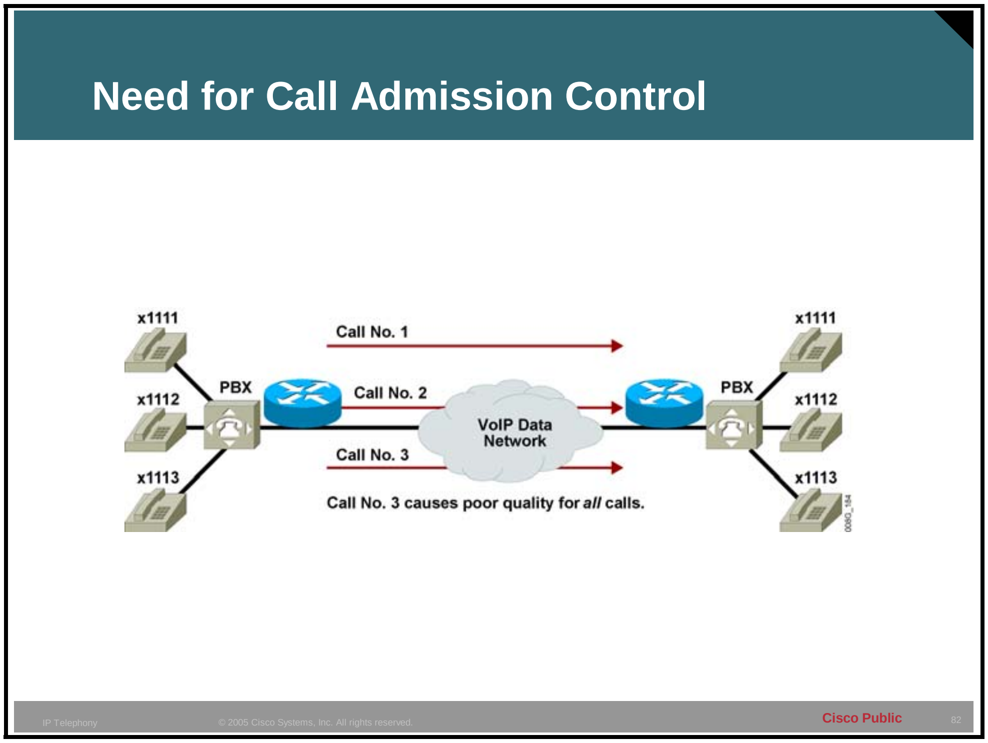
Configuring CAC

Need for CAC

This topic explains why CAC is needed.



CAC is a concept that applies to voice traffic only, not data traffic. If an influx of data traffic oversubscribes a particular link in the network, queuing, buffering, and packet drop decisions resolve the congestion. The extra traffic is simply delayed until the interface becomes available to send the traffic, or, if traffic is dropped, the protocol or the end user initiates a timeout and requests a retransmission of the information.

Because real-time traffic is sensitive to latency and packet loss, resolving network congestion when real-time traffic is present will jeopardize the QoS. For real-time delay-sensitive traffic such as voice, it is better to deny network access under congestion conditions than to allow traffic on the network to be dropped and delayed. Dropped or delayed network traffic causes intermittent impaired QoS and results in customer dissatisfaction.

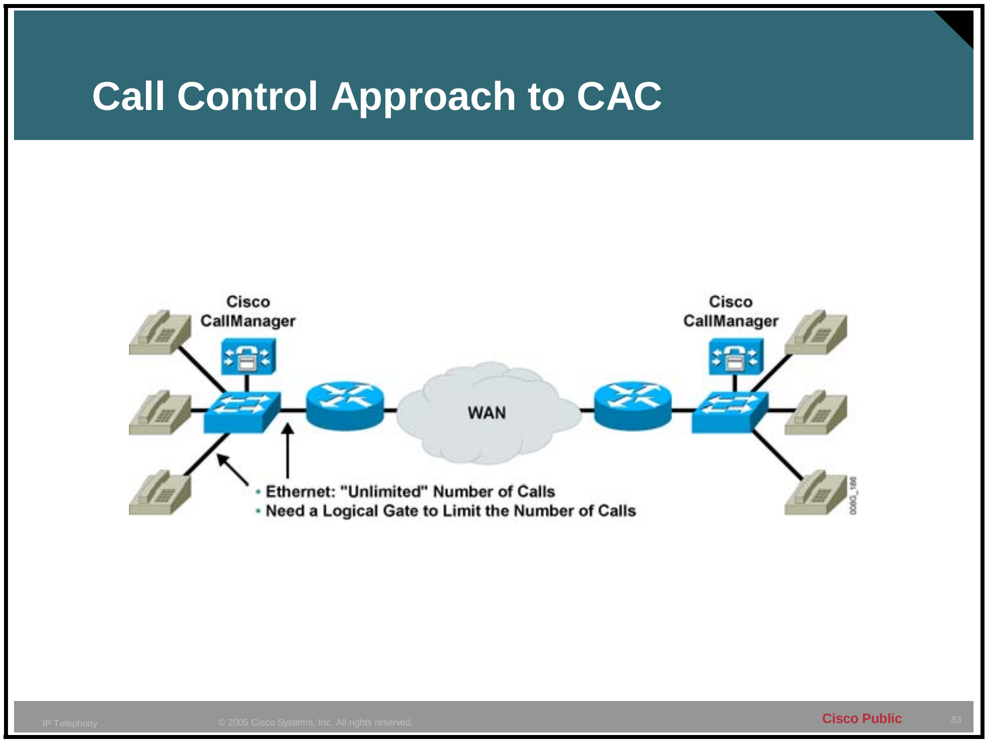
CAC is a determining and informed decision that is made before a voice call is established. CAC is based on whether the required network resources are available to provide suitable QoS for the new call.

Example: CAC Applied

CAC mechanisms extend the capabilities of QoS tools to protect voice traffic from the negative effects of other voice traffic and to keep excess voice traffic off the network. The figure illustrates the need for CAC. If the WAN access link between the two PBXs has the bandwidth to carry only two VoIP calls, admitting the third call impairs the voice quality of all three calls.

CAC as Part of Call Control Services

This topic describes CAC as a function of call control services.



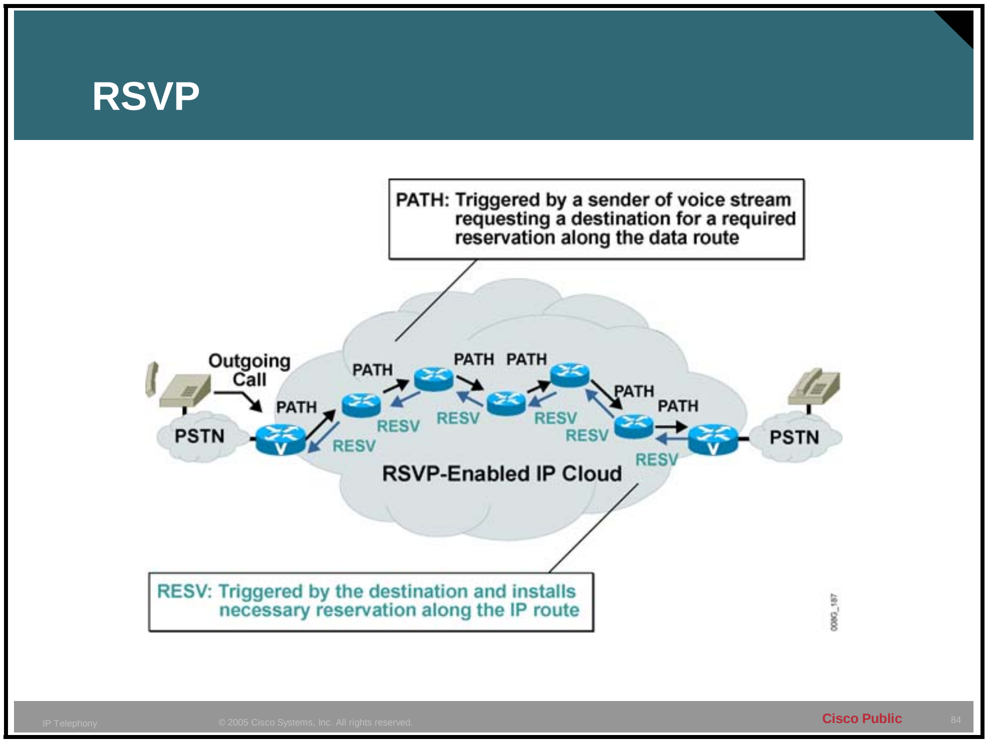
CAC, as part of call control services, functions on the outgoing gateway. CAC bases its decision on nodal information, such as the state of the outgoing LAN or WAN link. If the local packet network link is down, there is no point in executing complex decision logic based on the state of the rest of the network, because that network is unreachable. Local mechanisms include configuration items that disallow all calls that exceed a specified number.

Example: Call Control CAC

If the network designer already knows that bandwidth limitations allow no more than five calls across the outgoing WAN link, then the local node can be configured to allow no more than five calls. You can configure this type of CAC on outgoing dial peers.

RSVP

This topic describes Resource Reservation Protocol (RSVP).



RSVP is the only CAC mechanism that makes a bandwidth reservation and does not make a call admission decision based on a best guess before the call is set up. This gives RSVP the unique advantage of not only providing CAC for voice, but also guaranteeing the QoS against changing network conditions for the duration of the call. The RSVP reservation is made in both directions because a voice call requires a two-way speech path and bandwidth in both directions.

The terminating gateway ultimately makes the CAC decision based on whether both reservations succeed. At that point, the H.323 state machine continues with either an H.225 Alerting/Connect (the call is allowed and proceeds), or with an H.225 Reject/Release (the call is denied). The RSVP reservation is in place by the time the destination phone starts ringing and the caller hears ringback.

RSVP has the following important differences from other CAC methods discussed in this lesson:

- The ability to maintain QoS for the duration of the call.
- An awareness of topology. In concept, the RSVP reservation installs on every interface that the call will traverse through the network. RSVP ensures bandwidth over every segment without any requirement to know the actual bandwidth provisioning on each interface or the path on which the routing protocols direct the packets. RSVP, therefore, adjusts automatically to network configuration changes, and no manual calculations are necessary to keep different aspects of the configuration synchronized.
- To function correctly, RSVP is dependent on the correct configuration for all devices in the network. (It can have a scaling issue depending on how the network is designed.)

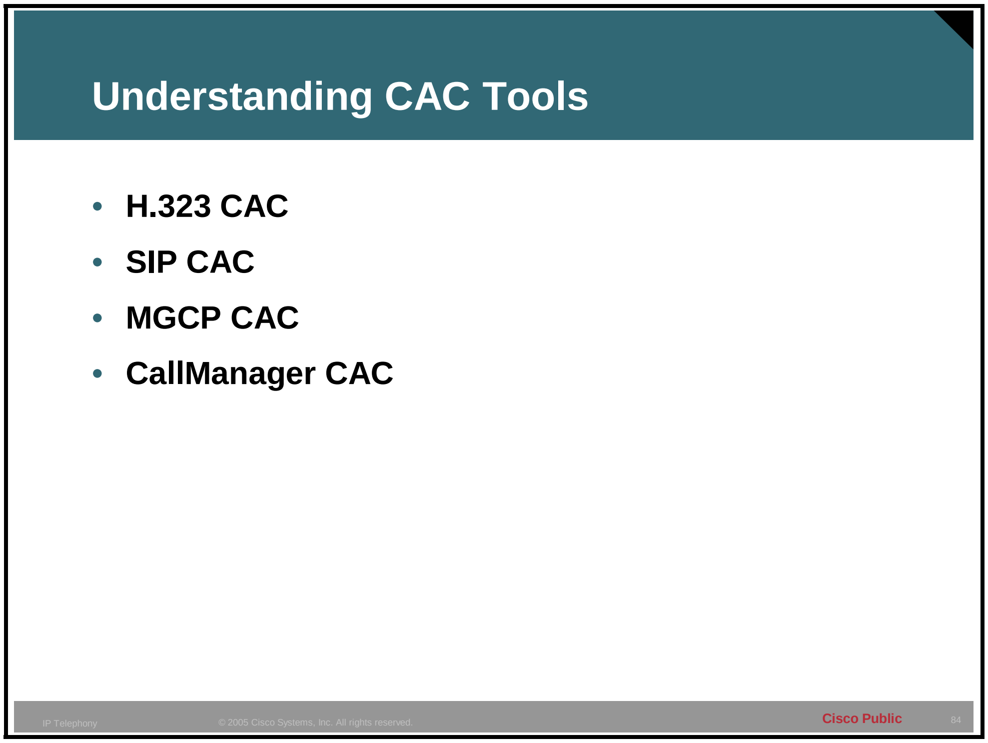
- RSVP provides end-to-end reservation per call and has visibility for that call only. RSVP is unaware of how many other calls are active from a site or across an interface, or the source or destination of any other call.

Example: RSVP

Configuring RSVP in Cisco routers allows the administrator to limit the amount of bandwidth requested per call and the total amount of bandwidth allowed for all calls. This configuration is entered directly against the interface that will permit or deny the calls. The configuration also requires RSVP to be configured on the dial peers for the calls that will be managed by RSVP.

Understanding CAC Tools

This topic describes CAC tools available for various protocols and systems.



As the many interesting aspects of CAC on packet networks have been considered, several different solutions have come into prominence. None of them solves the entire problem, but they all are useful to address a particular aspect of CAC. Unlike circuit-based networks, which reserve a free DS0 time slot on every leg of the path the call will take, determining whether a packet network has the resources to carry a voice call is not a simple undertaking.

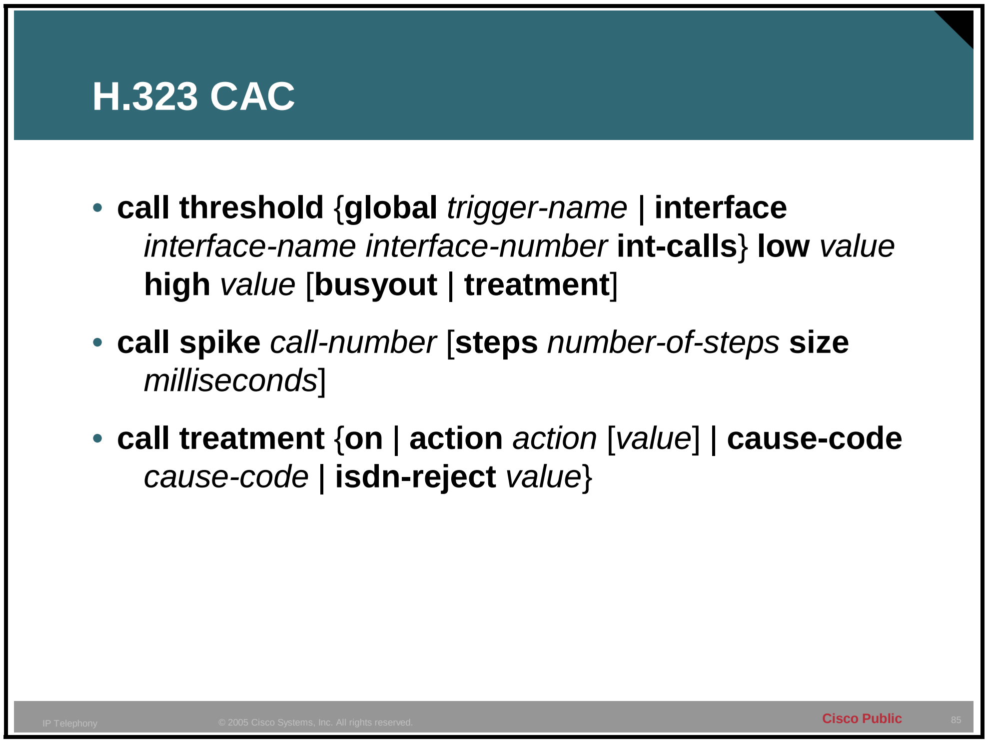
There are four areas in which CAC may be implemented. These areas are:

- H.323 CAC
- SIP CAC
- MGCP CAC
- CallManager CAC

Each area is associated with a specific protocol or system. Each of these areas will be explored in the following figures.

H.323 CAC

This topic describes the configuration options available for H.323 CAC.



The CAC for the H.323 VoIP gateways feature allows you to configure thresholds for local resources, memory, and CPU resources.

With the **call threshold** command, you can configure two thresholds, high and low, for each resource. Call treatment is triggered when the current value of a resource exceeds the configured high. The call treatment remains in effect until the current resource value falls below the configured low. Having high and low thresholds prevents call admission flapping and provides hysteresis in call admission decision making.

With the **call spike** command, you can configure the limit for incoming calls during a specified time period. A call spike is the term for when a large number of incoming calls arrive from the PSTN in a very short period of time; for example, 100 incoming calls in 10 milliseconds.

With the **call treatment** command, you can select how the call should be treated when local resources are not available to handle the call. For example, when the current resource value for any one of the configured triggers for call threshold has exceeded the configured threshold, the call treatment choices are as follows:

- **Time-division multiplexing (TDM) hairpinning:** Hairpins the calls through the POTS dial peer
- **Reject:** Disconnects the call
- **Play message or tone:** Plays a configured message or tone to the user

To enable the global resources of this gateway, use the **call threshold** command in global configuration mode. To disable this command, use the **no** form of this command.

call threshold {**global** *trigger-name* | **interface** *interface-name interface-number* **int-calls**} **low** *value* **high** *value* [**busyout** | **treatment**]

no call threshold {**global** *trigger-name* | **interface interface-name int-calls**}

Call Threshold Commands

Command	Description
global *trigger-name*	Specifies the global resources on the gateway The *trigger-name* arguments are as follows: ■ **cpu-5sec:** CPU utilization in the last 5 seconds ■ **cpu-avg:** Average CPU utilization ■ **io-mem:** IO memory utilization ■ **proc-mem:** Processor memory utilization ■ **total-calls:** Total number of calls. The valid range is from 1 to 10,000. ■ **total-mem:** Total memory utilization
interface *interface-name interface-number*	Specifies the gateway. The types of interfaces and their numbers will depend upon the configured interfaces.
int-calls	Number of calls through the interface. The valid range is from 1 to 10,000 calls.
low *value*	Value of low threshold. The valid range is from 1 to 100 percent for the utilization triggers.
high *value*	Value of high threshold. The valid range is from 1 to 100 percent for the utilization triggers.
busyout	(Optional—global only) Automatically busies out the T1/E1 channels if the resource is not available.
treatment	(Optional—global only) Applies call treatment from session application if the resource is not available.

To configure the limit of incoming calls in a short period of time, use the **call spike** command in global configuration mode. To disable this command, use the **no** form of this command. The **call spike** command uses a sliding window to determine the period in which the spike is limited. The sliding window period is defined using the **size** command, with valid ranges from 100 to 250 ms. If a longer spike period is desired, the **steps** command is used as a multiplier for the **size** command. For example, if the **steps** were set to 2 and the **size** was set to 250, the spike period would be 500 ms.

call spike *call-number* [**steps** *number-of-steps* **size** *milliseconds*]

no call spike

Call Spike Commands

Command	Description
call-number	Incoming call numbers for spiking threshold; valid range is from 1 to 2,147,483,647
steps *number-of-steps*	(Optional) Number of steps; valid range is from 3 to 10
size *milliseconds*	(Optional) Step size in milliseconds; valid range is from 100 to 2000

To configure how calls should be processed when local resources are unavailable, use the **call treatment** command in global configuration mode. To disable the call treatment triggers, use the **no** form of this command.

> **call treatment** {**on** | **action** *action* [*value*] | **cause-code** *cause-code* | **isdn-reject** *value*}
>
> **no call treatment** {**on** | **action** *action* [*value*] | **cause-code** *cause-code* | **isdn-reject** *value*}

Call Treatment Commands

Command	Description
on	Enables call treatment from default session application
action *action*	Action to take when call treatment is triggered. The *action* argument has the following possible values: ▪ **hairpin:** Hairpin ▪ **playmsg:** Specifies the audio file to play (URL) ▪ **reject:** Disconnect the call and pass down cause code
value	(Optional) (For the *action* playmsg argument only) Specifies the audio file to play; URL format
cause-code *cause-code*	Specifies reason for disconnect to caller The *cause-code* argument can have the following values: ▪ **busy:** Indicates that gateway is busy ▪ **no-QoS:** Indicates that the gateway cannot provide QoS ▪ **no-resource:** Indicates that the gateway has no resources available
isdn-reject *value*	Selects the ISDN reject cause-code. The *value* argument has the following: ▪ 34–47 (ISDN cause code for rejection)

ISDN Cause Codes

Cause No.	Description	Function
34	No circuit available (circuit/channel congestion)	Indicates that there is no channel available to handle the call.
38	Net out of order	Indicates that the network is not functioning properly and it is likely to last a long time. Re-attempting the call is not likely to be successful.
41	Net problem, redial (temporary failure)	Indicates that the network is not functioning properly and it is not going to last a long time. Re-attempting the call is likely to be successful.
42	Net busy, redial (switching equipment congestion)	Indicates that the switching equipment is experiencing high traffic load.
43	Access/user information discarded	Indicates that the network is unable to deliver user information to the remote users as was requested.
44	No channel available (requested circuit/channel not available)	Indicates that the circuit or channel indicated by the requesting side cannot be used by the other side of the interface.
47	Resource unavailable/new destination	Indicates a resource unavailable event only when no other cause in the resource unavailable class applies.

Example: H.323 CAC Configuration

The following example will busyout the total-calls resource if 5 (low) or 5000 (high) is reached:

```
call threshold global total-calls low 5 high 5000 busyout
```

The following example enables thresholds of 5 (low) and 2500 (high) for interface calls on interface Ethernet 0:

```
call threshold interface Ethernet 0 int-calls low 5 high 2500
```

The following example will busyout the average CPU utilization if 5 percent (low) or 65 percent (high) is reached:

```
call threshold global cpu-avg low 5 high 65 busyout
```

The following configuration of the **call spike** command has a call number of 30, 10 steps, and a step size of 2000 milliseconds:

```
call spike 30 steps 10 size 2000
```

The following example enables the call treatment feature with a hairpin action:

```
call treatment on
call treatment action hairpin
```

The following example displays proper formatting of the **playmsg action** keyword:

```
call treatment action playmsg tftp://keyer/prompts/conjestion.
au
```


Note The **congestion.au** file plays when local resources are not available to handle the call.

The following example configures a call treatment cause-code to display no-QoS when local resources are unavailable to process a call:

```
call treatment cause-code no-qos
```