NetQoS
Performance Experts

# VoIP: Do You See What I'm Saying?

## Managing VoIP Quality of Experience on Your Network

NetQoS, Inc.

# Chapter 2 - VoIP Call Setup Performance

Your experience with any phone system begins when you pick up the handset. You may then press a button to talk or take some other step to get a dial tone. Or you may skip the dial tone altogether and go directly to dialing the phone number you want to call. As new unified communications dashboard applications roll out, you may just click on an icon to place a call to another person. But regardless of the actions you take to place a call, the quality of experience that you have with that phone system is largely shaped by your perception of the availability and call quality that the system provides. From a user's perspective, phone system availability can be summarized with the following basic criteria:

- Do you get a dial tone when you pick up the phone?
- Does the call connect successfully?
- If so, does it connect within a reasonable amount of time?

If you pick up the phone and don't receive a dial tone within several seconds, then you'll likely hang up, thinking the phone system is down. If you dial a phone number and don't hear ringing or a busy signal within several seconds, then your perception is that you can't make calls. If the call fails to connect and you hear a fast busy signal, then you'll likely think that the call did not go through and that you have to try again. All of these experiences – getting a dial tone, connecting the call, ringing the other party – are dependent on the performance of the call setup protocols in a VoIP system.

The PSTN has done a good job of shaping our expectations for call setup performance. In fact, over the years, call setup time has steadily decreased. Eyers and Schulzrinne point out that call setup time has dropped steadily over the last 80 years, from a high of 4 minutes in 1923, to 1.2 minutes in 1928, down to 10.9 seconds in 1978, and to less than 2.5 seconds in 1998 [1]. So as we've come to expect high performance for call setup from the PSTN, what happens on a VoIP network? Can this same level of call setup performance be obtained? Should it even be a goal? The answer is yes, but it takes a certain amount of both attention and management.

In the previous chapter, we introduced some of the key call setup protocols like SIP, MGCP, H.323, and SCCP. If you are not familiar with the basics of these protocols, we encourage you to take a look at Chapter 1. In the present chapter, we will discuss the performance characteristics of the call setup protocols and how you can tune them to ensure optimal user experience.

First, it's important to understand that in any VoIP system, there are a couple of different call types that involve different components and protocols. The interaction of calls of a certain type with your components and the protocols they "speak" can affect the call setup performance that users experience, for better or worse.

## OnNet Calls

An OnNet call is a call that takes place between two IP phones on the same logical network. In this scenario, the IP phones use call setup protocols like SIP or SCCP to interact with a call server that sets up and takes down each phone call. These calls are typically "all IP," meaning they are carried on the IP network and do not have to go out to the PSTN.

For an OnNet call, several events have to take place during the call setup phase to provide good call performance. The simplest scenario is when a call is made between two IP phones that are known by the same call server or by a single cluster of call servers (intracluster calls).

## Intracluster Calls

To enable a successful intracluster call, the phone and its call server need to communicate with each other via a call setup protocol. In the case of a Cisco IP phone, as soon as it starts up, the phone establishes a TCP connection with the call server, a phase that's often referred to as the registration process. The connection between phone and call server will be long-lived, and it will be kept active by a periodic exchange of messages, every 30 seconds or so. If network connectivity is lost or intermittent, the connection may be broken, and the "keep-alive" messages may not be received. In this case, the phone is said to have unregistered with the call server. If the network is unstable, with links coming up and going down frequently, the phones may have to constantly re-register with the call server. And anytime a phone is not registered with the call server, it cannot originate or receive a phone call. Figure 2-1 shows the call setup message flow between an IP phone and call server.

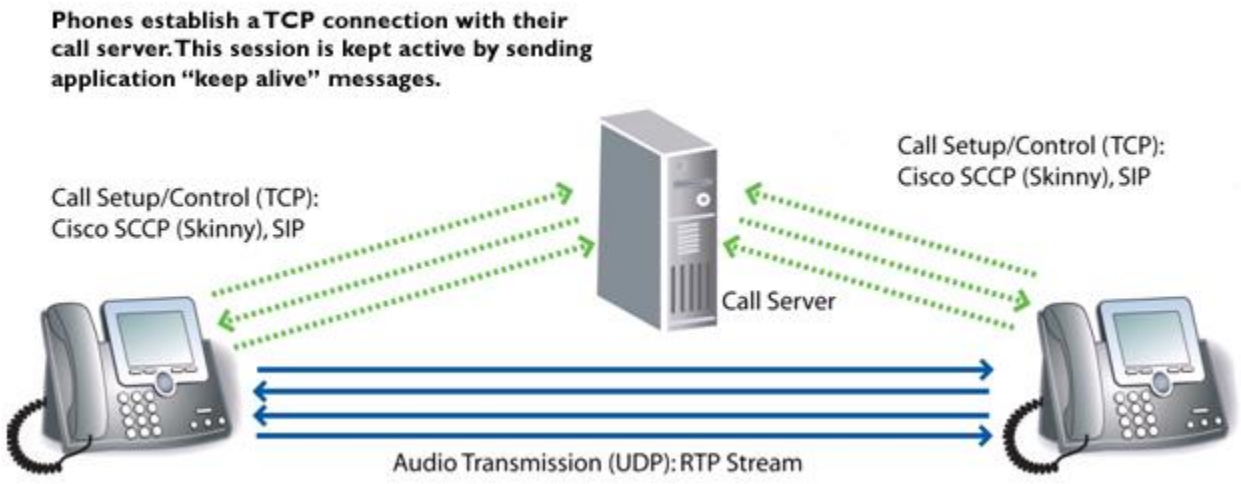


Figure 2-1 – Call setup messages in a Cisco system

When a user picks up the handset, the phone sends a message to the call server, which responds with a message that tells the phone to play the dial tone. If the call server is busy when the request is made, or if the network round-trip time is high, this message could be delayed. And if the delay is considerable, the user experience with the phone system is affected.

After getting the dial tone, the user's next step would be to dial the number. As the user dials the phone number, a series of setup message flows occur between the calling IP phone, the call server, and the called IP phone:

1. Calling phone sends dialed digits to call server.
2. Call server determines location of called phone.
3. Call server sends setup messages to called phone, telling it to ring.
4. Call server sends setup messages back to calling phone, telling it that the called phone is ringing.

At numerous points, call setup performance can be impacted by interruptions in these flows. First, if the call server is busy, it may take some extra time to look up the called phone information. If the round-trip time between calling phone and call server or between called phone and call server is high, then the overall delay will also be high. Figure 2-2 shows the high-level concepts that must occur for the call to go through.

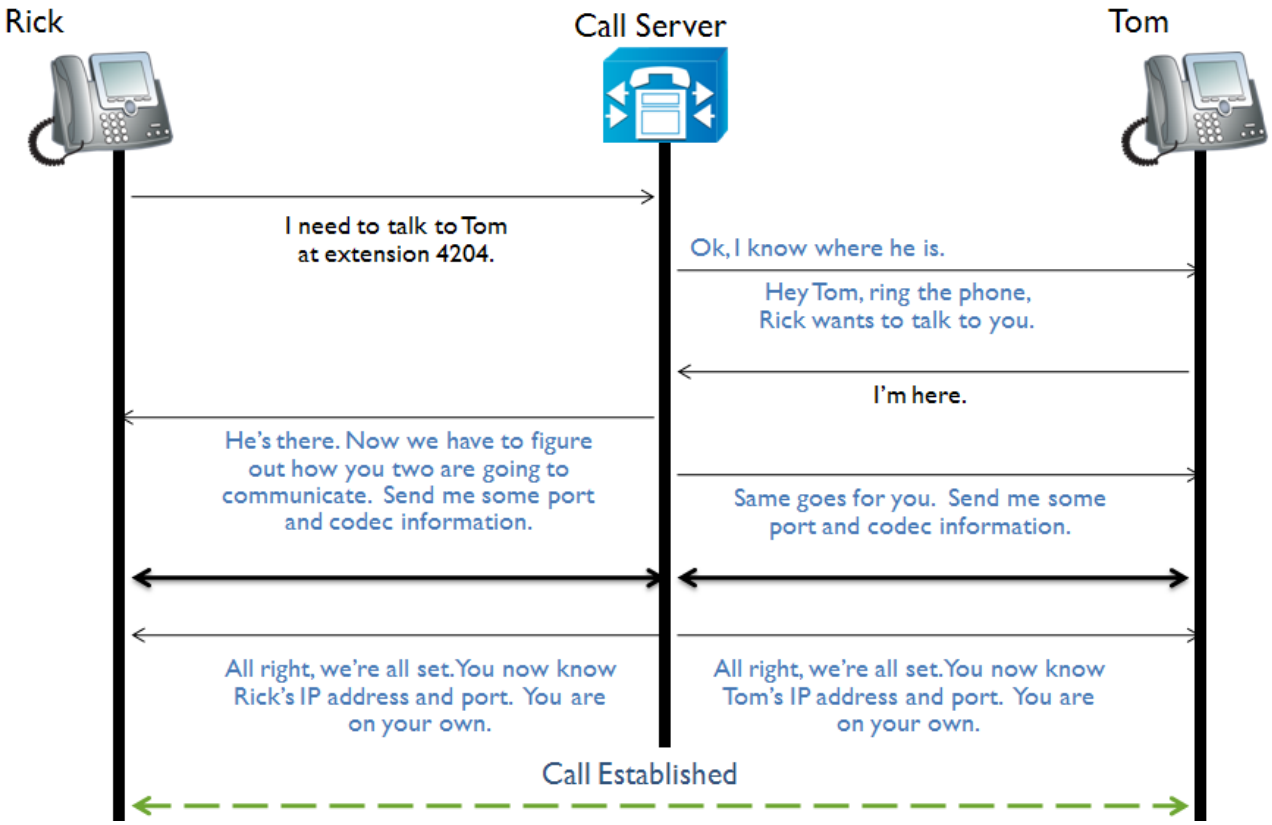


Figure 2-2 – Call setup exchanges that occur when a call is made.

With any intracluster call, it is important to understand the network path between the call server and the IP phones. If the phones and clustered call servers are on the same LAN, the potential areas of concern are substantially reduced. However, if the phones are in a branch office connected by a WAN to the data center where the call server cluster resides, call setup performance may be an issue.

In a Cisco environment, the call servers in a cluster can be geographically distributed, adding another layer of complexity to call setup procedures. Cisco network design requirements are rather strict for these scenarios: network round-trip time cannot exceed 40 ms, and at least 1.544 Mbps of bandwidth must be reserved for communications between the call servers.

In a distributed environment, you also need to be aware of which call servers your phones are registered with. In a failover case, phones at one site may fail over to a call server in a different geographical site. Call setup performance may be affected while the calls are using this backup server as call setup messages are routed over the network links to this other site.

### Intercluster Calls

An intercluster call takes place between two IP phones that are registered with different call server clusters. In this case, a connection between the clusters known as an intercluster trunk allows the call servers to communicate with each other using H.323 or SIP. In this scenario, a kind of "mini-call setup" occurs between the two clusters whenever an intercluster call is made. Figure 2-3 shows an example of an intercluster call.

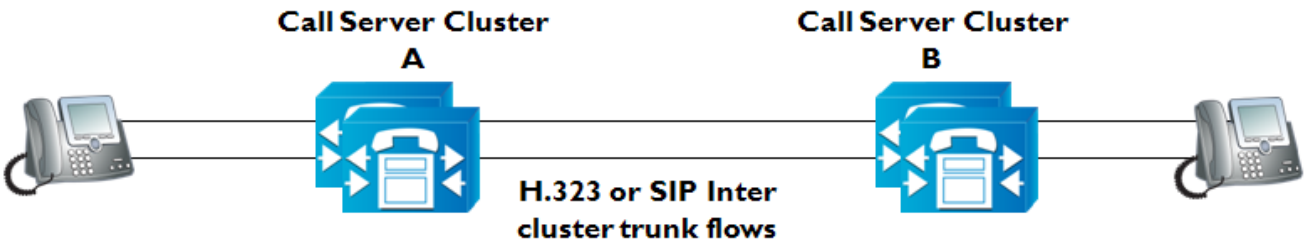


Figure 2-3 – A call setup process must occur between clusters to enable an intercluster call.

In the intercluster scenario, performance issues could occur between each IP phone and its call server, as well as between the two call server clusters. Depending on the composition of the network connecting the two clusters, bandwidth and QoS may be an issue.

We've discussed some examples of calls that typically stay on the IP network. Now let's look at the procedures used to set up calls that travel outside the IP network.

## OffNet Calls

An OffNet call is a call that travels from the IP network to the PSTN. This type of call would typically pass through a voice gateway and could originate either from the IP phone or from a phone in the PSTN. As soon as a voice gateway gets involved, call setup complexity and the potential for performance problems increase; you now have a situation where the call server communicates with the gateway, which in turn communicates with a separate network (the PSTN). Figure 2-4 shows an example gateway configuration for OffNet calls.

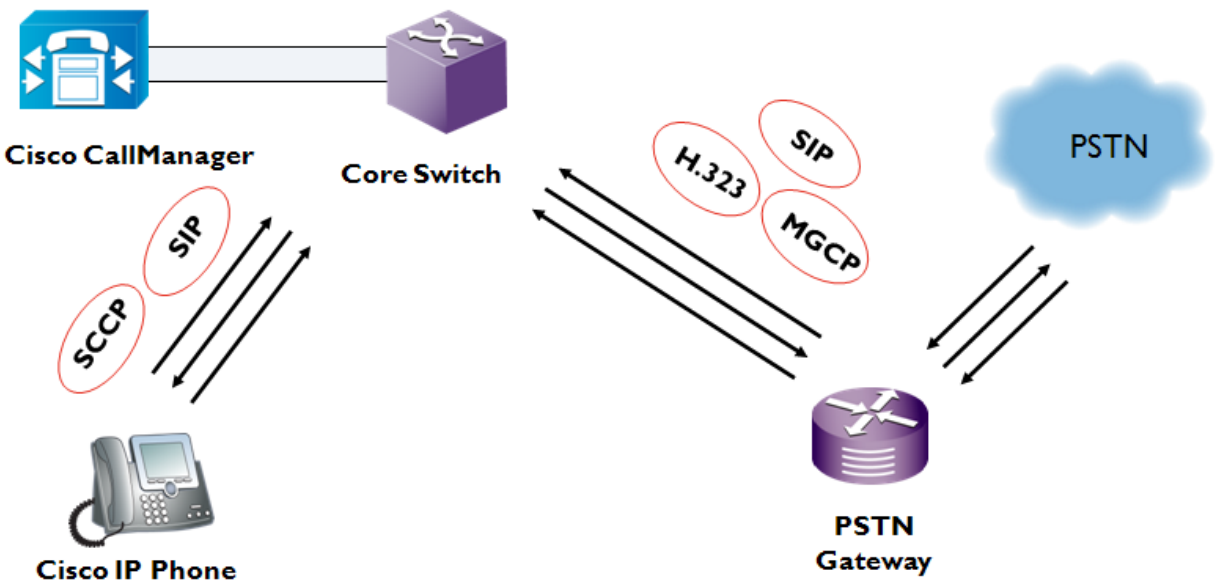


Figure 2-4 – OffNet calls involve a gateway to handle routing to the PSTN

OffNet calls could be local, long distance, or even international. To stay on top of call setup performance in a VoIP deployment, it's important to understand some of the call setup performance requirements for OffNet calls and the role that the voice gateway plays in enabling them.

## Gateways and Call Setup

To take a closer look at call setup for an OffNet call, let's discuss an example involving a typical PRI connection in a gateway using the MGCP protocol. With respect to call setup performance, what issues should you be concerned about? One of the first things to consider is that call setup through an MGCP PRI gateway will typically use both UDP and TCP flows. The MGCP protocol itself uses UDP with application-layer reliability built in. And a PRI connection uses a TCP channel that's established between the gateway and call server. In order to set up a call, the network must handle the UDP and TCP flows in a timely manner. Figure 2-5 shows a typical inbound call setup procedure using an MGCP PRI gateway.

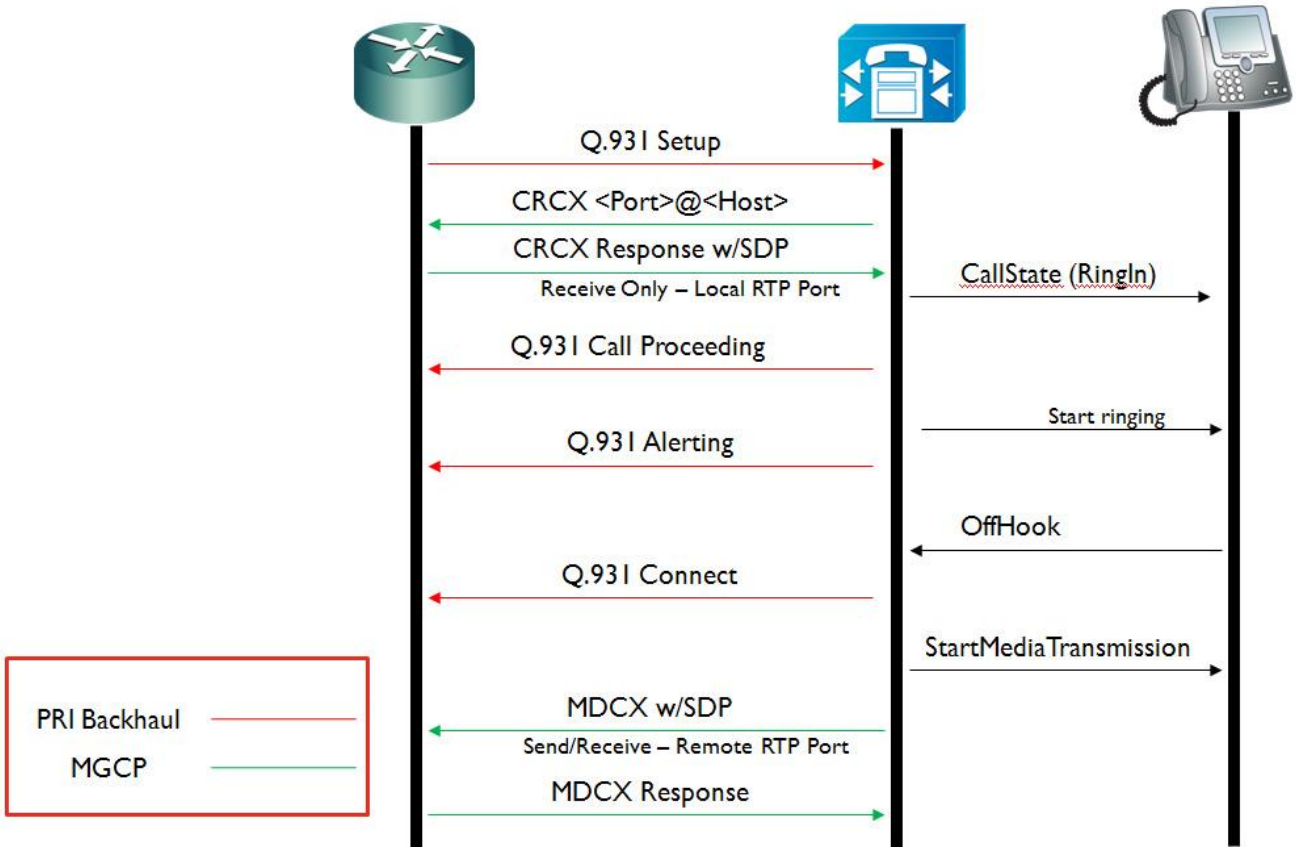


Figure 2-5 – Gateway example showing MGCP call setup flows

The network components between the gateway and call server need to handle the TCP and UDP call setup packets with low latency. These call setup packets should therefore use a QoS marking to ensure proper prioritization. The DSCP setting of "Class Selector 3" or CS3 is used to mark MGCP packets. QoS settings for call setup are typically different than those used for the VoIP call data. Different queues and classes allow for different levels of prioritization – and the VoIP data traffic is usually given priority over the call setup traffic.

Another common gateway call setup protocol, H.323, uses a lot of TCP flows to set up OffNet calls. This chatty protocol requires a number of back-and-forth flows among gateway, call server, and phone. In the previous chapter, we mentioned the impact that latency can have on protocols that require many flows. The latency adds up for each round trip, and it can easily affect the user experience with the VoIP phone system. H.323 actually consists of two protocols that handle the call setup tasks: H.225 and H.245. Figure 2-6 shows an example of H.323 call setup with the underlying H.225 and H.245 flows.

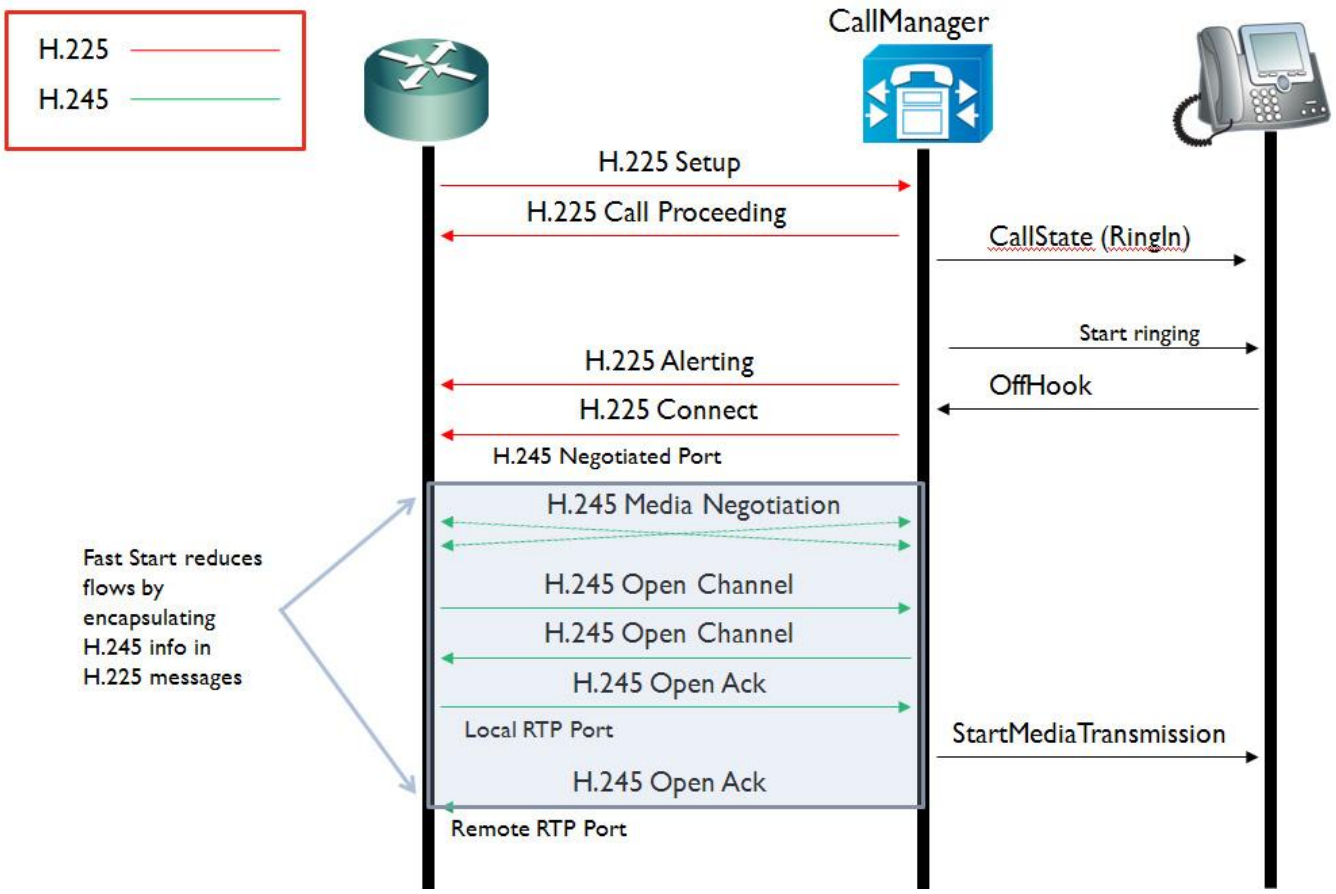


Figure 2-6 – Gateway call setup example using H.323 flows

The large number of H.323 flows required for call setup has resulted in new implementations that offer an option called "Fast Start." H.323 call setup using Fast Start provides a way to include the H.245 information within the H.225 flows, which reduces the total number of flows required to set up the call. Not all gateways support Fast Start, so be aware of this possible limitation when investigating issues related to H.323 performance.

The gateway has a lot of decisions to make when connecting the IP side of the network with the PSTN. For starters, it must pick the appropriate port or channel by which to send the call out to the PSTN. If a channel is not available, the gateway must provide a failure indication to the party that is trying to set up the call. All of this has to happen within a few seconds, or the user may assume that there is a problem.

Because the user experience is really what quality VoIP performance is all about, let's consider how you can measure VoIP performance from the perspective of call setup. Several key metrics are relevant for understanding the user experience with the call setup portion of a VoIP call.

## Key Call Setup Metrics

Just as with any network application, the underlying network performance will have a direct effect on user perceptions of call setup performance. Metrics like transaction time and network round-trip time are often used to measure TCP application performance and can be helpful when you need to tune network components. However, these metrics may be difficult to relate to the user experience with the phone system for several reasons. The network specific metrics are not application - or protocol - aware. Transaction time could be skewed by many keep-alive requests sent from the phone to its call server. Network round trip time is a good indicator of latency, but what if the call server processing speed is the problem?

A more VoIP-specific focus is required when tuning the network to carry high-quality phone calls. As a result, IP telephony experts instead rely on several metrics that relate directly to call setup performance to help them measure the likely user experience when interacting with the system. Let's discuss these key metrics individually.

### Delay to Dial Tone

The first user experience with a phone is likely the audible presence of a dial tone. To produce a dial tone, the IP phone sends a message to the call server letting it know that the phone is now off hook. And the call server sends a message back to the phone instructing it to play the dial tone. The time it takes for the off-hook message to be sent and the call-server response to be received is called the "delay to dial tone." Figure 2-7 shows an example of the delay to dial tone calculation for an IP phone running the SIP protocol.

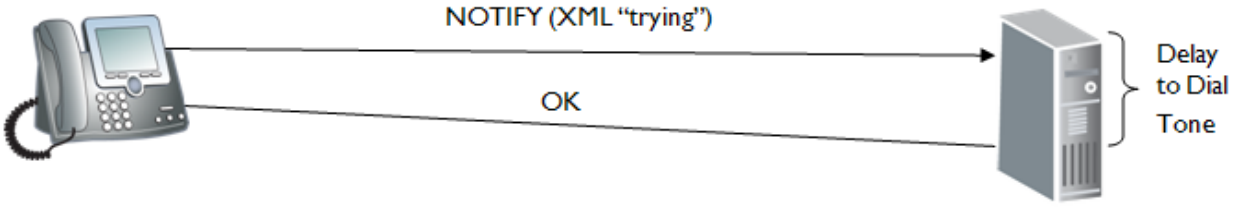


Figure 2-7 – The SIP phone sends a Notify and gets a response before playing a dial tone

The components involved in the delay to dial tone metric are the network round trip time (NRTT) and some amount of server processing time required to receive the off hook message and send the response. Network congestion and/or server congestion can both be culprits when delay to dial tone values are excessive. An overloaded call server can be slow to respond with the message telling the phone to play a dial tone. How much delay is too much when waiting for a dial tone? A reasonable time is 2 to 4 seconds. Any delay beyond 4 seconds may cause the user to think that the phone system is down.

The delay to dial tone metric usually only applies to outbound IP call legs. By contrast, an inbound, voice gateway call leg would not have a delay to dial tone metric because the dial tone for the PSTN phone is nearly instantaneous. And there are other cases where a delay to dial tone metric may not be applicable. For example, if you dial a number by pressing a Dial or Redial softkey, you may not get a dial tone; the call is dialed immediately.

## Post-Dial Delay

After the user dials a phone number, the next thing he or she expects is to hear a ringing or busy tone. The time between when the last digit of the phone number is dialed and when the user hears the call indication (ringing or busy) is called the post-dial delay metric. The PSTN has set the bar for post-dial delay quite high (or low, as the case may be). Calls on the PSTN usually connect in a very short period of time. In a VoIP network, this same level of performance can be achieved, with some careful tuning and management.

Some guidelines have been established for the post-dial delay metric and are accepted industry-wide. The ITU E.721 standard defines target values for different call types. Table 2-1 shows the target post-dial delay values.

| Call Type | Post-Dial Delay Target (Normal Load) |
|---|---|
| **Local connection** | 3 seconds |
| **Toll connection** | 5 seconds |
| **International connection** | 8 seconds |

Table 2-1 – Post-dial delay guidelines from ITU E.721

The post-dial delay metric depends on the protocol flows that occur after the user has entered the last digit of the phone number. Figure 2-8 shows an example of post-dial delay, calculated for the SIP protocol.

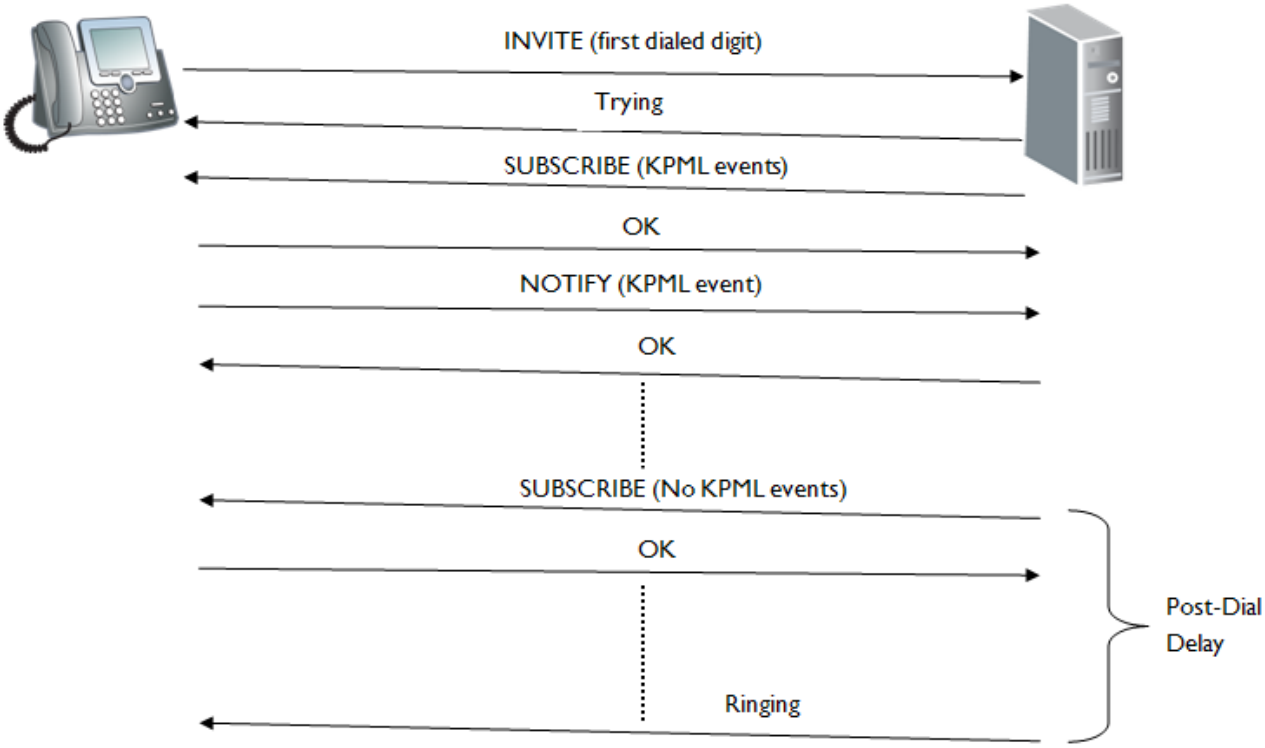


Figure 2-8 – Simple SIP example showing post-dial delay calculation

When examining the post dial delay metric, you should not factor in user time. For example, a caller may start dialing a number and then pause to look up the rest of the number. Post-dial delay calculation begins after the last digit in the number is pressed, or in some cases after the entire number is sent in a single flow.

Perhaps the first thing to look at more closely when investigating a performance issue with excessive post-dial delay is the configuration of the dial plan in the phone system. The dial plan is configured at the call server and/or gateway. It contains the information needed to route calls to their destination. The way the dial plan is structured can actually introduce an unnecessary delay in call processing. In a variable-length dial plan, the call server or gateway must wait to make sure that the user will not enter more digits.

For example, you might have a dial plan that allows numbers like 42xx or 42xxx. If a user dialed the number 4203, it could match either one of the two patterns: 42xx or 42xxx. Anytime a number is dialed, the call server or gateway must therefore wait to make sure that the user is not going to enter another digit. This delay is often called the *interdigit timeout* and it is usually a configurable parameter on the call server and gateway. The default value may be as high as 10-15 seconds. This means that a user could enter the last digit of a phone number, but the phone would not ring until 10-15 seconds later, when the interdigit timeout expired and the call server or gateway let the call proceed.

## Call Setup Failures

In addition to delay to dial tone and post-dial delay, the final aspect of call setup performance that requires VoIP-specific measurements is whether the calls are actually connecting successfully. When a call fails during the setup phase, often a "fast busy" tone is played by the phone. A call can fail to connect for many reasons. Cisco has a lengthy list of call failure cause codes in "Cisco Unified CallManager Call Detail Record Definitions". [2] Table 2-2 lists some of the more common call setup failure codes and their causes.

| Call Setup Failure Code | Description | Cause |
|---|---|---|
| **1** | Unassigned number | The number that was dialed does not exist or has not been assigned. |
| **3** | No route to destination | The number dialed is in the routing plan, but the called party cannot be reached through the network by which the call has been routed. |
| **27** | Destination out of order | The number dialed is valid, but there was a physical or data link layer failure at the remote party. |
| **34** | No circuit/channel available | There are no bearer channels available for this call. Could indicate a capacity problem with a need for additional channels. |
| **38** | Network out of order | The network is not functioning and may be down for an extended period of time. |
| **125** | Out of bandwidth | Call admission control has denied this call due to lack of bandwidth. Could indicate a capacity issue with a need for additional bandwidth at this location. |

Table 2-2 – Common call setup failure codes

A call setup failure in some cases may be expected. Access lines to the PSTN are a valuable, limited resource—a fixed cost for any organization. Because it is too cost-prohibitive and inefficient to provide a dedicated PSTN line for each employee, most enterprises operate on the assumption that not all users are on the phone at any given time. Depending on the calling patterns and volume at your enterprise, this is probably a good assumption. Traditional telecom professionals have a set of established metrics for tracking the percentage of time that a telephone user attempts to make a call and cannot get an outbound line (call setup failure code = 34). For a given phone system,

the probability that an attempted call will be blocked is called the Grade of Service (GoS). One way of calculating GoS is shown by the following equation:

**GoS = number of setup failures / number of call attempts**

A common metric used in telecom SLAs, GoS is expressed as a decimal fraction. A GoS of <= 0.01 is the typical benchmark. This value means that 1% or less of the call attempts failed.

Proactive monitoring of the key user experience metrics for call setup—delay to dial tone, post dial delay, and call setup failures—is an important part of any management plan. If you see degradation in any of these metrics, a closer look may be necessary to troubleshoot the problem.

## Troubleshooting Call Setup Performance

Identifying the problem is the first step in tackling a call setup issue. Looking in the right place is a good second step. To identify the problem, you first need to understand which call setup performance metric is impacted. There are a couple of ways to find out. One method is to simulate call setup protocol flows and measure the results. This is the general approach taken by the Cisco IPSLA function that is a part of almost all Cisco routing and switching devices. Another way to identify the target call setup metric is by continually monitoring the call setup metrics for your phones as users make real calls. Both approaches provide value to a proactive management solution. In either case, when you are monitoring call setup metrics, you need to be able to quickly see which users are affected and which metrics are impacted. A good troubleshooting process must therefore include a step-by-step approach to isolating the problem. Figure 2-9 outlines a troubleshooting process for call setup performance issues.

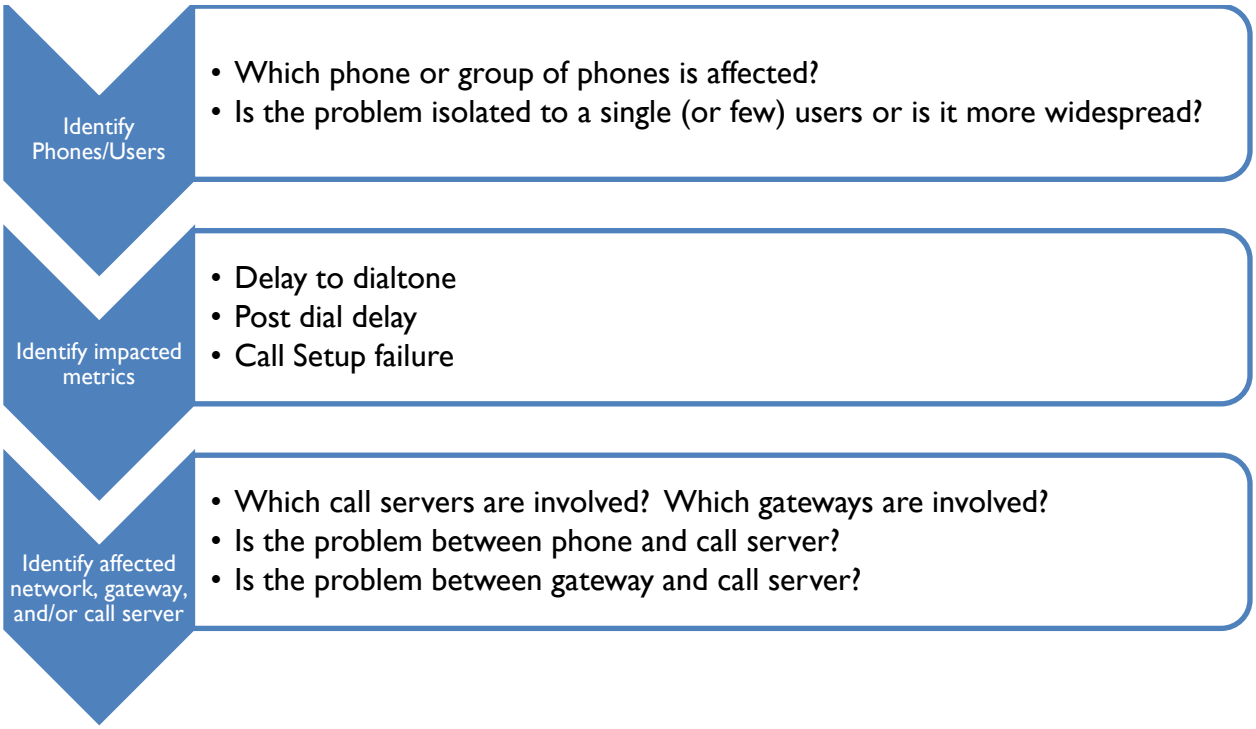


Figure 2-9 – Step-by-step call setup problem identification

Once you know the phone, the server or voice gateway, and the performance metric involved, you can take a look at the devices and network segments involved to determine whether a given problem is device – or network related. Knowing which metrics are impacted can help you look in the right place from the outset. Let's take a look at each metric and discuss some procedures for troubleshooting.

## Delay to Dial Tone

As we explained above, the delay to dial tone metric is associated with flows between the IP phone and its call server. With this in mind, we can immediately narrow our focus to either the network between the phone and call server, or to the call server itself. The following questions can help you narrow the focus of troubleshooting efforts when delay to dial tone values are higher than normal:

1) Which phone or group of phones is affected? Is the problem isolated to a single user, to a few users, or is it more widespread?
2) Which call server is handling call setup processing for the affected phone(s)?
3) What does the network path between phone and call server include?
4) Has the network path changed recently?
5) What is the associated latency for each hop in the path?
6) Do the latency statistics point to a single hop as the source of the problem?
7) Is the call server so heavily loaded that it can't respond to requests in a timely manner?

One rule of thumb that should serve as a starting point: With delay to dial tone performance issues, the network between phone and call server is the primary suspect.

## Post-Dial Delay

A problem with post-dial delay can be more complicated than a delay to dial tone issue. The post-dial delay metric involves more interactions between phone, call server, gateway, and PSTN than the delay to dial tone metric. In addition, the post-dial delay problem may be on another network, the network where the called phone resides. Answer the following questions to begin isolating the problem:

1) Which phone or group of phones is affected? Is the problem isolated to a single user, to a few users, or is it more widespread?
2) Which call server is handling call setup processing for the affected phone(s)?
3) There are at least two potential problem networks for the post-dial delay metric. What is the composition of the expected network path between phone and call server? What is the composition of the network path between gateway and call server?
4) Has either network path changed recently? Or could the call setup traffic be using an unexpected path?
5) What is the associated latency for each hop in the paths? Do the latency statistics point to a single hop as the source of the problem?
6) The call server or the voice gateway (or both) could be slow in responding due to resource utilization issues. Is the call server or gateway so heavily loaded that it can't respond to requests in a timely manner?
7) Is the dial plan configured correctly?

8) Are the affected calls using shared lines? Shared lines require setup flows to ring all phones that are "sharing" the line.
9) Does the call traverse an intercluster trunk? This type of call requires additional setup flows between call servers.

Post-dial delay issues involve more components and networks than delay to dial tone issues do. To address user complaints about post-dial delay, you have to look at the entire call setup flow between phone, call server, and possibly gateway to determine the likely cause.

## Call Setup Failures

A spike in failures of call setup could be linked to any number of components involved in processing the call. The reasons for a call setup failure could be as simple as the number dialed was not a valid phone number, or they could be more complex, as when a PSTN trunk is temporarily down. The following questions should put you on a path toward resolving the underlying issues:

1) What is the breakdown of the failure cause codes that are being returned? Does one particular cause code represent the majority of the call setup failures that are occurring?
2) Do the cause codes and their descriptions offer any clues?
3) What call servers and/or gateways are involved in the calls that are failing?
4) Are the failures always related to inbound calls? To outbound calls? Are the failed calls being made to a specific extension?
5) Are the failures related to calls into or out of a specific voice interface? What is the voice interface, and is the channel known?
6) Which phone or group of phones is affected? Is the problem isolated to a single user, to a few users, or is it more widespread?
7) Does the cause code point to a call server or gateway configuration problem?
8) Does the cause code point to a capacity issue? (For example, it may indicate that not enough channels are available for the call volume.)

Call setup failures can be limited to a very specific area of the system, or they can be quite widespread. The failure code itself is the key piece of information to unraveling these types of problems.

## Performance Incidents

Applying thresholds to the call setup metrics we've discussed in the previous sections is a good way to take a proactive approach to managing call setup performance. Multi-tier thresholds provide a way to alert the network manager when performance is deteriorating and before the problem becomes acute. By setting up good thresholds, you can be alerted to performance problems before getting complaints from irate users. But what are good thresholds? Are they industry-standard, or network-specific? Besides the GoS metric discussed earlier, is there an equivalent to a MOS for call setup?

Because we are dealing with the likely user experience when we attempt to manage the performance of a VoIP system, the subject of thresholds will necessarily be a little subjective. However, earlier in this chapter we discussed the typical numbers expected from the PSTN and the ITU guidelines for call setup metrics. From these sources, we can come up with some best practice thresholds for call setup metrics. Figure 2-10 lists default thresholds for consideration.

| Metric | Degraded Threshold | | Excessive Threshold | | Minimum Calls Originated |
|---|---|---|---|---|---|
| Delay to Dial Tone | Milliseconds | 2000 | Milliseconds | 4000 | 5 |
| Post Dial Delay | Milliseconds | 2000 | Milliseconds | 4000 | 5 |
| Call Setup Failures | Percentage | 2 | Percentage | 10 | 5 |

Figure 2-10 – Guidelines for call setup performance thresholds

For any type of call monitoring, look at a set of calls for a given time period - let's say 15 minutes. During this time period a number of calls are completed. Monitoring systems are notorious for generating tons of alerts, so you may want to filter some of the noise by choosing a reasonable value for the "Minimum Calls Originated," a minimum number of calls that must be placed, or at least attempted, by phones in the monitored system before any alerts can be raised. In the threshold guidelines above we selected 5 calls originated as the minimum threshold.

For a proactive monitoring solution, you'd like to know if the metrics for the calls were rated Normal, Degraded, or Excessive. A performance incident or alert should include information to help you quickly identify the location of the problem. Information such as the phone location (network subnet) and call server or gateway that handled the call setup should be included.

Network performance is variable, and the same factors that affect network latency and packet loss will likely affect VoIP call setup performance. As you configure thresholds and alerting for performance issues, consider the possibility that certain network locations are going to have worse call setup performance. For example, a group of Wi-Fi phones are likely to receive below-average call setup performance due to the higher latency in the wireless access network. For these phones, apply a set of threshold values that allow for routinely higher latency metrics at the specific wireless location. Otherwise, you'll keep seeing the same alerts each time someone uses those phones to place a call.

## Network Considerations

When addressing call setup performance problems, you certainly can't rule out general network issues. The network plays a key role in call setup performance, so bandwidth usage and QoS consistency are important parts of the network management equation that must be considered when troubleshooting VoIP call setup. Understanding these two items often requires network traffic analysis, with visibility into the composition and volume of network data flow. This type of analysis, best performed after maximum visibility into that data flow is achieved, is a critical component for a smooth and successful VoIP deployment and will help you far into the future as you plan for upgrades and the inevitable expansion of the VoIP system.

NetFlow is an example of a data source that provides information needed to effectively manage call setup performance. NetFlow is built into most Cisco network routers and switches. Statistics are kept about the protocol bandwidth usage and QoS markings for call setup (and all other) flows. You can then use a third-party monitoring package to collect, parse, and report on the NetFlow data.

### *Bandwidth Usage*

Once you have the required tools in place to help you analyze and understand the traffic that's flowing over your network links, you are better poised to avoid capacity and utilization issues that could potentially affect the VoIP system. We discussed the bandwidth usage associated with the most popular VoIP codecs in the previous chapter, but codecs handle VoIP conversation, not call setup, traffic. Do you have any idea how much bandwidth is used by call setup protocols? Do you understand the percentage of usage traceable to VoIP when compared to the mix of other protocols? These are good questions that can be answered by looking more closely at bandwidth consumption on your network. Figure 2-11 shows a breakdown of protocol bandwidth usage for a single router interface.

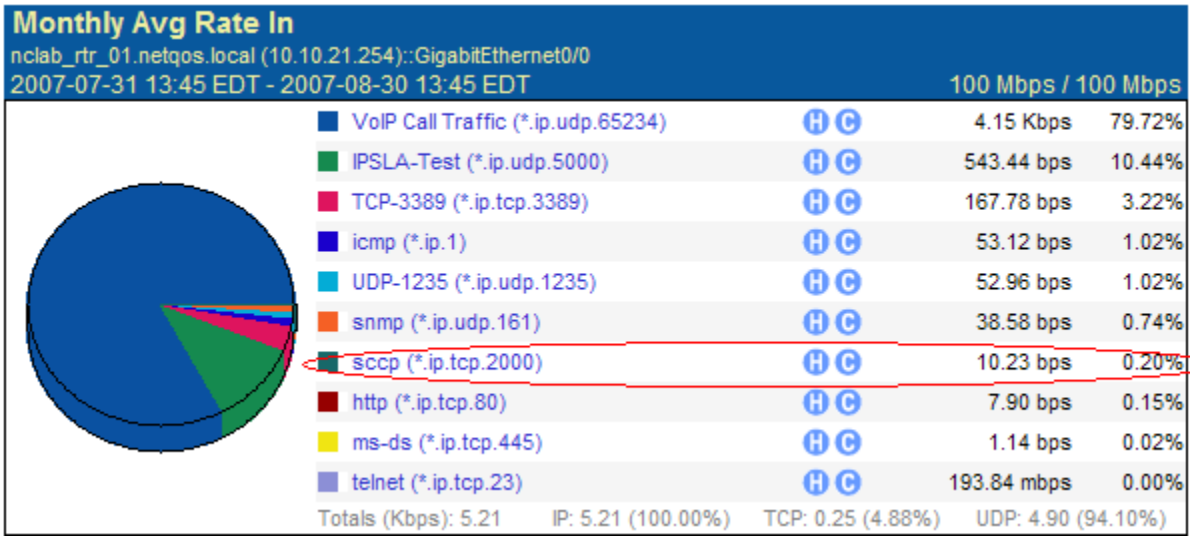


Figure 2-11 – Call setup protocol (SCCP) bandwidth usage on a particular interface

In most cases, bandwidth usage should not be an issue for call setup protocols. Relative to the amount of bandwidth required for call traffic, the call setup bandwidth usage is minimal per call. The call setup packets are typically very small (100-500 bytes). As we stated earlier, some IP phones send keep-alive messages, but these packets are small and only sent periodically. Overall bandwidth consumption per call for setup is relatively low.

However, in some environments, call setup might consume more bandwidth than you would expect. Consider the case of shared lines. A shared line is a phone number that is assigned to multiple physical phones. If the number is called, then all the phones that are "sharing" the line ring simultaneously. When a shared line is called, the call server must send setup flows to every phone in that shares the line. If you have several shared lines over a slower speed WAN link, this can lead to additional flows and more bandwidth consumption than you might expect.

Another scenario where bandwidth visibility is important is for calls over intercluster trunks. In this case there may be hundreds or thousands of call setups traversing the intercluster trunk link. Bandwidth usage could fluctuate greatly. Knowing how much bandwidth is used for these setups and being able to alert on rapid increases or degradations is important for call setup performance management.

In a Cisco environment, you can use the visibility provided by NetFlow to determine the mix of protocols and their bandwidth usage on particular links. You may find that call setup traffic is using more bandwidth or is going over an interface that you didn't expect.

### *QoS Mismatch*

QoS consistency for call setup packets is another item that can be determined via NetFlow analysis. IP phones and gateways will mark each packet with the desired QoS setting (e.g., CS3). As the packet traverses the network, routers along the way may alter the marking of the packets. This is particularly true of MPLS networks, where a packet entering a carrier network with one type of QoS marking may leave the network with a different QoS marking. QoS is only as good as its weakest link, and the configuration of any router can impact the prioritization that the packet receives along the network path. VoIP packets are relatively small, and call setup packets may be tiny, making them likely candidates for queuing behind larger application packets if they don't have the correct prioritization bit settings.

NetFlow information can be used to show the breakdown of QoS packet markings for packets passing through a router interface. The TOS byte in the IP header contains the QoS marking. This byte is also known as the DiffServ Code Point (DSCP). The bits within the DSCP byte represent different QoS levels. The value of the byte is usually included in common DiffServ naming conventions. For example, a value of 24 in the TOS byte field would be known as DSCP24. Using the TOS byte values from NetFlow, a breakdown of the different QoS values can be reported. Figure 2-12 illustrates this concept.

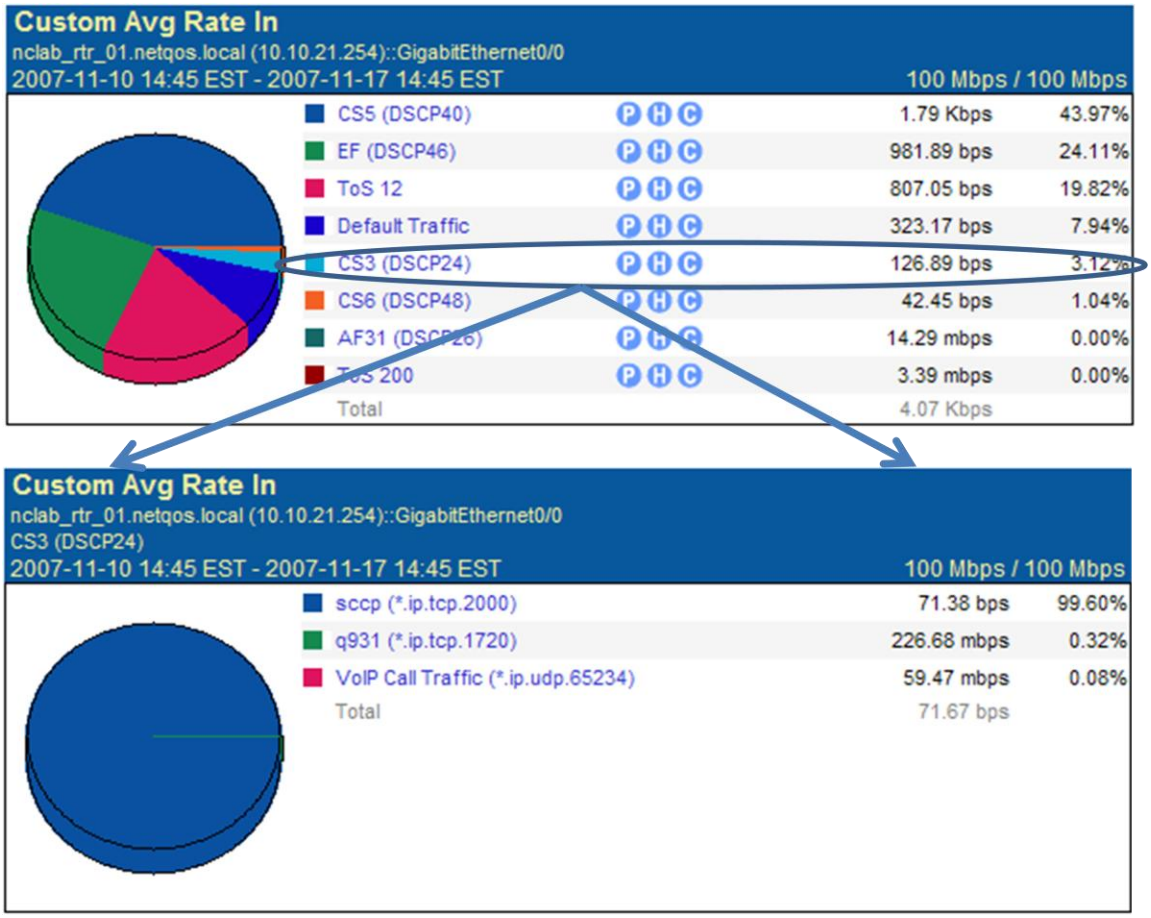


Figure 2-12 – Use QoS marking information to look for mismatches.

In addition to bandwidth allocation and QoS configuration, the manner in which you deploy phones, gateways, and call servers can also greatly impact the performance. Let's discuss some of the deployment models and considerations.

### *Network Deployment Considerations*

A number of different network deployment models are valid for a VoIP system and can deliver excellent performance. But whenever possible, it's best to plan carefully and select optimal placement for critical components. For example, the locations of the IP phones with respect to the call servers, and of the call servers with respect to the voice gateways, can easily have an impact on call setup performance.

A common enterprise deployment model is multiple sites with centralized call processing. Call servers are centralized and phones may be local or remote. For the case where phones are located in remote or branch offices, the call setup packets may have to traverse the WAN. Many enterprises are moving to MPLS networks which can provide QoS mechanisms for the VoIP traffic.

When planning the network aspect of a VoIP deployment, you can use the "further away principle" when considering phone, call server and gateway location. Translated to network terminology "further away" means more network hops. This principle states:

> "The further away a phone or voice gateway is from its call server, the more likely you are to encounter call setup performance issues."

As you add network hops between the phone and call server or call server and gateway, you inevitably introduce additional latency. Processing at each device along the way adds up and don't forget the propagation delay which is related directly to the physical distance that the packets must travel: they can't go faster than the speed of light. Some network scenarios that have the most potential for call setup performance issues:

- Phones or gateways located at the end of slow-speed links or links with high latency.
- A wireless phone accessing the network via high-latency wireless network.
- Phones or gateways with a WAN link (or links) connection to the call server.
- Phones that fail over or load balance to a different call server across a lengthy network path.
- Phones that must traverse a carrier network (that you don't own) to get to the call server.
- Geographically distributed call server clusters.
- Gateways located in different datacenter from where the call server is located.

Good network design principles are critical for ensuring optimal call setup performance. Include QoS early, or you will have to add it at a later date when it might not be as easy.

### *WAN Optimization*

Many enterprises are centralizing data center resources. As this occurs, a relatively new technique known as WAN optimization is being used to improve performance of critical customer applications and remote locations. WAN optimization is an umbrella term for many different techniques, including: TCP optimization, application protocol optimization, and data caching. The goal is to reduce network flows in an effort to decrease application latency and provide more available bandwidth.

Should you consider using WAN optimization techniques with VoIP in an attempt to improve call setup performance? Let's discuss the key points:

- WAN optimization would not help UDP based call setup protocols. Most WAN Optimization technology is focused on TCP applications. This would eliminate MGCP, and, in some cases SIP from consideration, but for other TCP based call setup protocols like SCCP, there are other issues.
- WAN optimization techniques do not work well with protocols that have small packet sizes. A quick test with a call using SCCP shows that 80% of the packets are less than 127 bytes and 97% of the packets are less than 255 bytes.

- WAN optimization data caching techniques would not help call setup protocols. Data caching is used for cases where large amounts of the same data are frequently requested by client applications on the WAN side. Call setup protocols do not send large amounts of data and frequent keep-alive messages are needed for the protocol to have awareness of the phone or call server connectivity.

It is likely that WAN optimization techniques would have a negative direct impact on call setup performance - if an attempt was made to optimize the call setup protocols. However, this doesn't mean that WAN optimization could not offer indirect help. By using WAN optimization techniques, other application traffic could be reduced. This would effectively provide more bandwidth and less contention for the VoIP call setup traffic – thus improving the overall performance for the call setup traffic.

## Chapter Summary

In this chapter, we've discussed the concepts that are important for ensuring optimal call setup performance. Why should you be concerned about call setup performance? The reason is that the user's first perception of the availability of a phone system is typically based on the call setup performance. In order to understand this aspect of user experience, you need some additional metrics beyond those used for traditional networked applications, metrics such as:

- Delay to dial tone
- Post-dial delay
- Call setup failures

Once you have the call setup performance optimized, the next area of focus should be on call quality performance. Good or bad call quality goes a long way toward shaping user perceptions of a VoIP system. Here are some questions to consider when it comes to call quality:

- What is the key quality of experience metric, and how is it measured?
- How do quality of experience metrics like MOS relate to the traditional network QoS for each call?
- What are the considerations for call quality when calls traverse the WAN?
- What are the considerations for call quality when calls go through a voice gateway to the PSTN?

In the next chapter of this ebook, we'll examine these questions and find out how they are related to VoIP call quality performance.

**References**


1. Eyers, Tony, and Henning Schulzrinne. Predicting Internet Telephony Call Setup Delay. 1999. http://www.cs.columbia.edu/~hgs/papers/Eyer0004_Predicting.pdf
2. Cisco Systems, Inc. Cisco Unified CallManager Call Detail Record Definitions (for Cisco Unified CallManager version 5.0). http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_programming_usage_guide09186a00806c22b5.html#wp144513