12.7.1 Troubleshooting Methodology

Click one of the buttons to take you to that part of the video.

Troubleshooting Methodology 0:00-1:36

Being a good troubleshooter is a key part of being an effective Linux system administrator. I've been teaching new system administrators for about two decades now and this is one of the hardest skills for some students to master. Some new admins just seem to have an intrinsic sense for how to troubleshoot problems, but others don't. The reason for this, in my opinion, is that troubleshooting is actually part art and part skill.

Just as it's difficult for some of us--me--included to learn how to draw, sculpt, paint, or, heaven forbid, dance--what a disaster. It's also difficult for some of us to learn how to troubleshoot problems. However, I have noticed as well that with a little training and a lot of practice, most new system administrators can eventually learn how to be very effective and efficient troubleshooters. There are three keys to doing this.

Number one, you need to use a solid troubleshooting procedure. Number two, you need to have a working knowledge of the various troubleshooting tools at your disposal. And then, finally, you need to gain a lot of experience actually troubleshooting real problems.

Now, the last point is beyond the scope of this lesson. The only way you get troubleshooting experience is to spend several years working in the field.

Network problems can be caused by a wide array of issues, and we can't even begin to cover all of them here. Instead, what I want to do is focus on using a standardized troubleshooting process when you're working with network issues. By using a standardized process, you can adapt to confront and resolve a broad range of different types of network problems.

Review Standardized Troubleshooting Model 1:35-2:48

The network model we're going to look at here is by no means all inclusive. You're going to have to add, remove, or reorganize the various steps to match each particular situation. However, this model should give you a fairly good base to start from.

Before we begin looking at this model, I need to point out that many new system administrators make a catastrophic mistake when they troubleshoot system or network problems. Instead of using a methodical troubleshooting approach, they go off halfcocked and start trying to implement fixes before they really know what the problem is.

I call it shotgun troubleshooting. The administrator tries one fix after another hoping that one of them will just magically repair the problem. This is a very dangerous practice. I've watched system administrators do this. And usually what happens is they cause more problems than they solve. Sometimes they have even caused catastrophic problems.

Instead of using shotgun troubleshooting, you should use a standardized troubleshooting model instead. The goal of this model is to concretely identify the source of the problem before you start trying to fix things. I know that sounds like a simple concept, but really a lot of system administrators struggle with this.

Here's a suggested model that you can use to develop your own personal troubleshooting methodology.

Gather information 2:49-3:07

The first step is to gather information. This is absolutely critical. You need to determine exactly what has happened. What are the symptoms? What were the error messages that were displayed? What did they say? How extensive is the problem? Is it isolated to just one system, or are there many systems on the network experiencing exactly the same problem?

Identify what has changed 3:08-3:21

Step two is to identify what has changed. In this step, you identify what has changed in the system. Has new software been installed? Has new hardware been installed? Did a user change something? Did you change something?

Create a hypothesis 3:22-4:07

Step three is to use this information to create a hypothesis. Using all this information that you gathered in step one and two, you might actually be able to develop several hypotheses that could explain the problem.

Now, more than likely, you're going to have to do some research in this step. You need to be able to go on the internet and then look at various sources of information--such as frequently asked questions, knowledge bases, forums, and so on--to find answers to problems.

In addition, you should also consult with your peers to kind of validate your hypothesis. Based on their experience, they might be able to give you some good feedback as to what the likely problem is.

Then, using all this information, you need to narrow your results down to maybe one or two possible, most likely causes.

Determine the appropriate fix 4:08-4:48

Step four is to determine what the appropriate fix is. Now, we're not repairing anything yet. We're just deciding what the problem is and how we're going to fix it. We haven't actually started installing software, changing settings, or anything yet.

To determine an appropriate fix, you need to consult various sources on the internet, talk to your peers, and maybe even draw upon your own experience to identify what steps you're going to have to go through in order to fix the problem.

As you do these you need to be sure to identify the possible ramifications of implementing the fix and decide how you're going to account for those ramifications. Be aware that many times the fix may have side effects that are as bad or worse than the original problem.

Implement the fix 4:49-5:21

Step five is to implement the fix. At this point, you're ready to actually start making changes. Notice in this troubleshooting model that we did a ton of research before we started implementing fixes. This is key.

Doing this greatly increases the likelihood of having a successful troubleshooting experience. Then after you're done implementing the fix, please take the time and go through the effort necessary to verify that the fix really did repair the problem and that that issue doesn't reappear again.

Ensure user satisfaction 5:22-6:07

Step six is one that we are very bad at as system administrators. We need to ensure user satisfaction. This is a key mistake that is made by many, many, many system administrators. We system administrators are notoriously poor communicators.

If a problem affects users, you need to make sure you communicate the nature of the problem to them. Make sure they are aware that it has actually been fixed. Don't leave them hanging. If applicable, you should also educate them as to how to keep the problem from occurring in the future--especially if something they did is causing the problem.

In addition, you should also communicate with the user's supervisors to ensure that they know that the problem has been fixed as well.

And then step seven, absolutely critical. This is another thing that system administrators are really bad at.

Document the solution 6:07-6:50

That is to document the solution to the problem, because I can just about guarantee that the problem is going to happen again. It may be a month down the road. It could be a year or two down the road.

The problem is, if you don't document it, and that problem reoccurs, you're going to say things like, "Oh yeah. That happened. How did we fix that? I can't remember."

If you document it, you can go look through your documentation and say, "Oh, I remember this happened last year, and oh, that's how we fixed it." You don't have to go through as much of the troubleshooting process because you've already done that.

You've already identified how the problem was fixed earlier. Now, if you follow this methodology, you too can learn to be a very effective troubleshooter, especially as you gain a lot of hands-on experience in the real world.

Summary 6:51-6:56

That's it for this lesson. In this lesson, we emphasized the importance of using a standardized troubleshooting model when troubleshooting system problems.
