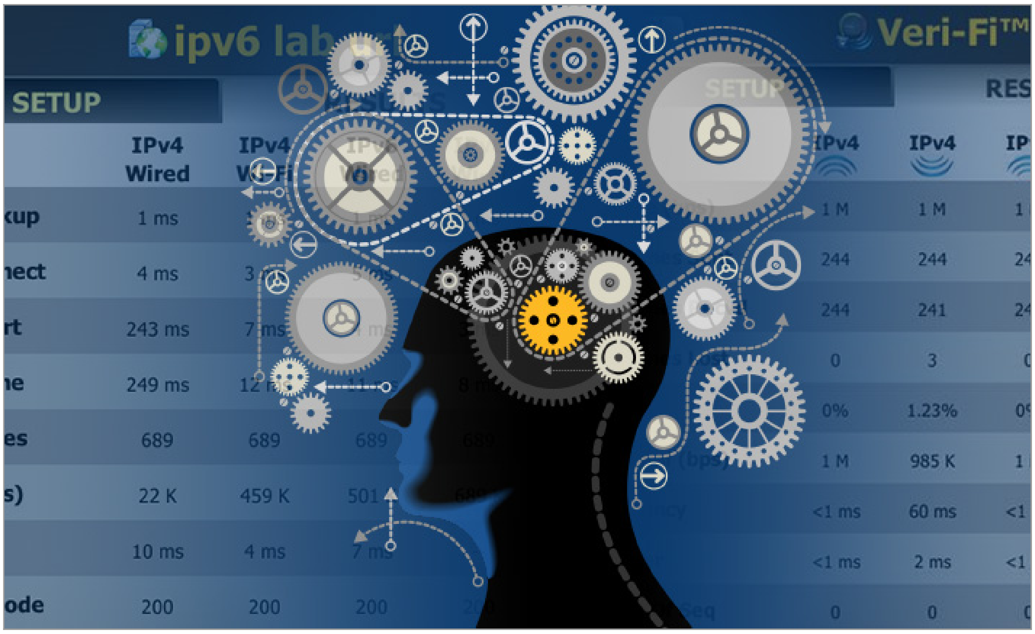
# White Paper: Implementing Best Practices Troubleshooting for IT

*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?

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 Fluke Networks 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.

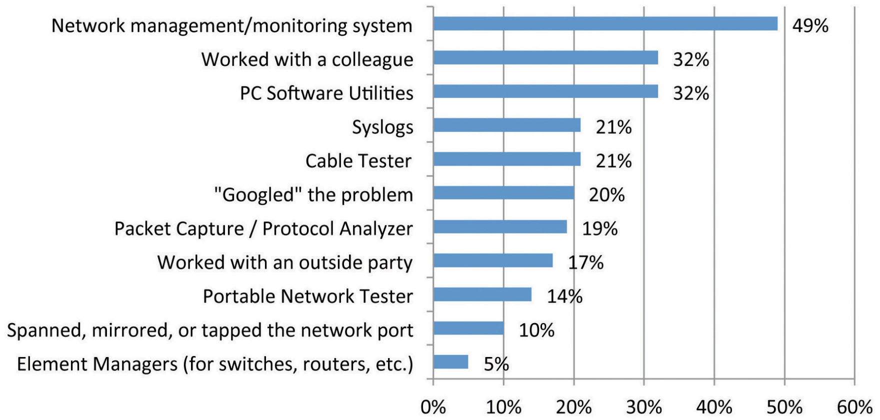


***Figure 1:** Which of these tools did you use to troubleshoot your most recent user problem?*

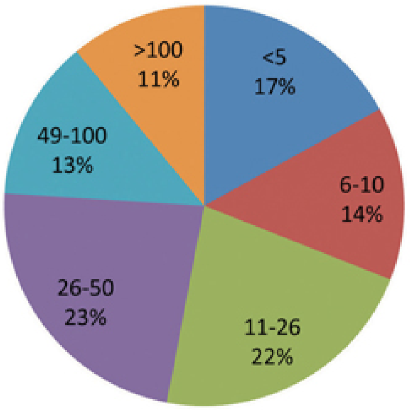


***Figure 2:** How many trouble tickets do you process in a typical month?*

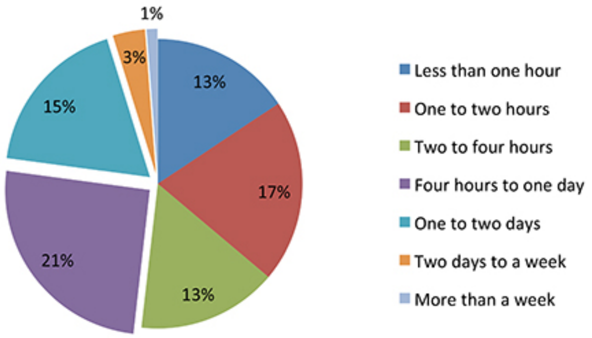


***Figure 3:** What is your group's average time to close a trouble ticket? Note that 48% report more than four hours.*

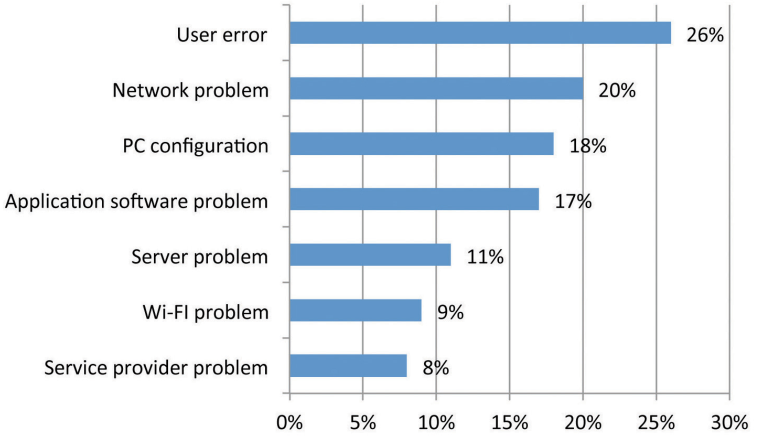


***Figure 4:** What was to root cause of the last user problem you solved? (Multiple responses allowed.)*

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.

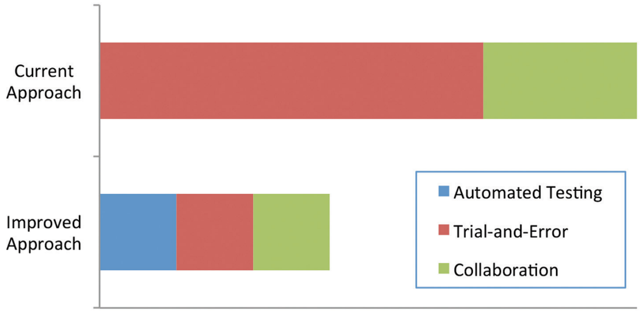


***Figure 5:** By replacing much of the trial-and-error testing with an automated test, as well as reducing the time to perform collaborative troubleshooting, a new approach can greatly reduce problem solving time.*

## Improving the Process

There's an old saying in medicine that "common problems occur commonly." This simple tautology can help provide a best practices approach for troubleshooting. By setting up a first step where common problems are quickly identified, a great deal of time can be saved. Creating a checklist of problems to check for first, in order of likelihood, would cut troubleshooting time. Later, we'll talk about how automating that process can result in even greater savings. So, a better process would actually have three steps:

1. Automated Testing of Common Problems
2. Manual Trial-and-Error Troubleshooting
3. Collaboration With Others

It may seem counterintuitive that adding a step will reduce time. But if the automated testing step can greatly reduce the time spent in manual trial-and-error troubleshooting, the total time can be reduced.

## Automated Testing

What are the common problems that can quickly be tested? Well, a good place to start would be with the root causes reported in Figure 4. Let's look through each of these and see what could be tested quickly.

**User errors / PC Configuration** – User errors can show up in many different ways. An incorrect wireless password can keep a user from connecting to the network at all. The wrong URL shortcut would prevent access to services. Changing network control panel settings can cause all sorts of issues.

One of the fastest ways to determine if the user is doing something wrong is to attempt to do the same thing with a known-good configured device. If the device can access the resources the user can't, then the user's PC is the culprit. If not, the problem is somewhere in the network or the device being accessed.

**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 discuss further below.

**Wi-Fi Problems** – Like testing a wired network, starting a layer 1 and working up to layer 3 is an effective way to test a Wi-Fi network

- Wi-Fi environment (signal strength, utilization, S/N ratio)
- Wi-Fi settings (SSID, security)
- DHCP (response time, values)
- DNS (response time, values)
- Gateway router (response time, availability)

A quick way to validate Wi-Fi performance is access a number of servers or services on both the wired and wi-fi side of the network and compare response times. This can quickly show if the issue is limited to a single server, or the Wi-Fi network, or if everything is slow. An alternative is to run performance testing to measure throughput, loss, and latency across the Wi-Fi network.

**Service Provider Problems** – These problems may be out of the realm of the technician to solve, but identifying the source of the problem will speed its resolution. One technique is to compare the performance of on-site and off-site (cloud) applications and see if there is a larger difference than expected. A more in-depth approach would be to measure the performance (throughput, loss, latency) of the service provider link in question.

## Automating the Process

Once a process is defined, it can be automated. There are several benefits to an automated test. First, it's much faster than performing the tests manually. Second, it's not subject to the human error of leaving out a test. Third, it allows anyone, regardless of skill level, to run these tests and identify these problems.

The savings from automation can be substantial. Tests using the Fluke Networks OneTouch™ AT show that about an hour's worth of standardized tests as described above can be done in a minute or less – a potentially huge savings over trial-and-error troubleshooting.

## Collaboration Best Practices

As noted earlier, network technicians regularly need to work with someone else to resolve problems. Many of the common problems noted above, such as server, application and service provider problems, almost always require that the tech work with others. The process of getting the right information in front of the right people, however, can take hours or even days. Even if the tech is able to work on other problems during this period, that's little comfort to an end user who can't get their job done or the IT manager missing targets for trouble ticket times. Here are some best practices to speed the collaborative process.

**Reports** – A detailed report of everything that the tech has tested and observed allows the tech to show a colleague exactly what was happening when they were observing the problem. A complete report from an automated test could include things that a less-experienced tech might not have thought to look at – but are there for a more knowledgeable team member to evaluate.

**In-Line Packet Capture** - Having a trace file is indispensable for very difficult problems or as evidence to an outside group such as application developers, service providers, or equipment suppliers. Collecting this information typically requires reconfiguration of the switch. This can take 30 minutes or more. Worse, many techs may not have the access or the knowledge to perform switch provisioning. That means even more delay as the problem is escalated to another individual.

To reduce capture time, each tech can be outfitted with an inline capture tool that will allow packet capture without having to access the switch, plus packet filtering and splicing to ensure that the pertinent packets are capture.

**Remote Interface** – The tech's test equipment should have software that allows remote access by other personnel in the organization (using a VNC client on a tablet or smartphone for example). Not only can the remote user see what the tech is seeing, but they can control the test equipment remotely and export trace files or reports from the tester to their local device. Remote access can be a source of security vulnerabilities and it must be installed and maintained properly.

**Summary of Best Practices Troubleshooting**

| Troubleshooting Phase | Issues | Tasks |
|---|---|---|
| Finding Common Problems | User Errors / PC Configuration | Quickly determine if the problem is in the PC or not, and what is not correct. |
| | Network Problems | Find common problems in network connectivity. |
| | Server / Application Problems | Test the response time of servers and applications. |
| | Wi-Fi Problems | Check the wi-fi environment, connectivity, and performance. |
| | Service Provider Issues | Test the response time of applications across the WAN or in the cloud; measure performance. |
| Automation | Automate the tests noted above | Can cut time required by 90% or more. |
| Collaboration | Reporting | Complete report of all results from the tests above remove the need for support teams to repeat tests. |
| | Remote Control / Sharing | Allow remote support staff see results and run tests from wherever they are while the problem is happening. |
| | Packet Capture | Provide a fast and simple way for the tech to capture problem traffic right where and when it occurs. |

**Table 1. Summary of a "best practices" approach for troubleshooting.**

Trial and error troubleshooting will still be needed for unique problems that can't be isolated with the techniques noted above. However, troubleshooting time can be greatly reduced by setting up the support team with defined process to isolate the majority of problems and providing a set of tools that can reduce collaboration time. These "best practice" approaches can reduce trouble ticket resolution time and free up staff to work on forward-looking projects.

Fluke Networks operates in more than 50 countries worldwide.
To find your local office contact details, go to www.flukenetworks.com/contact.