleid.

Methode drie is nuttig in het oplossen van problemen scenario's, waarin de logboeken tijdens het telefoonbootup of voordat de telefoon IP bereikbaar is, moeten worden verzameld, zoals:

- DHCP-problemen oplossen
- VPN-probleem
- Telefonische bereikbaarheid of beveiligingsproblemen.

In deze situaties wordt een console-kabel gebruikt.

## Typen troostpoorten

Er zijn twee soorten console-poorten beschikbaar voor Cisco-telefoons, afhankelijk van het model.

- USB-console-poort op telefoonmodellen 99XX en 8961
- Een zes-pins AUX-poort op telefoonmodellen 79XX en 78XX

Opmerking: Telefoonmodel 88XX heeft zowel USB- als AUX-poorten.

## Hoe u de console-kabel maakt

De twee soorten console kabels zijn:

1. USB-console-poortkabel
2. Zes-pins aux-poortkabel

### USB-console-poortkabel

Voor het maken van een USB-console poortkabel hebt u:

- Twee USB seriële (DB9) kabels zoals in deze afbeelding getoond:



- Eén seriekabel met vrouwelijke poort op DB9 aan elke kant zoals in deze afbeelding wordt getoond:



De seriële poorten van de drie kabels moeten onderling worden aangesloten om de kabel te voltooien zoals in deze afbeelding wordt weergegeven:





Wanneer de USB-console poortkabel is aangesloten op de telefoon, verschijnt er een flash-bericht aan de telefoon: USB-seriële console-kabel gedetecteerd.

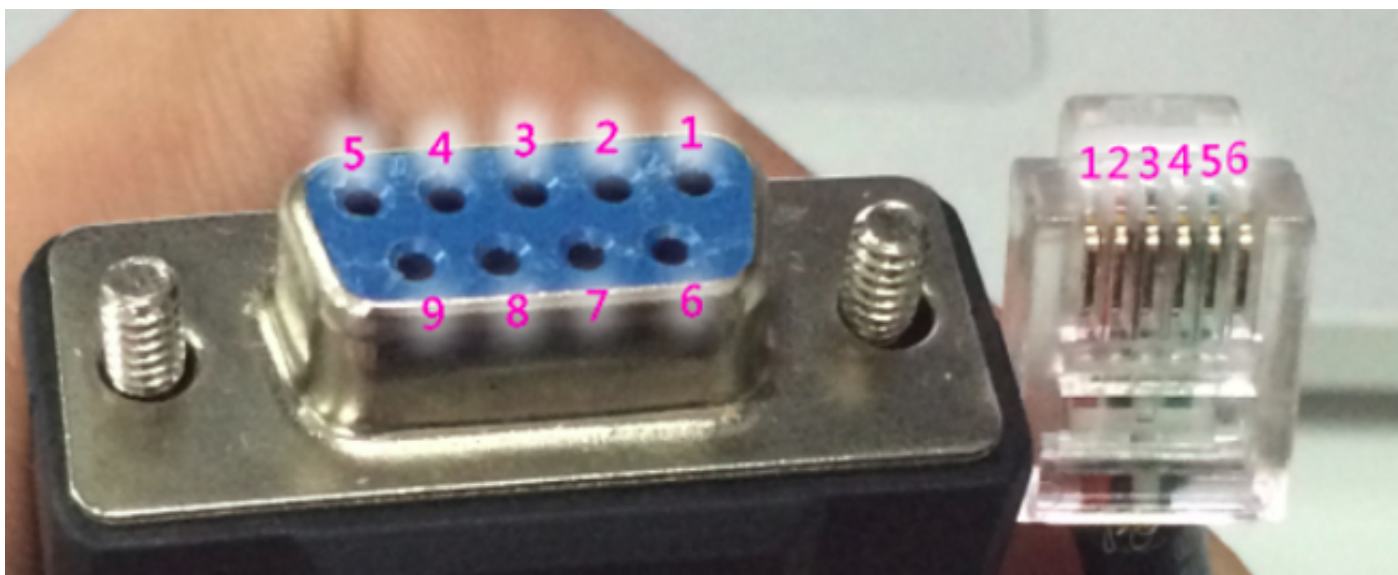
### **Zes pin aux console poortkabel**

U heeft als volgt zes pinnen per poort nodig:

- Eén USB seriële (DB9) kabel zoals in deze afbeelding getoond wordt:



- Eén zes pins RJ11-kabel en één DB9 (vrouwelijke) poort zoals in deze afbeelding wordt getoond:



In de DB9-haven worden de getallen acuut per haven genoemd. Voor RJ11 poort wordt het kabelnummer van links geteld terwijl de clip naar beneden gericht is.

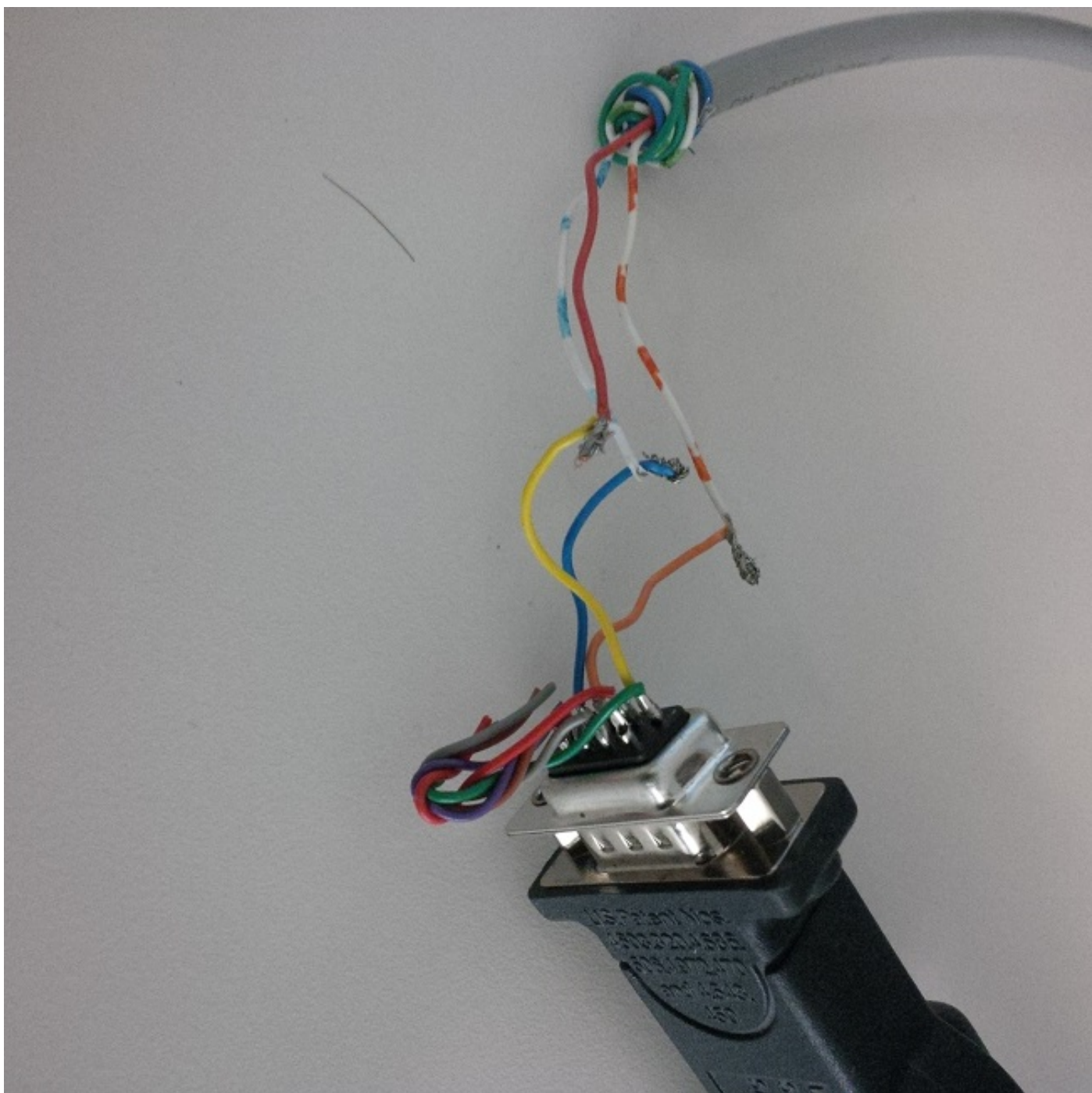
Draai de kabels van RJ11 en DB9 met behulp van:

**DB9 RJ11**

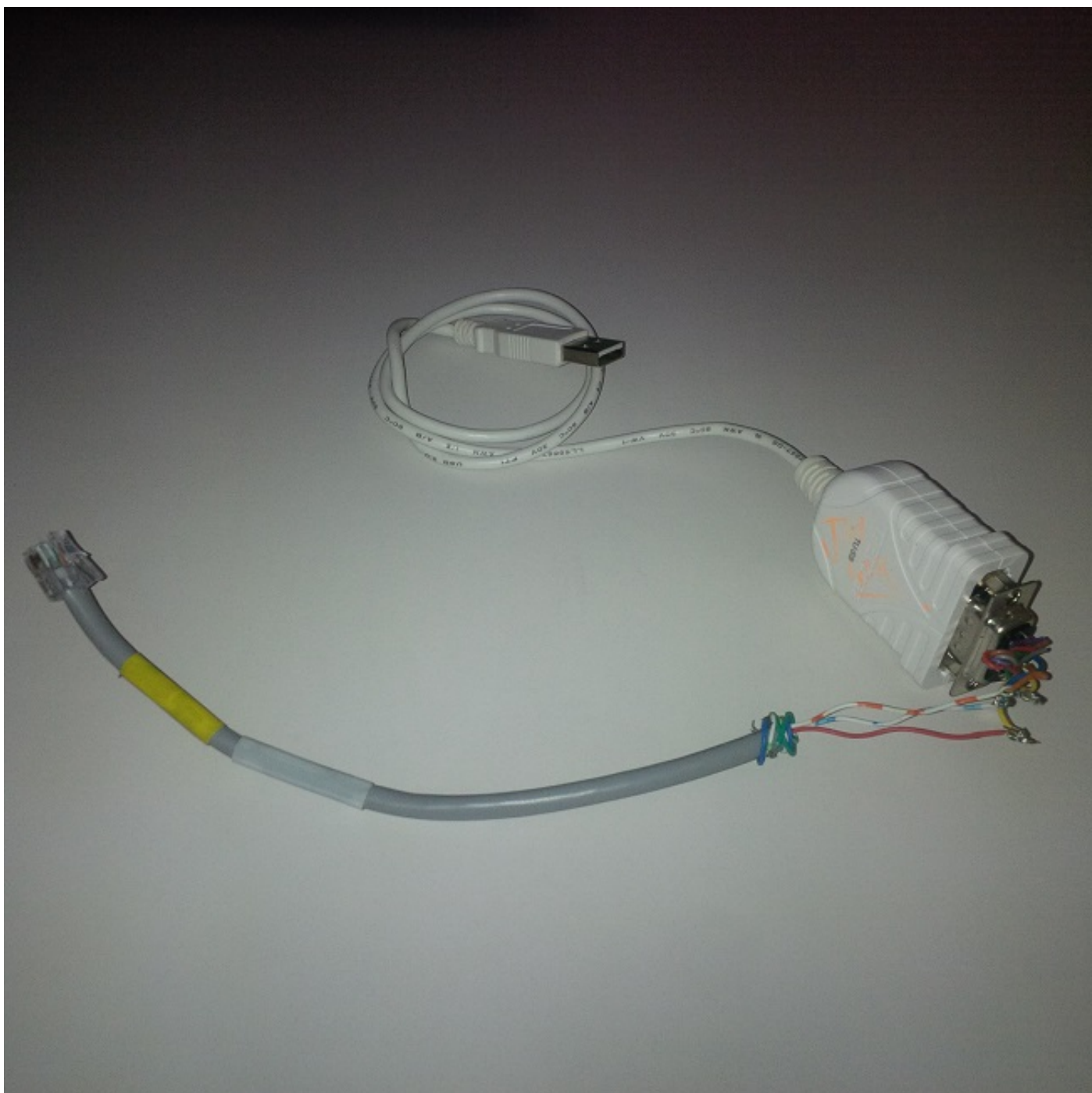
2	2
3	4
5	3

De afbeelding toont een vrouwelijke RJ 11-poorts en DB9-poort die samen zijn gepatcheerd.





In deze afbeelding wordt een kabel met zes pinnen aux poortcontrole getoond.

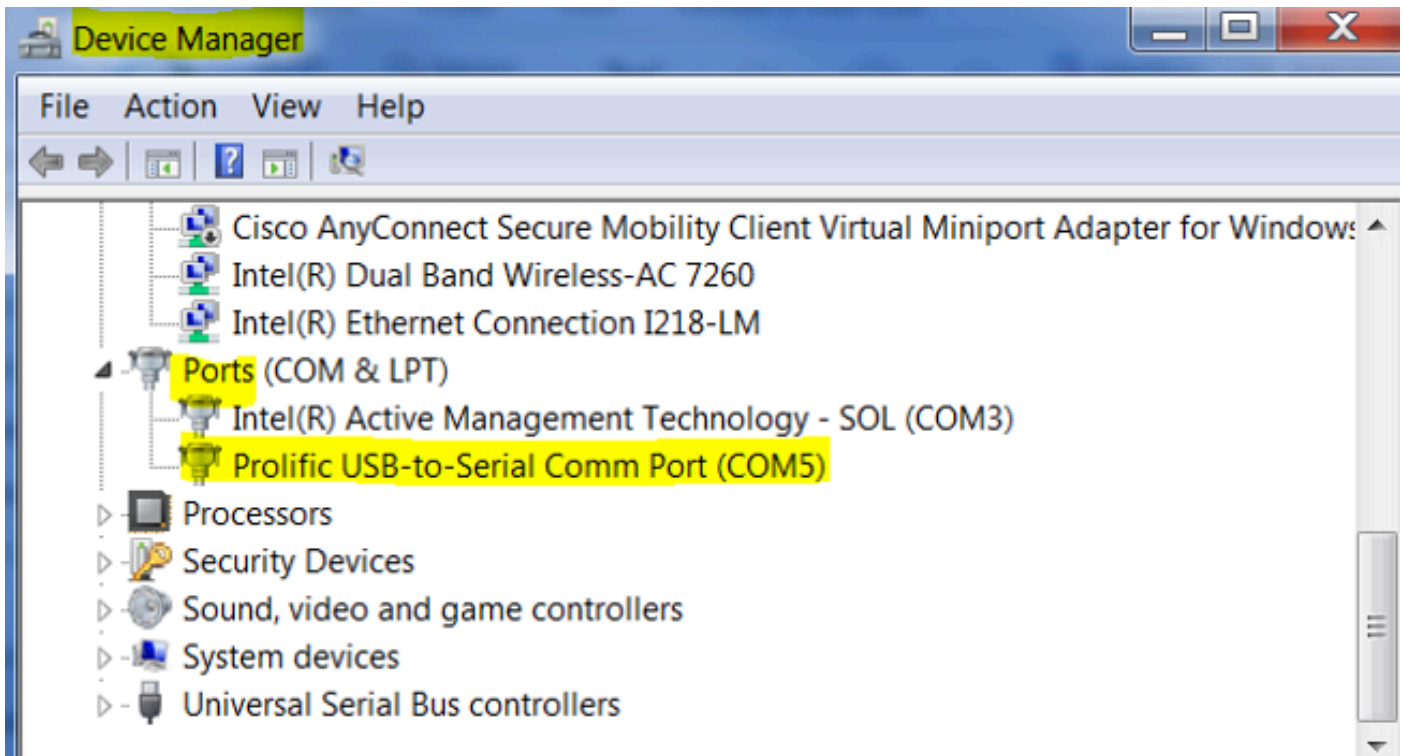


## Stappen om de console poortkabel te gebruiken

1. Sluit de juiste console kabel aan op de console van de telefoon AUX of USB, afhankelijk van het telefoonmodel.
2. Controleer voor USB-console poort of de USB-poort is ingeschakeld vanaf de Apparaatpagina van de telefoon in Call Manager.
3. Nadat u de kabel op de computer hebt aangesloten, dient u het COM-poortnummer voor deze verbinding te raadplegen door naar Apparaatbeheer te navigeren.

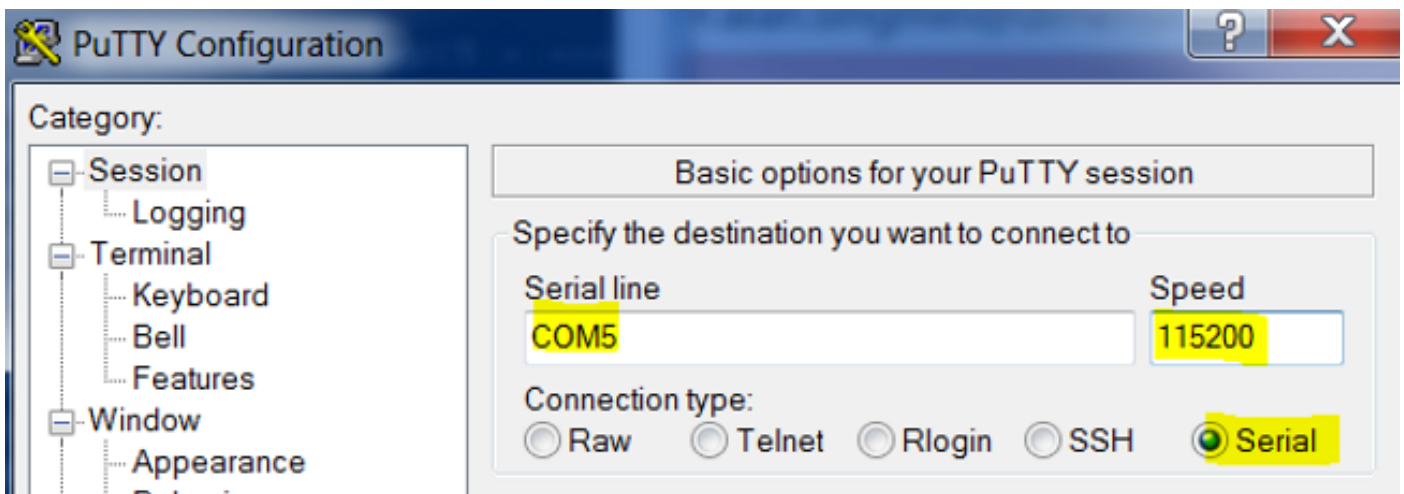
Opmerking: Als de kabel niet is gedetecteerd in Apparaatbeheer, zorg er dan voor dat de stuurprogramma's (indien aanwezig) van de USB seriekabel zijn geïnstalleerd.





4. Op de Windows-machine heeft u toegang tot de seriële poort door gebruik te maken van de PuTy-sessie met Baud Rate 15200.

5. Om elke specifieke debugs in eerste instantie in te schakelen, moet u deze configureren via de telefoondebug-modus door een SSH-toegang te gebruiken.



Dit beeld illustreert de defecten die op het consolescherm inloggen.

```
2994: WRN 18:05:04.624182 JVM: Startup Module Loader|cip.mmgr.dt:? - [MediaMgrSM]:
  Unhandled Event, State = StateHandsetOffHook Event = EventSetSpeakerMode
2995: NOT 18:05:04.664184 JVM: Startup Module Loader|cip.cfg.ConfigManager:? - ---
>ConfigManager PropertyChanged: device.callagent.callcount
2996: NOT 18:05:04.665787 JVM: Startup Module Loader|cip.cfg.ConfigManager:? - <--
-ConfigManager PropertyChanged: device.callagent.callcount
2997: NOT 18:05:04.667458 JVM: Startup Module Loader|cip.cfg.ConfigManager:? - ---
>ConfigManager PropertyChanged: device.callagent.callcount
2998: NOT 18:05:04.669038 JVM: Startup Module Loader|cip.cfg.ConfigManager:? - <--
-ConfigManager PropertyChanged: device.callagent.callcount
2999: WRN 18:05:06.358267 JVM: Startup Module Loader|cip.mmgr.dt:? - [MediaMgrSM]:
  Unhandled Event, State = StateOnHook Event = EventSetSpeakerModeOff
3000: WRN 18:05:06.359889 JVM: Startup Module Loader|cip.mmgr.dt:? - [MediaMgrSM]:
  Unhandled Event, State = StateOnHook Event = EventEndcall
3001: NOT 18:05:06.399467 JVM: Startup Module Loader|cip.cfg.ConfigManager:? - ---
>ConfigManager PropertyChanged: device.callagent.callcount
3002: NOT 18:05:06.401203 JVM: Startup Module Loader|cip.cfg.ConfigManager:? - <--
-ConfigManager PropertyChanged: device.callagent.callcount
```