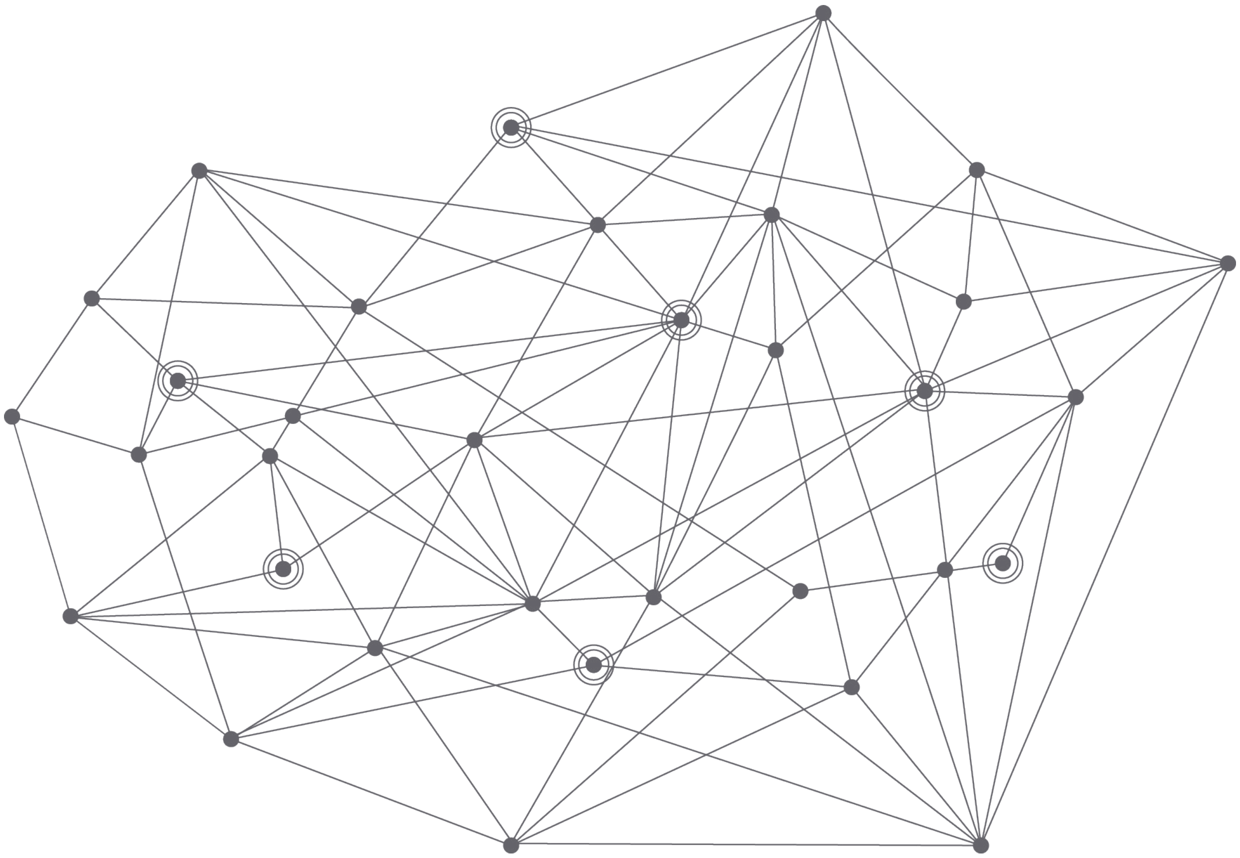


6 Steps for IT Troubleshooting at Remote Locations

WHITEPAPER

AppNeta

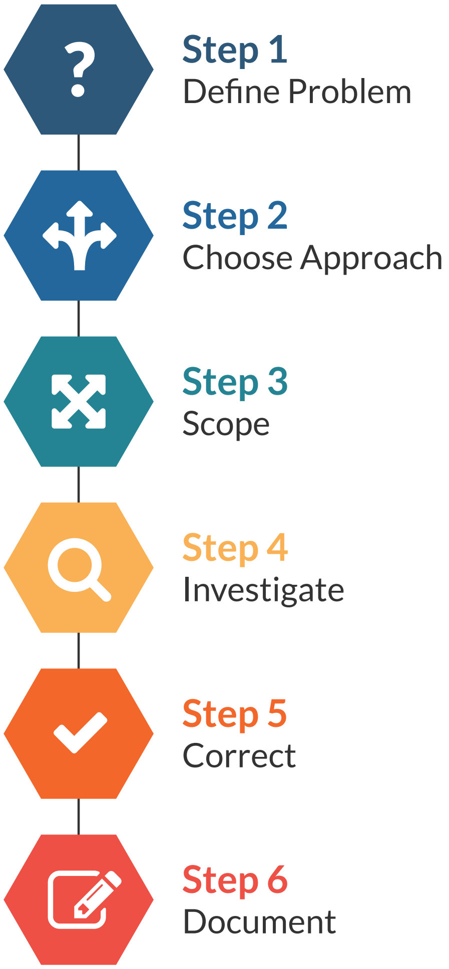
IT problems at a distance can be hard for IT to solve. Here are the steps to take to fix them quicker.

The essential trouble with troubleshooting remote locations is that they're remote. The term "remote location" can mean a lot of different things. Some may be typical branch offices away from company headquarters, while others may be banks, retail stores, university campuses, doctors' offices or hospitals.

Whatever the case, these locations are away from the main office or data center, which makes it hard for you in IT to easily diagnose any problem. Between you and any remote location, there are likely multiple networks served by multiple ISPs. The users at those locations may be accessing company resources over the VPN, SaaS applications and cloud-hosted workloads or files. These layers of technology add a lot of complexity to the IT troubleshooting process.

There's also another missing piece on-site: IT resources. Instead of being there or being able to call a coworker who's there, you have only imperfect visibility into those locations and have to rely heavily on various monitoring tools. You have to make some assumptions when trying to troubleshoot at remote locations. Ideally, you'll have remote access somehow, whether through SSH or VPN, or another method to test for issues with DNS/DHCP.

Today, most applications don't go down entirely—they just get slow. This, added to the complexity of the infrastructure serving users, means that IT teams serving remote location users may have unhappy users and recurring slowdowns without a clear way to eliminate the problem. Here are the steps to take to find and solve remote office problems.



Step 1
Define Problem

Step 2
Choose Approach

Step 3
Scope

Step 4
Investigate

Step 5
Correct

Step 6
Document

The Six Steps for Solving Remote Location Issues

1. Define the Problem

Lots of remote location technical issues are reported by users, unless your monitoring system alerts quickly on issues (and specifically on the right issues). When you hear about an incident, first:

Document the user description of the issue. With IT ticketing systems, this is often simple. See if you can add context around the issue or expand on it by starting a dialogue with the user. This will be especially helpful if you keep hearing about the same issue over and over.

Review the symptoms. See if the particular set of symptoms rings a bell from other helpdesk tickets or past issues. Is this a known problem that you or your team has experience with? Consult an internal knowledge base if it exists to get more data. Also, note the possible areas of effect of the symptoms. Note the connections between the user and the application or network they are having performance issues with.

One note here: Users at remote locations may experience “ghost” issues that can stymie IT’s efforts to identify them. You’ll know it’s a ghost issue when the user puts in a ticket or calls you—but then tells you the problem disappeared. Keep notes on these issues and when they occur. With continuous monitoring, you can pinpoint the cause.

2. Decide on an Approach

Once you know what the issue is, decide how you’ll proceed. It may be helpful to think of the OSI model here. Will you go for a top-down approach and start with the application? Or will you try a bottom-up approach that starts at the physical layer?

As you do this step, document your assumptions and why you chose this method. It may seem obvious to you, but think about the eventual use of this information in a knowledge base for a new employee. Your assumption may indicate a complex architectural choice that was made before their time.

3. Figure Out the Scope of the Problem

It’d be nice if incidents were small-scale, but they aren’t always. See if you can figure out the number of users or systems affected as soon as possible. The main reason to scope the issue is to reduce the mean time to repair (MTTR) by reducing the problem footprint. However, it must be done carefully, as incorrectly scoping the problem could lead to investigating the wrong link in the delivery chain. Try these steps to get the details you’ll need.

- Identify who is affected: Is it all users, all apps and all locations?
- Diagnose when the issue occurred or if it is ongoing.
- Assuming it’s an application issue, can you determine where in the application delivery path the issue exists, whether LAN, WAN or application location?
    - See whether there are external factors that could be impacting performance.
    - Events of local, national or international importance can often distract employees all at once. Consider sporting events or breaking news that might indicate excess video streaming or activity.
    - This is the only time we’ll suggest looking for widespread internet outages. Sites like Reddit or HackerNews are good at quickly identifying these and providing more context, like links to status pages or articles.

4. Undertake Your Investigation

Here's where you start the detective work. Write down your assumptions of what could be wrong and a short list of priorities to eliminate the most likely causes first.

For a top-down approach: Consult synthetics monitoring to the application to view coarse network, server and browser timings from your (ideally multiple) locations. While outside the purview of most IT teams, total latency timings can be broken down to the steps of a synthetic script to isolate whether the issue is with a certain page, data retrieval, etc. This information can then be used to help the SaaS app provider identify the problem.

For a bottom-up approach: First, identify the application delivery path on the network from the user to the app where the problem exists. Look at the LAN, WAN and vendor regions of the delivery path to determine if the issue is in your infrastructure, the wider internet or with your SaaS provider. You may also consider using multi-protocol route determination to mimic application (TCP) or voice (UDP) traffic to look for routing differences.

5. Fix That Problem

Fixing the issue is, of course, highly variable to the type of problem and the steps to mediation. Whatever the issue, it is most important to note differences in the process as compared to your expectations. Note the time that it takes to perform each step. Tracking the details of issues and the time it takes to fix them will become important if the same issue continues to pop up.

Set Yourself Up for Success

In a fast-paced world where IT is under pressure to provide good end-user experience, it's helpful to take charge. IT has lost control in a lot of ways, but they can reclaim it. Here, a few tips on moving to being more proactive when remote location problems arise.

Consider an in-house SLA. If you don't already have an SLA to other departments or users, create one. When problems do occur, you can point to the SLA and its objectives to give a realistic picture of when users can expect fixes.

Track helpdesk tickets. Make sure you collect data on helpdesk tickets—how many are received and solved in a given period of time. Having solid data will help back you up if any business leaders are offering criticism or decisions based on hunches or guesses.

Know application owners. With SaaS and cloud-based apps now dominating IT, there are many cooks in the application kitchen. But you can know who owns which applications throughout your organization. This will help identify which apps are actually being used, and know where to go when troubleshooting and eliminating possible sources.

What if you can't fix the problem? This is a very real possibility, especially with those ghost issues that come and go. Some issues may not be crippling to user productivity, but are recurring annoyances that may signal a bigger problem. It's very hard to pin down ghost issues if your monitoring tool is a legacy product that depends only on devices in the infrastructure. That will be particularly unhelpful when remote locations come into play, since those tools can't monitor applications delivered over the WAN. We're pretty sure continuous monitoring is the way to go to identify the ghost issues. You'll be able to see historical data to start eliminating possibilities to get to the cause of problems.

6. Do Mitigation and Documentation Follow-up

After identifying and fixing the problem, look into ways to mitigate this issue in the future. If you missed key information that your monitoring couldn’t detect, look into complementary or replacement solutions that can give you that missing visibility. Make sure to communicate the issue or log it in a report if it was widespread enough.

Doing this kind of post-mortem and documenting your findings will allow you to expand your knowledge base and save important infrastructure information for your team. Looking into the lessons learned will allow you to discern where you can speed up your process.