You might expect the number of trouble tickets and the time to close a ticket to decline as network technology matures. Unfortunately, for network support organizations, with every advancement in reliability and simplicity there is an offsetting technology advancement that makes networking more complex and prone to problems. Technologies such as unified communications, BYOD, high-speed Wi-Fi, cloud computing, and IPv6. Leading network support organizations are deploying new troubleshooting practices to reduce the number and duration of problems even while tasked with supporting the newest, most advanced technologies.

**TABLE OF CONTENTS**

- Are Problems Becoming a Thing of the Past?
- How It’s Done Today
- A Better Approach
- Improving the Process
- Checklist Testing
- Automating the Process
- Collaboration Best Practices
- Summary of Best Practices

*Troubleshooting*

1 of 4
Are Problems Becoming a Thing of the Past?
A recent research study of over 300 network professionals in large and medium sized organizations found that:

- Forty-eight percent of all organizations average longer than half a day to close trouble tickets
- Forty-six percent of all organizations are under pressure to reduce the time it takes to close trouble tickets
- Network professionals spend about 25% of their time solving problems

Why is this happening in light of all the IT advances designed to eliminate problems? One explanation is that for every advancement in reliability and simplicity, there is an offsetting technology advancement that makes things more complex: unified communications, 802.11n, cloud computing, or IPv6. Regardless of the reason, there is still much to be gained by improving problem-solving productivity.

How It's Done Today

How does the IT department deal with issues associated with troubleshooting? Approaches from the past, such as more staff and more training, are nonstarters in today’s era of tight budgets. Wholesale replacement or upgrades of the network are also a tough sell. Many IT departments might wish for better users, but that remains a dream.

A big part of the problem is in the ad-hoc approach most organizations take to troubleshooting. The vast majority (72 percent) of organizations do not follow a standardized process. Not only does the process vary within an organization, the tools these organizations use to troubleshoot problems vary substantially. Survey respondents report using up to eight different types of tools to solve a problem. In 47 percent of the situations, two or more tools were needed. With all the variability in troubleshooting practices and tools, it’s not surprising that 63 percent of troubleshooting sessions lasted more than an hour.

So the opportunity to reduce troubleshooting time can be found not in more people or more training, but in a better process for solving problems.

A Better Approach

Looking outside of the IT department provides ideas for a best practices in troubleshooting. Communication Service Provider technicians follow detailed troubleshooting processes. Medical providers follow protocols to examine and diagnose patients. Standardized checklists in the operating room reduce complications. If you think about it, most of the tasks people perform are much better organized than the ad-hoc approach to network troubleshooting.

Today’s Process

Let’s start by looking at today’s process as it is. When people think of troubleshooting, the first thought that comes to mind is the trial-and-error approach. The technician tries something, and sees if it solves the problem. This is repeated until the problem no longer exists. This step is entirely dependent on the skill and experience of the technician. In fact, it’s not really a process at all.

There’s a second step in troubleshooting, however. In many cases, technicians can’t resolve the problem themselves. Sometimes they need help with an especially difficult problem. In other instances, it’s because the problem lies outside their domain of responsibility, and they need to work with a separate group inside the enterprise (e.g.: server management or application developers) or outside (service providers or equipment vendors). This is far from a rarity – our research indicates that 41 percent of all issues require collaboration of this sort. This part of the process can take too long for at least two reasons. First, it’s not always easy to give the responsible parties visibility to the problem when it’s occurring. Second, the technician may not have the ability to easily capture the trace files that are often required (19 percent of the time) to resolve these problems.
A Study of Problem-Solving Techniques

This white paper refers to a research study conducted by NETSCOUT of 315 network professionals in April of 2012. The respondents came primarily from medium to large size networks in a variety of industries. Most of them were top level networking support staff.

The survey asked respondents to identify the root cause of their most recent user-reported problem. (Respondents could select more than one root cause.) The number one single cause was network problems (wired or wi-fi), occurring in 27 percent of instances. The combination of end user configuration and operation problems was the cause in 42 percent of cases.

Wired network problems – many things can go wrong in the network: cabling failures, hardware failures, and device misconfigurations. Many of these things can be tested in a relatively straightforward manner. One of the best methods is to start at the physical layer and then work up to the network layer:

- Cabling (opens, shorts, split pairs)
- Power over Ethernet (class, voltage, pairs used)
- Ethernet settings (signal level, speed, duplex settings)
- Switch configuration (Port and VLAN)
- DHCP (response time, values)
- DNS (response time, values)
- Gateway router (response time, availability)
- Overall network health (errors, discarded packets)

Performance tests to measure throughput, loss, latency and jitter can also be run to determine if the network is running slow for some reason.

Application Software and Server Problems – The most common complaint in this area is that “something is slow.” A quick way to check for that problem is to connect to the server or application in question and check the response time. For example, when loading a page from an HTTP server, we might want to know the:

- Lookup time
- Connect time
- Data start time
- Transfer time

From these, we can determine if there is a network or server issue. Further, if we test multiple servers and applications we can quickly compare those to see if the problem is isolated to a single server or is present on all of them (indicating a network problem). Many of these problems are more complex and will require collaboration, which we will...