

# Come personalizzare il cavo della console Cisco IP Phone

## Sommario

[Introduzione](#)

[Premesse](#)

[Tipi di porte console](#)

[Come realizzare il cavo della console](#)

[Cavo porta console USB](#)

[Cavo porta console AUX a sei pin](#)

[Passaggi per l'utilizzo del cavo della porta console](#)

[Discussioni correlate nella Cisco Support Community](#)

## Introduzione

In questo documento viene descritto come personalizzare il cavo della console di Cisco IP Phone in modo che venga usato per raccogliere le note telefonate direttamente dalla porta della console.

## Premesse

I log di debug telefonici vengono raccolti utilizzando tre metodi diversi:

- Registri console da accesso Web
- Modalità di debug con accesso SSH (Secure Shell)
- Porta console del telefono

Utilizzare solo i metodi uno e due quando il telefono è raggiungibile tramite IP. In alcuni casi, l'accesso Web o SSH non è disponibile a causa della sicurezza della rete, ad esempio nei criteri del firewall.

Il metodo 3 è utile negli scenari di risoluzione dei problemi, in cui i registri devono essere raccolti durante l'avvio del telefono o prima che l'IP del telefono sia raggiungibile, ad esempio:

- Risoluzione dei problemi relativi a DHCP telefonico
- problema VPN
- Problemi di raggiungibilità del telefono o di sicurezza.

In questi casi viene utilizzato un cavo console.

## Tipi di porte console

Sui telefoni Cisco sono disponibili due tipi di porte console, a seconda del modello.

- Porta console USB sui modelli 99XX e 8961

- Una porta AUX a sei pin sui modelli di telefoni 79XX e 78XX

**Nota:** Il telefono modello 88XX è dotato di porte USB e AUX.

## **Come realizzare il cavo della console**

I due tipi di cavi console sono:

1. Cavo porta console USB
2. Cavo porta AUX a sei pin

### **Cavo porta console USB**

Per realizzare il cavo della porta console USB, è necessario:

- Due cavi seriali USB (DB9), come mostrato in questa immagine:



- Un cavo seriale con porta femmina DB9 su ciascun lato, come mostrato nell'immagine:



Le porte seriali dei tre cavi devono essere interconnesse per completare il cavo, come mostrato in questa immagine:



Quando il cavo della porta console USB è collegato al telefono, sul telefono viene visualizzato un messaggio flash: Rilevato cavo console seriale USB.

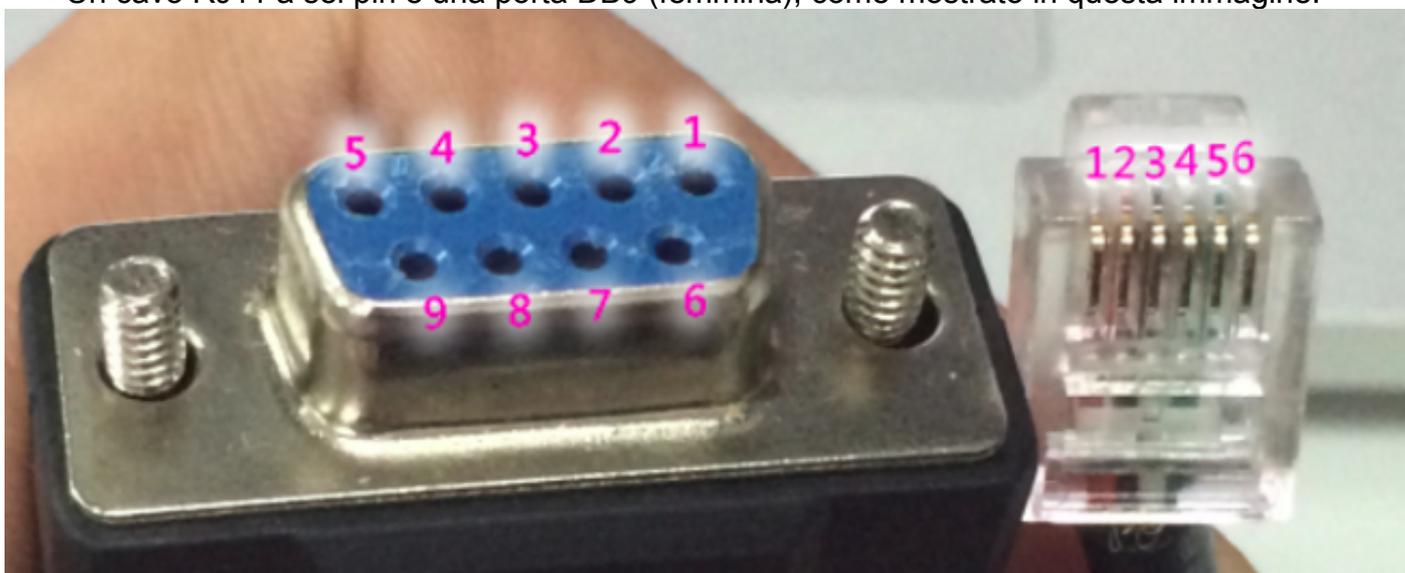
### **Cavo porta console AUX a sei pin**

Per realizzare un cavo a sei pin per porta AUX è necessario:

- Un cavo USB seriale (DB9) come mostrato nell'immagine:



- Un cavo RJ11 a sei pin e una porta DB9 (femmina), come mostrato in questa immagine:



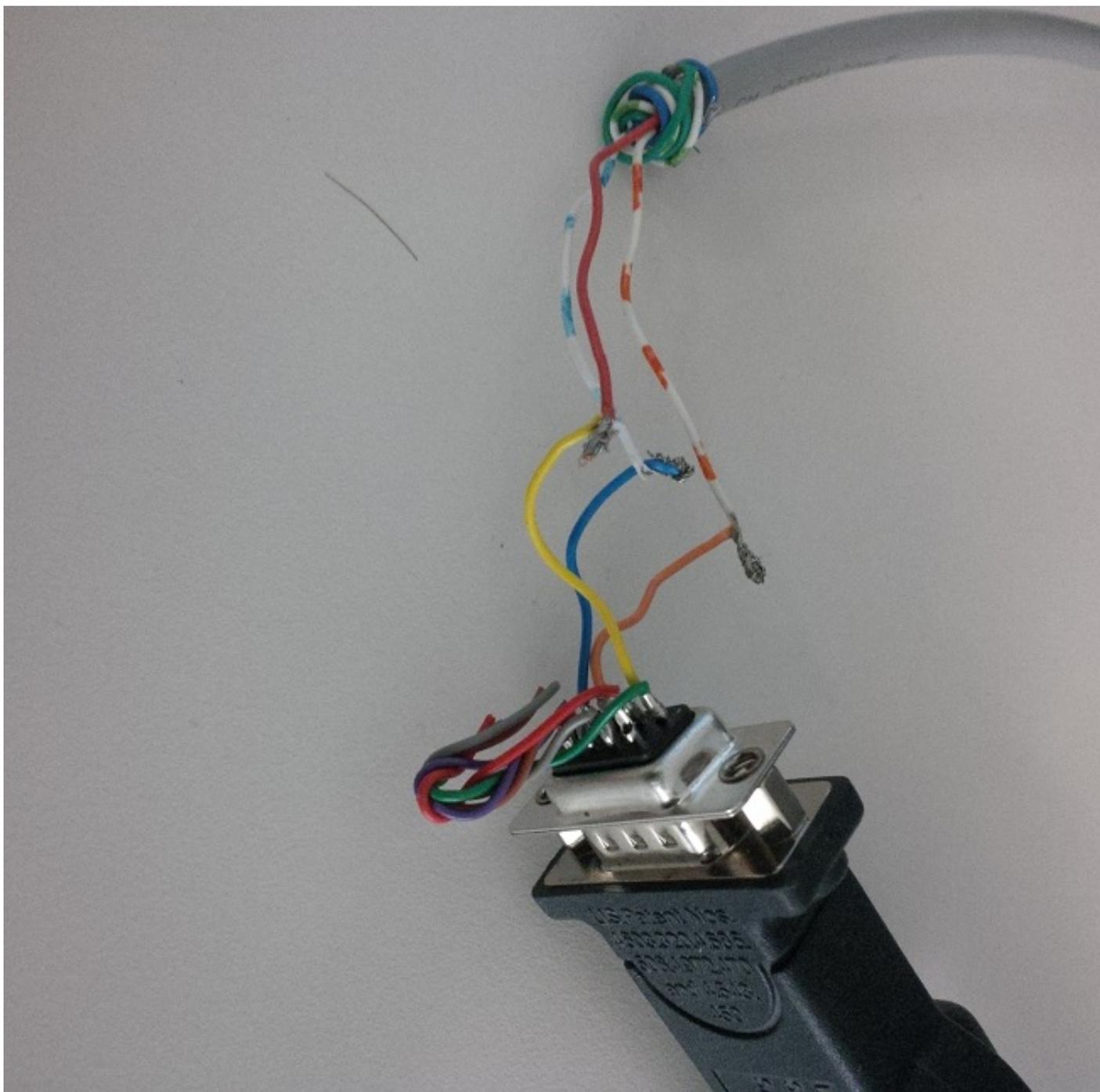
Sulla porta DB9, i numeri vengono indicati minuziosamente per ciascuna porta. Per la porta RJ11, il numero del cavo viene contato da sinistra mentre il clip è rivolto verso il basso.

Avvitare i cavi di RJ11 e DB9 utilizzando:

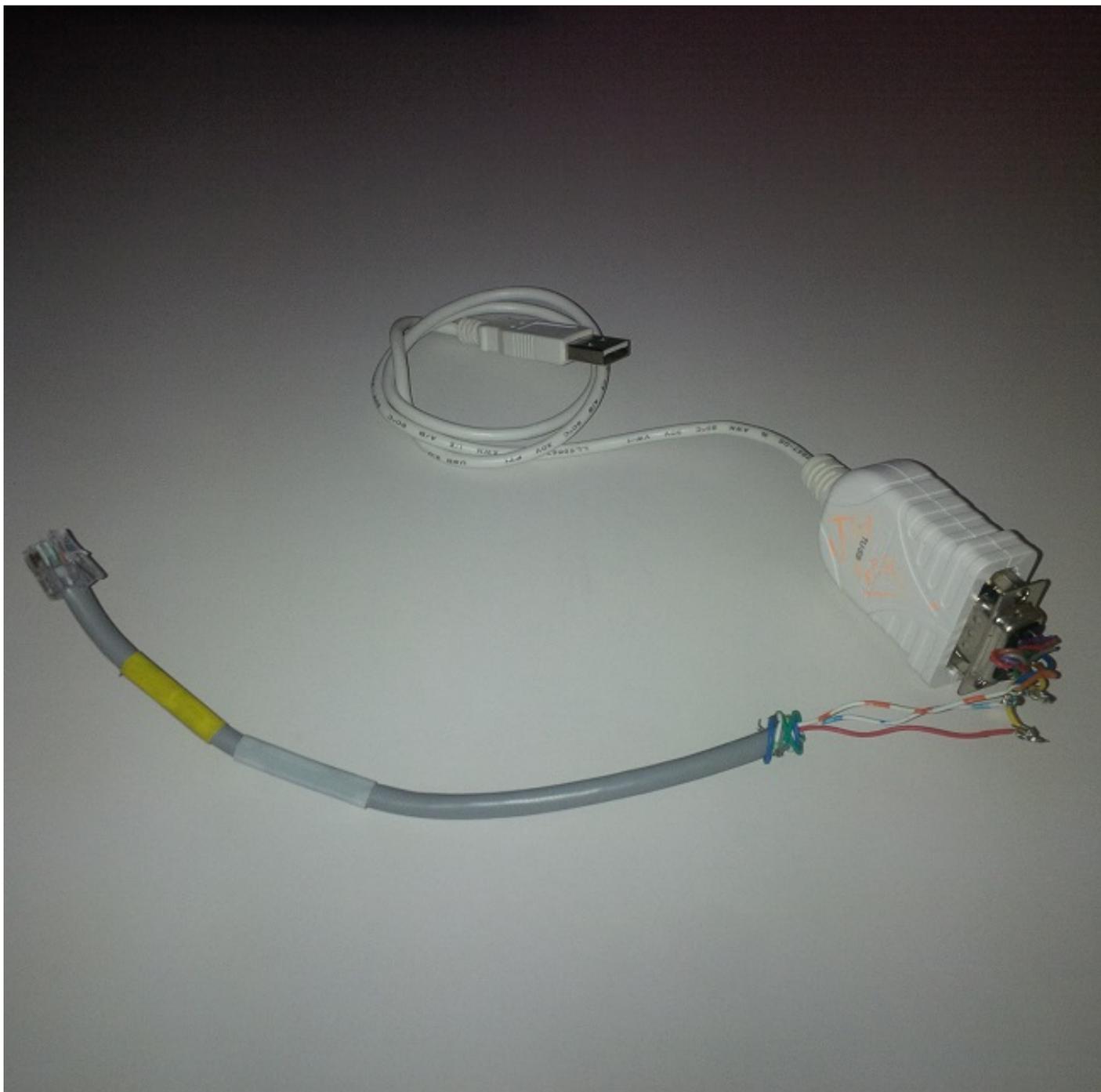
**DB9 RJ11**

2	2
3	4
5	3

Nell'immagine viene mostrata la porta RJ 11 e la porta femmina DB9 collegate insieme.



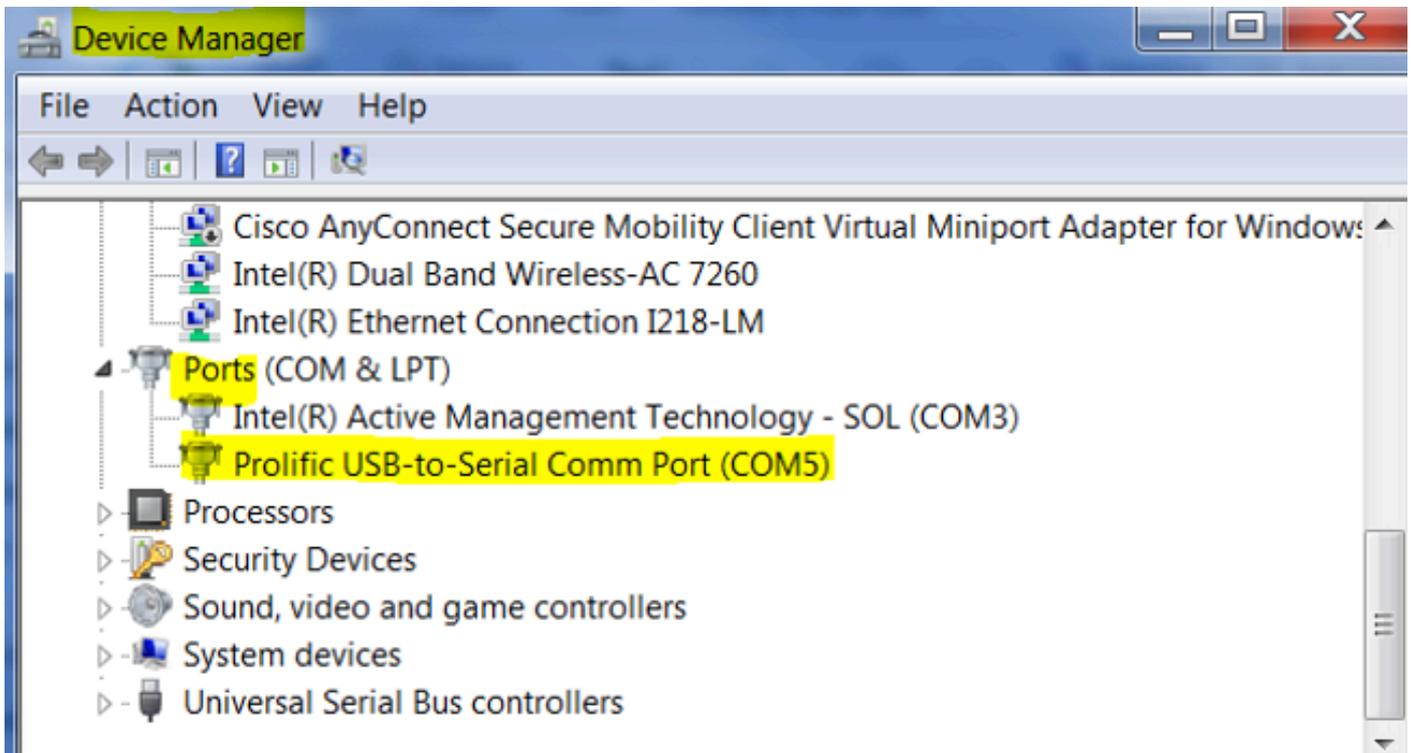
In questa immagine viene mostrato un cavo di controllo porta AUX a sei pin.



## Passaggi per l'utilizzo del cavo della porta console

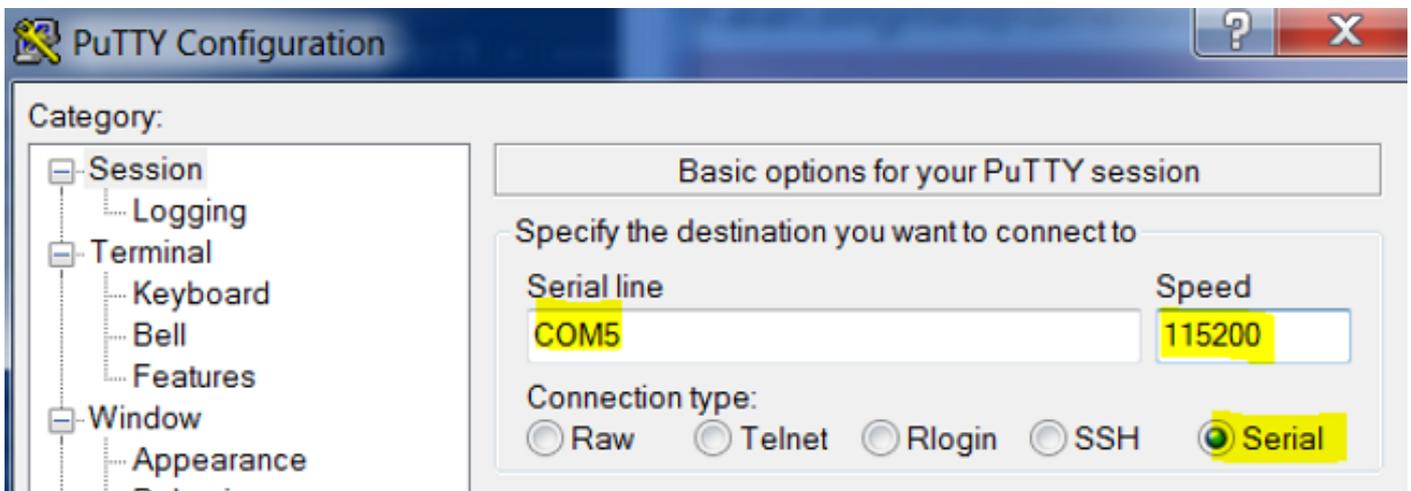
1. Collegare il cavo console appropriato alla porta console del telefono AUX o USB, a seconda del modello di telefono.
2. Per la porta console USB, verificare che la porta USB sia abilitata sul telefono dalla pagina Periferica di Gestione chiamate.
3. Dopo aver collegato il cavo al computer, individuare il numero di porta COM per la connessione, passando a Gestione periferiche.

**Nota:** Se il cavo non viene rilevato in Gestione periferiche, verificare che siano installati gli eventuali driver del cavo seriale USB.



4. Sul computer Windows, accedere alla porta seriale utilizzando la sessione PuTTY con velocità in baud 115200.

5. Per abilitare inizialmente eventuali debug specifici, eseguire la configurazione tramite la modalità di debug telefonico con un accesso SSH.



L'immagine mostra i debug che eseguono l'accesso alla schermata della console.

```
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
```