

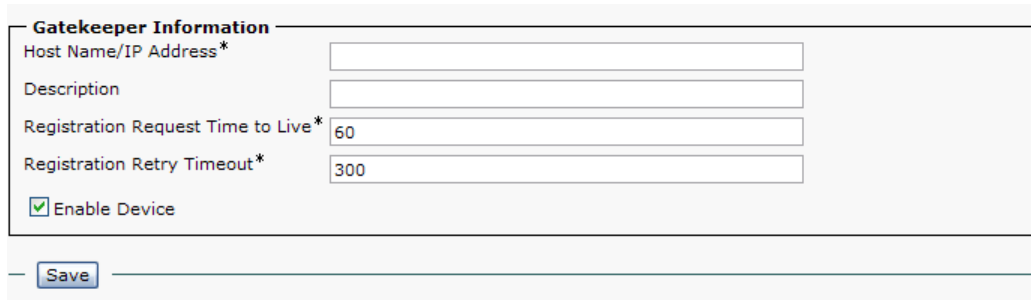
Goal

To route voice calls from our Cisco VoIP network to a Codian MCU 4205 Video bridge, via the GnuGK so that traveling users would be able to use their cell phones to dial into the Codian video conference bridge rooms and partake in voice conversation with the video conference users.

Cisco CallManager / CommunicationsManager Configuration Steps

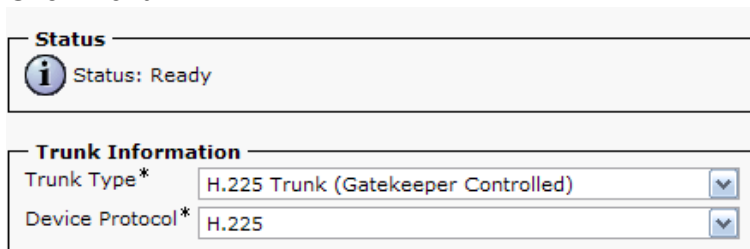
Note: This was written using version 6.1 of Cisco Communications Manager. Other versions should be very similar.

- 1) You must first add the GnuGK gatekeeper into Cisco CommManager.
 - To do this, go to Device > Gatekeeper and click 'Add New'.
 - Type in the IP address of the GnuGK. For example: 172.16.0.185
Important: The GnuGK should be on the same network as the Cisco Communications Manager server, or be able to route to it. Use a ping test.
 - Type in a description: GnuGK GateKeeper
 - Leave the last two fields as the default values and make sure that there is a check in the box to 'Enable Device'.
 - Click Save



The screenshot shows the 'Gatekeeper Information' configuration form in Cisco CallManager. It includes fields for 'Host Name/IP Address*', 'Description', 'Registration Request Time to Live*' (set to 60), and 'Registration Retry Timeout*' (set to 300). There is a checked checkbox for 'Enable Device'. A 'Save' button is at the bottom.

- 2) Next, build a trunk between CM and the new gatekeeper.
 - To do this, go to Device > Trunk and click 'Add New'.
 - Choose Trunk Type: *H.225 Trunk*
 - Device Protocol: *H.225*
 - Click *Next*



The screenshot shows the 'Status' and 'Trunk Information' configuration form. The 'Status' section shows 'Status: Ready' with an information icon. The 'Trunk Information' section has dropdown menus for 'Trunk Type*' (set to 'H.225 Trunk (Gatekeeper Controlled)') and 'Device Protocol*' (set to 'H.225').

- Device Name: *GnuGK_Trunk*
- Description: *GnuGK Trunk*
- Device Pool: *Default (Use whatever is appropriate)*
- Location: *Default (Use whatever is appropriate)*

- Media Termination Point Required – Unchecked
- Retry Video Call as Audio – Checked
- Wait for Far End H.245 Terminal Capability Set – Unchecked
- Scroll down to the Gatekeeper Information Section
 - Set the Gatekeeper Name to the gatekeeper that we added earlier
 - Terminal Type: *Gateway*
 - Technology Prefix: did not use one
 - Zone: did not use one
- Click *Save* button

Now you have the infrastructure built so that you route calls to the GnuGK Gatekeeper. Now we need to setup a route pattern on our Cisco Communications Manager to route calls to this new trunk.

3) Setup a Route Pattern

- Go to Call Routing > Route/Hunt > Route Pattern
- Click *Add New*
- Fill in the appropriate fields
 - Route Pattern – Number which you would like to route to the GnuGK (in our case, this was the number of the video conference bridge room: 9010)
 - Route Partition – Default (Use whatever applies in your situation)
 - Description: *Number 9010 routing to GnuGK*
 - Gateway/Route List: *Choose the GnuGK trunk that we created above*
 - Click *Save* button

Now, whenever anyone dials the number that you created in your route pattern (9010), it will send the call to the GnuGK and then the GnuGK will decide which endpoint to route that call to. Now we just need to configure the Codian to register its Conference Rooms as endpoints on our Gnu Gatekeeper.

Codian MCU 4205 Video Conference Bridge – Configuration Steps

1) Disable the built-in gatekeeper function of the Codian

- Click Gatekeeper and then choose Disabled from the drop-down box. Click Apply.

The screenshot shows a web interface with a top navigation bar containing links: Home, Status, Network, Settings, Conferences, Users, Endpoints, Gateways, and Gatekeeper. Below the navigation bar, the breadcrumb path is 'Home > Built-in gatekeeper'. The main content area is titled 'Built-in gatekeeper' and 'Gatekeeper configuration'. It features two sections: 'Configuration' and 'Neighbor configuration'. The 'Configuration' section has a 'Status' dropdown menu set to 'Disabled'. The 'Neighbor configuration' section includes two input fields for 'Neighbor gatekeeper 1' and 'Neighbor gatekeeper 2', both containing redacted IP addresses. Below these are three checkboxes: 'Accept LRQs' (checked), 'Forward LRQs for unknown IDs' (set to 'Enabled, using received return address' via a dropdown), and 'Accept LCFs from non-neighbors' (checked). An 'Apply changes' button is located at the bottom left of the configuration area.

Home Status Network Settings Conferences Users Endpoints Gateways Gatekeeper

Home > Built-in gatekeeper

Built-in gatekeeper

Gatekeeper configuration

Configuration

Status Disabled

Neighbor configuration

Neighbor gatekeeper 1 [Redacted]

Neighbor gatekeeper 2 [Redacted]

Accept LRQs ☒

Forward LRQs for unknown IDs Enabled, using received return address

Accept LCFs from non-neighbors ☒

Apply changes

- 2) Configure the Codian to register to the GnuGK Gatekeeper
- Click Settings > Gatekeeper tab
 - Setup the gatekeeper address as the IP address of the GnuGK.
 - My settings are below in the picture

Note: In the 'Send resource availability indications' section, make sure to indicate how many licensed ports that you have available on the Codian so that the GnuGK does not allow more video calls than your Codian can support.

When the Codian registers successfully to the GnuGK, the status area at the bottom of the page will say 'Registered'

H.323 gatekeeper settings	
H.323 gatekeeper usage	Enabled <input type="button" value="v"/>
H.323 gatekeeper address	172.16.0.185
Gatekeeper registration type	Terminal / gateway <input type="button" value="v"/>
Ethernet port association	<input checked="" type="checkbox"/> Port A <input checked="" type="checkbox"/> Port B
(Mandatory) H.323 ID to register	Codian MCU 4205
Use password	<input type="checkbox"/> Password: <input type="text"/>
Prefix for MCU registrations	<input type="text"/>
MCU service prefix	<input type="text"/> (optional)
ID registration for scheduled conferences	Enabled <input type="button" value="v"/>
Send resource availability indications	<input checked="" type="checkbox"/> Thresholds: <input type="text"/> conferences <input type="text" value="12"/> video ports
Current status	
H.323 gatekeeper status	registered with 172.16.0.185
Registered address	172.16.0.188:1720
Alternate gatekeepers available	0
Resource availability status	resources available
Number of active registrations	7 details
H.323 ID registration	Codian MCU 4205 registered
MCU service prefix	n/a

- 3) Now, configure the Codian conference rooms to register their room ID's to the GnuGK (So that you can call the room ID numbers from your Cisco Phone).
- Click Conferences and then create a new conference or select an existing one.
 - Once you click the conference, click on the 'Configuration' tab to edit the properties of the conference.
 - Give the conference a 'Numeric ID' that matches the number of the route pattern you created earlier on in the Cisco CommManager, in our case: 9010.
 - Then click the checkbox under 'Numeric ID Registration: H.323 Gatekeeper'

Participants	Configuration	Custom layout	Statistics	Send me
<h2>Conference "Room A - 9010"</h2> <p>Update conference</p>				
<h3>Parameters</h3>				
Name <input type="text" value="Room A - 9010"/>				
Description <input type="text"/> (optional)				
Numeric ID <input type="text" value="9010"/> (optional)				
PIN <input type="text"/> (optional)				
Guest numeric ID <input type="text"/> (optional)				
Guest PIN <input type="text"/> (optional)				
Numeric ID registration <input checked="" type="checkbox"/> H.323 gatekeeper <input type="checkbox"/> SIP registrar				
When only guests remain <input type="text" value="Disconnect all participants"/>				
Floor and chair control <input type="text" value="Allow floor control only"/>				
Owner <input type="text" value="admin"/>				
Visibility <input type="text" value="Public"/>				
Layout control via FECC / DTMF <input type="text" value="Enabled"/>				
Invite pre-configured participants <input type="text" value="At the start of the conference"/>				
Mute on join <input type="checkbox"/> Audio <input type="checkbox"/> Video				
Streaming <input type="text" value="Unicast"/>				
Content channel video <input type="text" value="Content not enabled"/>				
Content contribution from endpoints <input type="text" value="Content not enabled"/>				
Maximum video participants <input type="text"/> (optional - 0 currently in use)				

- 4) OR, if you'd rather have your Cisco route pattern dial into an auto-attendant, simply setup an auto-attendant on your Codian and have it register its numeric ID to the GnuGK just as done above.
- To create an auto attendant on the codian, click Conferences > Auto Attendants Tab
 - Click 'Add New Auto Attendant'
 - Give it a name and a numeric ID: 9010 (or whatever your Cisco route pattern is)
 - Select the checkbox for 'Numeric ID Registration – H.323 Gatekeeper'
 - Click Add Auto Attendant button

Summary

So now when we pick up our Cisco voIP phone and dial 9010, the Cisco Communications Manager server has a route pattern that matches 9010. It then knows to route the call to the new trunk that we created to the GnuGK Gatekeeper. Once that call is handed off to GnuGK, it attempts to match the called number (9010) to one of the registered devices/endpoints on the GnuGk. Because we told the Codian to register its numeric ID's to the GnuGK, the GnuGK can now route that call to the Codian where it belongs.

Then the Codian answers the call from the Cisco VoIP phone and the caller can now listen in and participate to a video conference from the road or from their desk.

Appendix

Here is my GnuGK configuration file:

```
; this is a basic startup configuration for GnuGk
; it allows anyone to register and make calls

[Gatekeeper::Main]
FortyTwo=42
Name=GnuGk
EndpointSuffix=_gnugk
TimeToLive=60
; change this to 1 or 2, if you want CDRs and RAS messages to be printed on the
status port
StatusTraceLevel=0
; enable these options if your endpoints use broadcast and/or multicast to
discover the gatekeeper
UseBroadcastListener=0
UseMulticastListener=0

; restrict access to the status port by an IP address
[GkStatus::Auth]
rule=explicit
; add more entries, if you access the status port from other hosts
127.0.0.1=allow
default=forbid
Shutdown=allow

[RoutedMode]
; enable gatekeeper signaling routed mode, route H.245 channel only if neccessary
(for NATed endpoints)
GKRouted=1
H245Routed=0
AcceptNeighborCalls=1
AcceptUnregisteredCalls=1
RemoveH245AddressOnTunneling=1
RemoveCallOnDRQ=0
DropCallsByReleaseComplete=1
SendReleaseCompleteOnDRQ=0
SupportNATedEndpoints=1
TranslateFacility=1

; proxy calls only for NATed endpoints
[Proxy]
Enable=0
; if port forwarding is correctly configured for each endpoint, you can disable
ProxyForNAT
ProxyForNAT=1
ProxyForSameNAT=0

[RasSrv::RRQFeatures]
; endpoint identifiers are assigned by the gatekeeper
AcceptEndpointIdentifier=0
; you may want to disable this, if you want to control gateway prefixes from the
config
AcceptGatewayPrefixes=1
```

```
[CallTable]
; don't print CDRs for neighbor calls to the status port
GenerateNBCDR=0
; print CDRs for unconnected calls to the status port
GenerateUCCDR=1
```

;I don't believe the following configuration details were even necessary to get this solution working, however I ;did have them in my configuration.

```
[RasSrv::Neighbors]
```

```
;This is the IP Address of our Cisco CallManager
GK2=192.168.1.20;*
```

```
[Endpoint]
```

```
;This is the IP Address of our Cisco CallManager
Gatekeeper=192.168.1.20
Type=Gateway
RRQRetryInterval=10
Prefix=*
TimeToLive=900
Discovery=0
```