Realtime
publishers

The Definitive Guide™ To

Active Directory Troubleshooting, Auditing, and Best Practices

2011 Edition

Don Jones

Chapter 3: Active Directory Troubleshooting: Tools and Practices 32
Narrowing Down the Problem Domain 32
Sean's Seven Principles for Better Troubleshooting 33
A Flowchart for AD Troubleshooting 34
Easy Stuff: Network Issues 35
Name Resolution Issues 36
Log Spelunking 37
AD Service Issues 37
Client-Domain Controller Issues 39
Replication Issues 40
AD Database Issues 42
Group Policy Issues 43
Kerberos Issues 45
Coming Up Next 46
Download Additional eBooks from Realtime Nexus! 46

Copyright Statement

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers or its web site sponsors. In no event shall Realtime Publishers or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com.

Chapter 3: Active Directory Troubleshooting: Tools and Practices

For the most part, in most organizations, Active Directory (AD) "just works." Over the past 10 years or so, Microsoft has improved both AD's performance and its stability, to the point where few organizations with a well-designed AD infrastructure experience day-to-day issues. That said, when things *do* go wrong, it can be pretty scary because a lot of us don't have day-to-day experience in troubleshooting AD. The goal of this chapter is to provide a structured approach to troubleshooting to help you put out those fires faster.

For this chapter, I'll be drawing a lot on the wisdom and experience of Sean Deuby, a fellow Microsoft Most Valuable Professional award recipient and a real AD troubleshooting guru. You might enjoy reading his infrequently-updated blog at http://www.windowsitpro.com/blogs/ActiveDirectoryTroubleshootingTipsandTricks.aspx. Although he doesn't post a lot, what he does post is worth the trip.

Narrowing Down the Problem Domain

"How do you find a wolf in Siberia?" It's a question I and others have used to kick off any discussion on troubleshooting. Siberia is, of course, a huge place, and finding a particular *anything*—let alone a wolf—is tough. The answer to the riddle is a maxim for troubleshooting:

Build a wolf-proof fence down the center, and then look on one side of the fence.

Troubleshooting consists mainly of tests, designed to see if a particular root cause is responsible for your problems. The answer to the riddle provides important guidance: Make sure your tests (that is, the wolf-proof fence) can definitively eliminate one or more root causes (that is, one whole half of Siberia). Don't bother conducting tests that can't eliminate a root cause. For example, if a user can't log in, you might first check their physical network connection. Doing so definitively eliminates a potential problem (network connectivity) so that you can move on to other possible root causes. Of course, checking connectivity only eliminates one or two possible root causes; a *better* first test would eliminate a whole host of them. For example, checking to see whether a different user could log in might eliminate the vast majority of potential infrastructure problems, making that a better wolf-proof fence.

Sean's Seven Principles for Better Troubleshooting

Here's where I'll repeat excellent advice Sean Deuby once offered. Follow these seven principles (which I'll explain through the filter of my own experience) and you'll be a faster, better troubleshooter in any circumstance.

1. Be Logical. Pay attention to how you're attempting to solve the problem. Before you do anything, ask yourself, "What outcome do I expect from this? If I get that outcome, what does it mean? If I don't get the expected outcome, what does *that* mean?" Don't do anything unless you know why, and unless you can state what the follow-up step would be.
2. Remember Occam's Razor. Simply put, the simplest solution is often the correct one. Don't start rebooting domain controllers until you've checked that the user is trying the correct password.
3. What Changed? If everything was working fine an hour ago, what's different? This is where change auditing tools can come in handy. Although I don't specifically recommend it, I've used Quest's ChangeAuditor for Active Directory in the past because it keeps a very detailed, real-time log of changes, and it's been a big help in solving some tricky issues. Whatever changed recently is a very likely candidate for being the root cause of your current woes.
4. Don't Make Assumptions. It's easy to make assumptions, but sticking with an orderly elimination of possible causes will get you to the root cause of the problem more consistently. For example, don't assume that just because one user can log on that everything's okay with the infrastructure; the problem user might be hitting a different domain controller, for example.
5. Change One Thing at a Time, and Retest. You won't get anywhere with five people attacking the problem, each one changing things as they go. You also won't get anywhere if *you're* changing multiple things at once. If the boss is tearing his hair out to get things fixed, remind him that you have just as much capability to *further break things* if you're not methodical.
6. Trust, but Verify, Evidence. Sometimes an inaccurate problem description can get you going in the wrong direction—so verify everything (this goes back to not making assumptions, too). "I can't log in!" a user cries over the phone. "Log into *what?*" you should ask, before diving into AD problems. Maybe the user is talking about their Gmail account.
7. Document Everything You Try. Especially for tough issues, documenting everything you try will help keep you from repeating steps, and will help you eliminate possible causes more easily. It's also crucial in the inevitable post-mortem, where you and your colleagues will discuss how to keep this from happening again, or how to solve it more quickly the next time.

A Flowchart for AD Troubleshooting

Sean has further helped by coming up with an AD troubleshooting flowchart, which I'll reprint in pieces throughout this chapter. You should check Sean's blog or Web site (which is shown at the bottom of the chart pages) for the latest revision of the flowchart. Sean's blog also offers a full-sized PDF version, which I keep right near my desk at all times. The flowchart starts with that is shown in Figure 3.1, which is the core starting point that gets you off to the different sections of the chart.

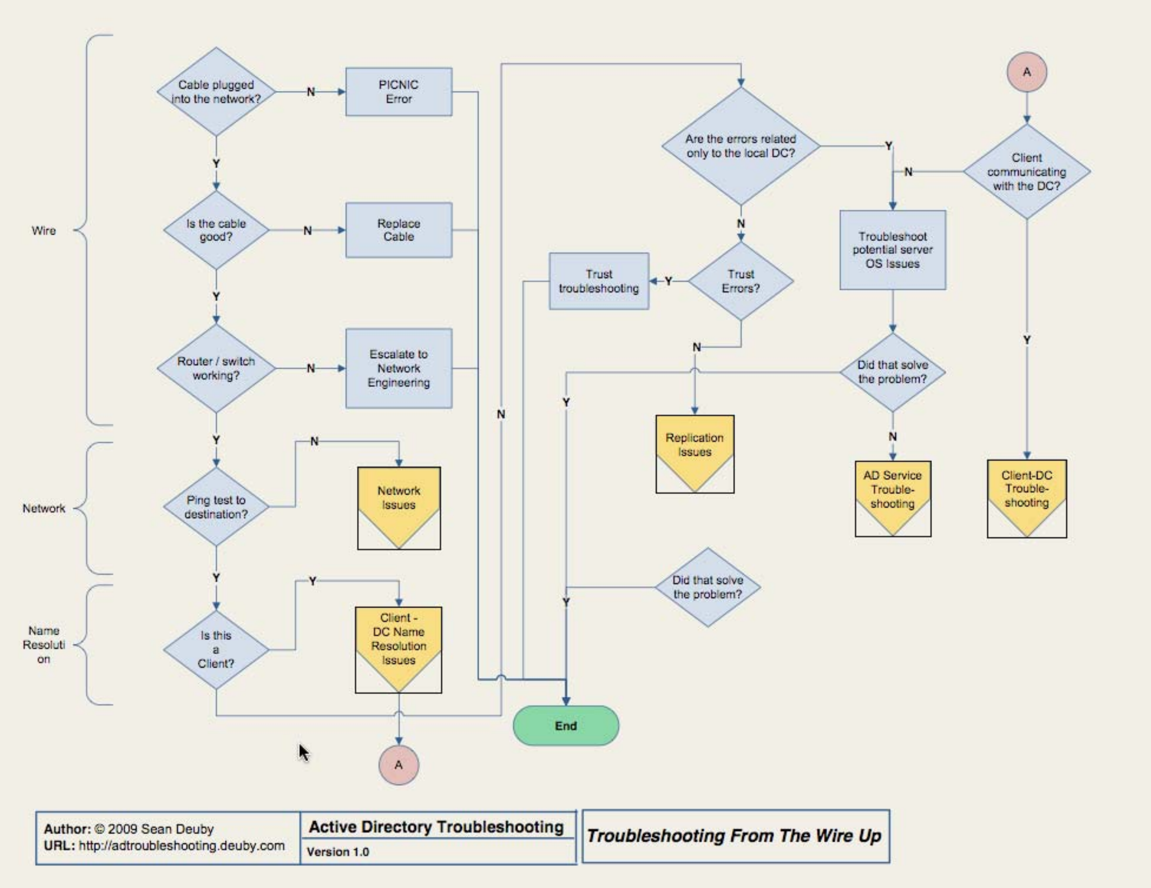
Wire
Network
Name Resoluti on
Cable plugged into the network?
N
PICNIC Error
Y
Is the cable good?
N
Replace Cable
Y
Router / switch working?
N
Escalate to Network Engineering
Y
N
Ping test to destination?
Network Issues
Y
Y
Is this a Client?
Client - DC Name Resolution Issues
A
End
Are the errors related only to the local DC?
Y
N
Trust troubleshooting
Y
Trust Errors?
N
Replication Issues
Y
Y
Did that solve the problem?
A
Client communicating with the DC?
N
Troubleshoot potential server OS Issues
Did that solve the problem?
N
AD Service Trouble-shooting
Y
Client-DC Trouble-shooting
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
Troubleshooting From The Wire Up

Figure 3.1: Starting point in AD troubleshooting.

Note

I strongly recommend that you head over to Sean's blog or Web site to download the PDF version of this flowchart for yourself. You may find a later version, which is great—it'll still start off in basically this same way.

Start in the upper-left, with "Cable plugged into network?" and work down from there. The basics—the "wire" portion—should be things you can quickly eliminate, but don't eliminate them without actually testing them. You might, for example, attempt to ping a known-good IP address on the network (using an IP address prevents potential DNS issues from becoming involved at this point). If that doesn't work, you've got a hardware issue of some kind to solve.

Easy Stuff: Network Issues

A ping does, of course, start to encroach on the "Network" section of the flowchart. Stick with IP addresses to this point because we're not ready to involve DNS yet. If the ping isn't successful, and you've verified the network adapter, cabling, router, and other infrastructure hardware, you're ready to move on to Figure 3.2, which is the Network Issues portion of the flowchart.

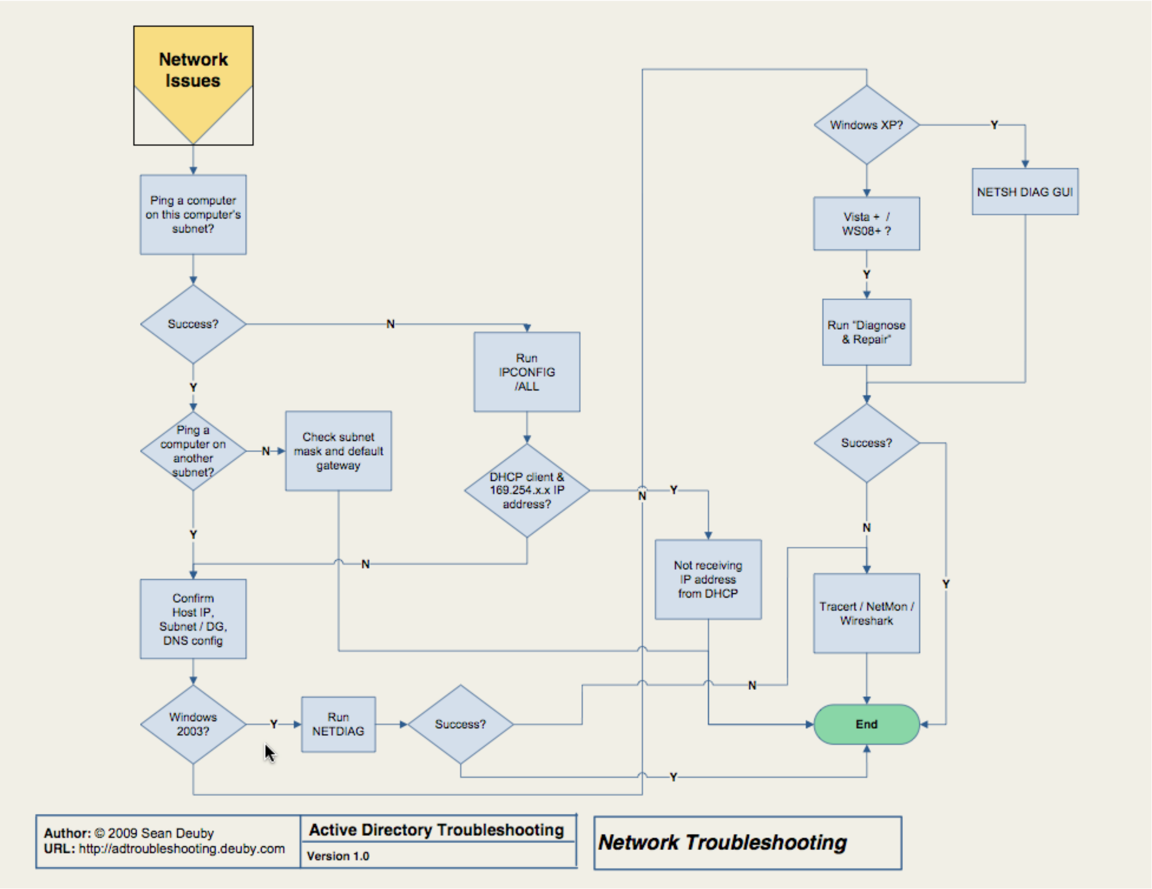
Network Issues
Ping a computer on this computer's subnet?
Success?
N
Y
Run IPCONFIG /ALL
Ping a computer on another subnet?
N
Check subnet mask and default gateway
DHCP client & 169.254.x.x IP address?
N
Y
Y
N
Confirm Host IP, Subnet / DG, DNS config
Windows 2003?
Y
Run NETDIAG
Success?
Windows XP?
Y
NETSH DIAG GUI
Vista + / WS08+ ?
Y
Run "Diagnose & Repair"
Success?
N
Y
Not receiving IP address from DHCP
Tracert / NetMon / Wireshark
N
End
Y
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
Network Troubleshooting

Figure 3.2: Network issues.

The tools here are straightforward, so I won't dwell on them. You'll be using ping, Ipconfig, Netdiag, and other built-in tools. At worst, you might find yourself hauling out Wireshark or Network Monitor to actually check network packets. That's not truly AD troubleshooting, so it's out of scope for this book, but the flowchart should walk you through to a solution if this is your root cause.

Name Resolution Issues

If a ping to a different intranet subnet worked by IP address, it's time to start pinging by computer name to test name resolution. Watch the ping command's output to see if it resolves a server's name to the correct IP address. Ideally, use the name of a domain controller or two because we're testing AD problems. If ping doesn't resolve correctly, or can't resolve at all, you're ready to move into the name resolution issues.

The "Client-DC Name Resolution Issues" flowchart is designed for when you're troubleshooting connectivity from a client to a domain controller; if you're troubleshooting problems on a server, you'll skip this step and move on in the core flowchart (Figure 3.1). If you *are* on a client, the flowchart that Figure 3.3 shows will come into play.

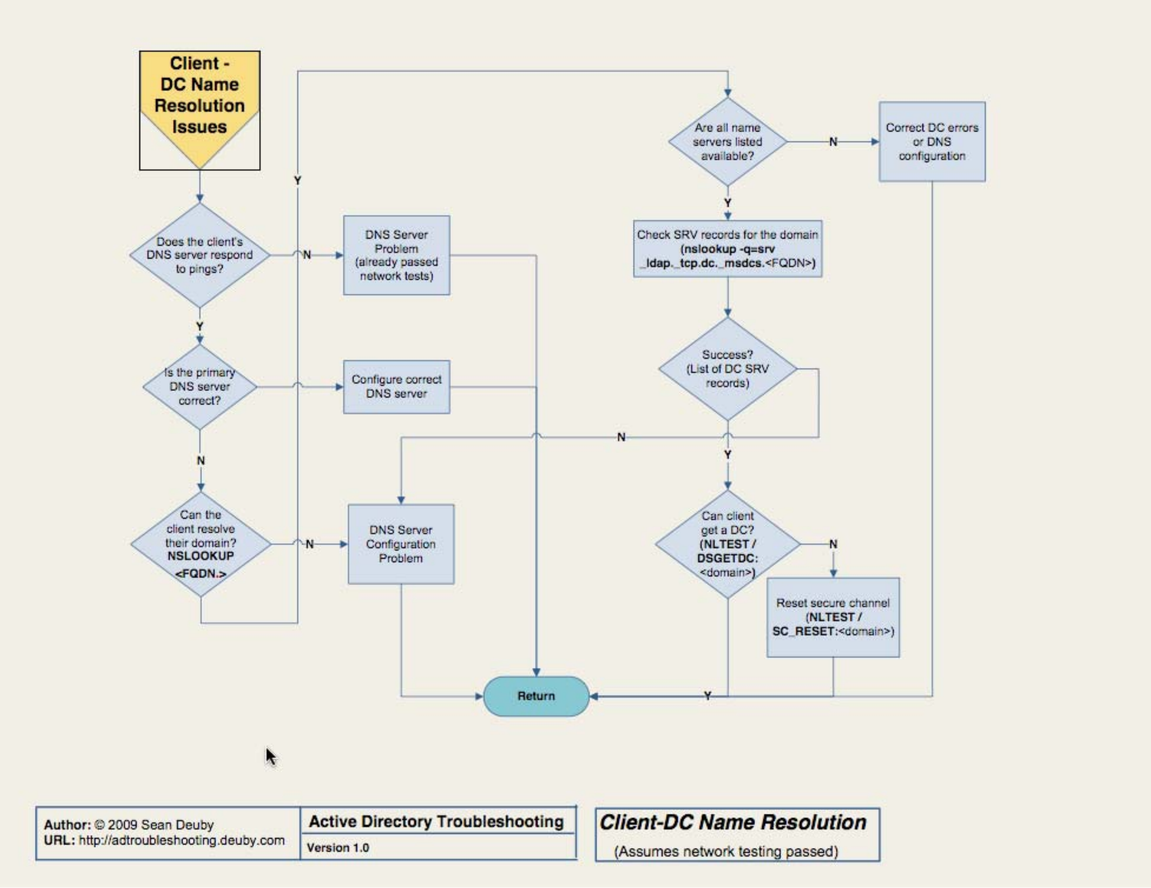
Client - DC Name Resolution Issues
Does the client's DNS server respond to pings?
N
DNS Server Problem (already passed network tests)
Y
Is the primary DNS server correct?
Configure correct DNS server
N
Can the client resolve their domain? NSLOOKUP <FQDN.>
N
DNS Server Configuration Problem
Y
Are all name servers listed available?
N
Correct DC errors or DNS configuration
Y
Check SRV records for the domain (nslookup -q=srv _ldap._tcp.dc._msdcs.<FQDN>)
Success? (List of DC SRV records)
N
Y
Can client get a DC? (NLTEST / DSGETDC: <domain>)
N
Reset secure channel (NLTEST / SC_RESET:<domain>)
Y
Return
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
Client-DC Name Resolution
(Assumes network testing passed)

Figure 3.3 Client-DC name resolution issues.

Again, the tools for troubleshooting name resolution should be familiar to you. Primarily, you'll rely on ping and Nslookup. Of these, Nslookup might be the one you use the least—but if you're going to be troubleshooting AD, it's worth your while to get comfortable with it. The flowchart offers the exact commands you need to use, provided you know the Fully-Qualified Distinguished Name (FQDN) of your domain (for example, *dc=Microsoft,dc=com* for the *Microsoft.com* domain).

The other tool you'll find yourself using is Nltest, which permits you to test the client's ability to connect to a domain controller, among other things.

Resource

A complete description of Nltest can be found at http://support.microsoft.com/kb/158148.

Log Spelunking

Once name resolution is resolved, or if it isn't the problem, you have a bit of checking to do before you move on. Specifically, you're going to have to look in the System and Application event logs on the domain controllers in the client's local site (or whatever domain controller you're having a problem with, if it's just a specific one). If you find any errors, you'll have to resolve them—and they may be more specific to Windows than to AD. Don't ignore anything. In fact, that "don't ignore anything" is a huge reason I *hate* domain controllers that do anything other than run AD, and perhaps DNS and DHCP. I once had a domain controller that was having real issues talking to the network. There were a bunch of IIS-related errors in the log, but I ignored those—what does IIS have to do with networking or AD, after all? I shouldn't have made assumptions: It turned out that IIS was more or less jamming up the network pipe. Shutting it down solved the problem for AD.

Log Exploring

Having to dig through the event logs on more than one domain controller—heck, even doing it on one server—is time-consuming and frustrating. This is where some kind of log consolidation and analysis tool can help tremendously. Get all your logs into *one* place, and have software that can pre-filter the event entries to just those that need your attention. Software like Microsoft System Center Operations Manager can also help because one of its jobs is to scan event logs and call to your attention any events that require it.

If you don't see any errors specific to the domain controller or controllers, you move on. You're looking first for errors related to trusts, and if you find any, you'll need to resolve them. If you did find errors related to the domain controller or controllers, and you corrected them but that didn't solve the problem, you're moving on to AD service issues.

AD Service Issues

Figure 3.4 contains the AD service issue portion of the troubleshooting flowchart. Here, we've moved into the complex part of AD troubleshooting. First, of course, look in the event log for errors or warnings. Don't ignore something just because you don't understand it; you're going to have to amass knowledge about obscure AD events so that you know which ones can be safely ignored in a given situation.

This is where *knowledge,* more than pure data, comes in handy. Operations Manager, for example, can be extended with Management Packs that *should* be called Knowledge Packs. When important events pop up in the log, Ops Manager can not only alert you to them but also explain what they mean and what you can do to resolve them. NetPro made a product called DirectoryTroubleshooter that went even further, incorporating a complete knowledge base of what those events meant and how to deal with them. Sadly, the product was discontinued when the company was purchased by Quest, but Quest does offer a similar product: Spotlight on Active Directory. Again, its job is to call your attention to problematic events and provide guidance on how to resolve them.

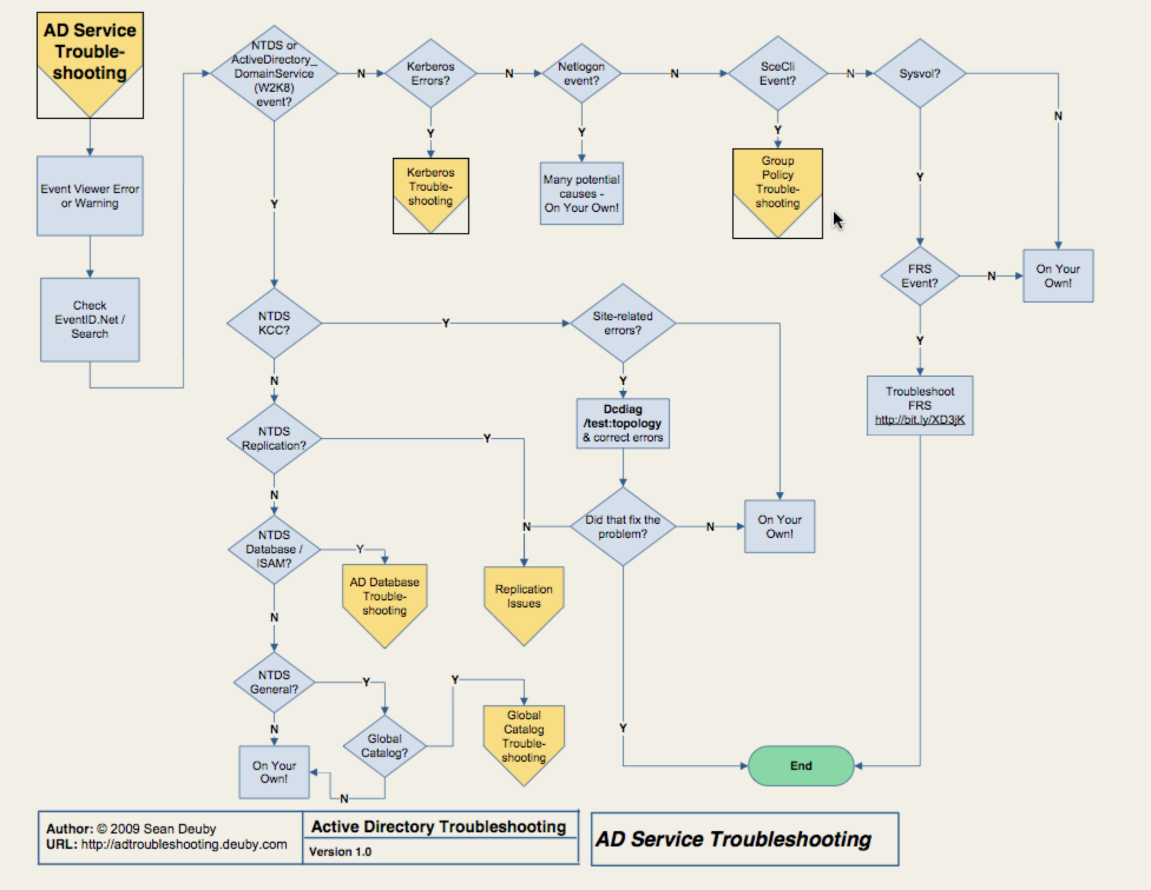
AD Service Trouble-shooting
Event Viewer Error or Warning
Check EventID.Net / Search
NTDS or ActiveDirectory_ DomainService (W2K8) event?
N
Kerberos Errors?
Y
Kerberos Trouble-shooting
N
Netlogon event?
Y
Many potential causes - On Your Own!
N
SceCli Event?
Y
Group Policy Trouble-shooting
N
Sysvol?
N
Y
FRS Event?
N
On Your Own!
Y
Troubleshoot FRS http://bit.ly/XD3jK
Y
NTDS KCC?
Y
Site-related errors?
Y
Dcdiag /test:topology & correct errors
N
NTDS Replication?
Y
Did that fix the problem?
N
On Your Own!
N
Replication Issues
Y
N
NTDS Database / ISAM?
Y
AD Database Trouble-shooting
N
NTDS General?
Y
Global Catalog?
Y
Global Catalog Trouble-shooting
N
On Your Own!
N
End
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
AD Service Troubleshooting

Figure 3.4: AD service troubleshooting.

The remainder of the AD service troubleshooting flowchart helps you narrow down the potential specific AD service involved in the problem based on the error messages you find in the log. You might be looking at Kerberos, the AD database, Global Catalog (GC), Replication, or Group Policy. Along the way, you'll also troubleshoot site-related issues and the File Replication System (FRS). We'll pick up most of these major service issues in dedicated sections later in this chapter.

Client-Domain Controller Issues

Assuming you resolved any client name resolution issues earlier, if you're still having problems with the client communicating with the domain controller, you'll move to the Client-DC Troubleshooting chart, which Figure 3.5 shows.

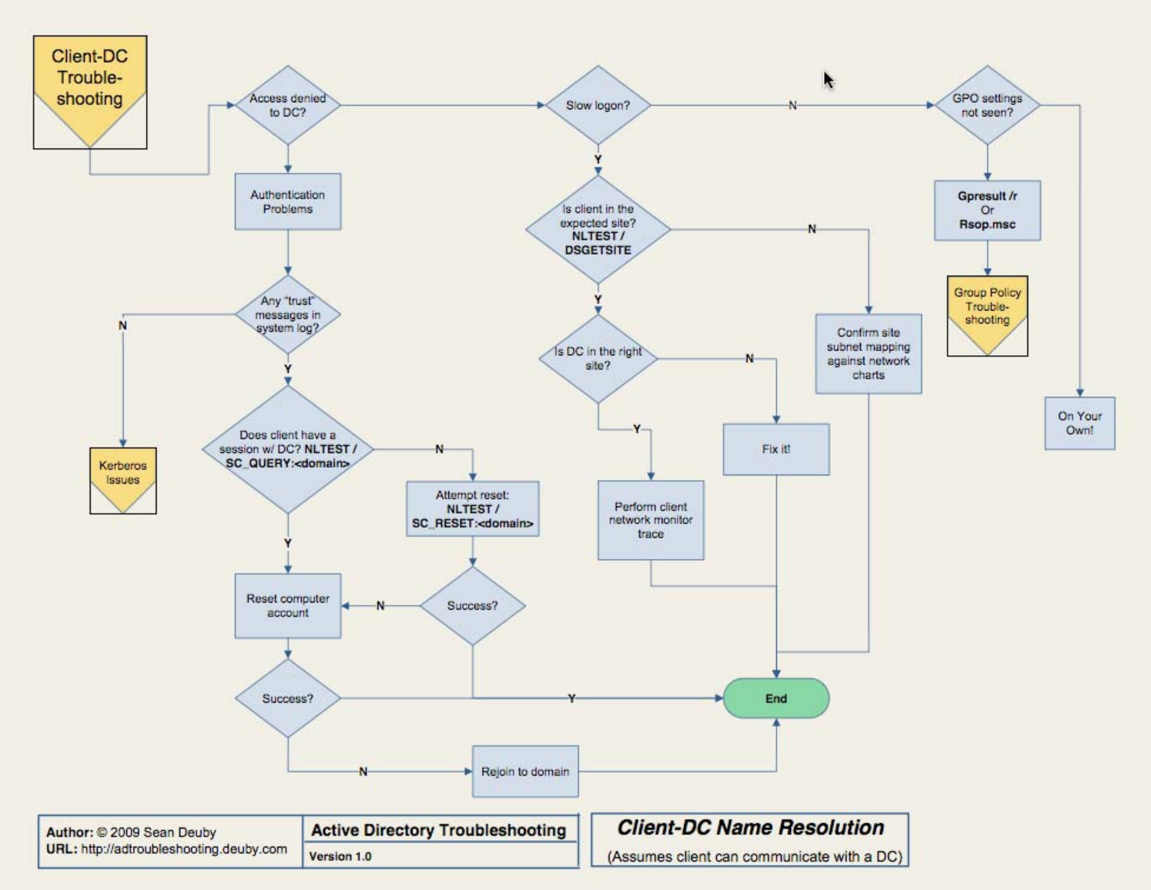
Client-DC Trouble-shooting
Access denied to DC?
Slow logon?
N
GPO settings not seen?
Authentication Problems
Y
Is client in the expected site? NLTEST / DSGETSITE
N
Gpresult /r Or Rsop.msc
Group Policy Trouble-shooting
Any "trust" messages in system log?
N
Y
Is DC in the right site?
N
Confirm site subnet mapping against network charts
On Your Own!
Y
Kerberos Issues
Does client have a session w/ DC? NLTEST / SC_QUERY:<domain>
N
Fix it!
Attempt reset: NLTEST / SC_RESET:<domain>
Perform client network monitor trace
Y
Reset computer account
N
Success?
Success?
Y
End
N
Rejoin to domain
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
Client-DC Name Resolution
(Assumes client can communicate with a DC)

Figure 3.5: Client-DC troubleshooting.

Here, you'll have to personally observe symptoms. For example, are you getting "Access Denied" errors on the client, or does logon seem unusually slow for the time of day? Are you logging on but not getting Group Policy Object (GPO) settings applied? You'll rely heavily on Nltest to verify client-domain controller connectivity and communications; you could wind up dealing with Kerberos issues, which we'll come to later in this chapter.

This is also the point where you're going to want a chart of your network so that you can confirm which domain controllers should be in which sites. You'll want that chart to also list each subnet that belongs to each site. You have to verify that reality matches the desired configuration, and don't skip any steps. It seems obvious to assume that a client was given a proper address by DHCP and is therefore in the same site; *don't ever* make that assumption. I once had a client that seemed to be working just fine but was in fact hanging onto an outdated IP address, making the client believe it was in a different site. The way our LAN was configured, the incorrect IP address was still able to function (we used a lot of VLAN stuff and IP addressing got incredibly confusing), but the client didn't see itself as being in the proper site—so it wouldn't talk to the right domain controller.

Replication Issues

If the flowchart has gotten you to this point, we're dealing with the page Figure 3.6 shows.

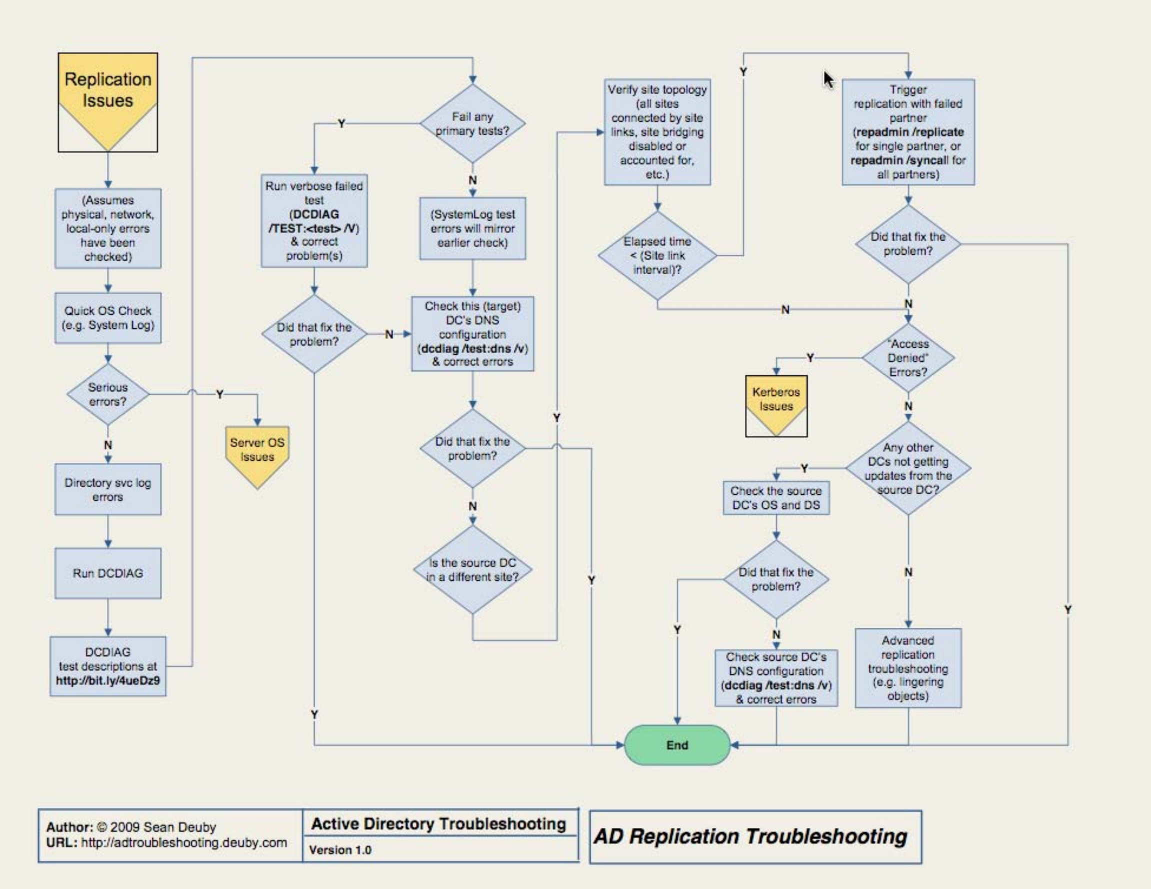
Replication Issues
(Assumes physical, network, local-only errors have been checked)
Quick OS Check (e.g. System Log)
Serious errors?
Y
N
Server OS Issues
Directory svc log errors
Run DCDIAG
DCDIAG test descriptions at http://bit.ly/4ueDz9
Fail any primary tests?
Y
N
Run verbose failed test (DCDIAG /TEST:<test> /V) & correct problem(s)
(SystemLog test errors will mirror earlier check)
Did that fix the problem?
N
Y
Check this (target) DC's DNS configuration (dcdiag /test:dns /v) & correct errors
Did that fix the problem?
Y
N
Is the source DC in a different site?
Y
Verify site topology (all sites connected by site links, site bridging disabled or accounted for, etc.)
Elapsed time < (Site link interval)?
Y
N
Trigger replication with failed partner (repadmin /replicate for single partner, or repadmin /syncall for all partners)
Did that fix the problem?
Y
N
"Access Denied" Errors?
Y
Kerberos Issues
N
Any other DCs not getting updates from the source DC?
Y
N
Check the source DC's OS and DS
Did that fix the problem?
Y
N
Check source DC's DNS configuration (dcdiag /test:dns /v) & correct errors
Advanced replication troubleshooting (e.g. lingering objects)
End
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
AD Replication Troubleshooting

Figure 3.6: Replication issues.

Troubleshooting AD replication is often perceived as the most difficult and mysterious thing you can do with AD. It's like magic: either the trick works or it doesn't, and you'll never know why either way. I see more people struggle with replication issues than with anything else, yet replication is the one thing that can come up most frequently, due in large part to its heavy reliance on proper configuration and the underlying network infrastructure.

Sean proposes four reasons, which I agree with, that make replication troubleshooting difficult for people. In my words, they are:

- They've not been trained in a formal troubleshooting methodology. More admins than you might believe tend to troubleshoot by rote, meaning they try the same things in the same order every time—which is good—without really understanding what they're testing—which is bad.
- They don't approach the problem logically. *Think* about what's happening. Does it make sense to test name resolution between two domain controllers when other communications between them seem unhindered?
- They don't understand how replication works. This, I think, is the biggest problem. If you don't understand what's happening under the hood, you have no means of isolating individual processes or components to test them. If you can't do that, you can't find the problem.
- They don't understand what the tools do. This is also a big problem because if you don't really know what's being tested, you don't know how to eliminate potential root causes from your list of suspects.

Ultimately, *you can't just run tools in the order someone else has prescribed.* Sean proposes four steps to help proceed; I prefer to limit the list to three:

1. Form a hypothesis. What do you *think* the problem is? A firewall rule? IP addressing problem? DNS problem? Apply whatever experience you have to just pick a problem that seems likely.
2. Predict what will happen. In other words, if you think external communications might be failing, you might predict that internal communications will be fine.
3. Test your prediction. Use a tool to see if you're right. If you are, you've narrowed the problem domain. If you're not, you form a new hypothesis.

If you remember science class from elementary school, you might recognize this as the *scientific method,* and it works as well for troubleshooting as it does for any science.

Replication troubleshooting *cannot proceed* unless you've already resolved networking, local-only issues, and other problems that precede this step in the core flowchart. Once you've done that, you'll find yourself quickly looking for OS-related issues in the event log, then move on to the Dcdiag tool—the flowchart provides a URL with a description of the tests to run.

You'll also have to exercise human review and analysis. Do your site links, for example, match your big network chart printout? In other words, are things configured as they should be? This is where a change-auditing tool can save a ton of time. Rather than manually checking to make sure all your sites, site links, and other replication-related configurations are right, you could just check an audit log to determine whether anything's changed. In fact, some change-auditing tools will alert you when key changes happen—like site link reconfigurations—so that you can jump on the problem before it becomes an issue in the environment.

AD Database Issues

Next, you'll move into troubleshooting the AD database, which is covered in the flowchart that Figure 3.7 shows.

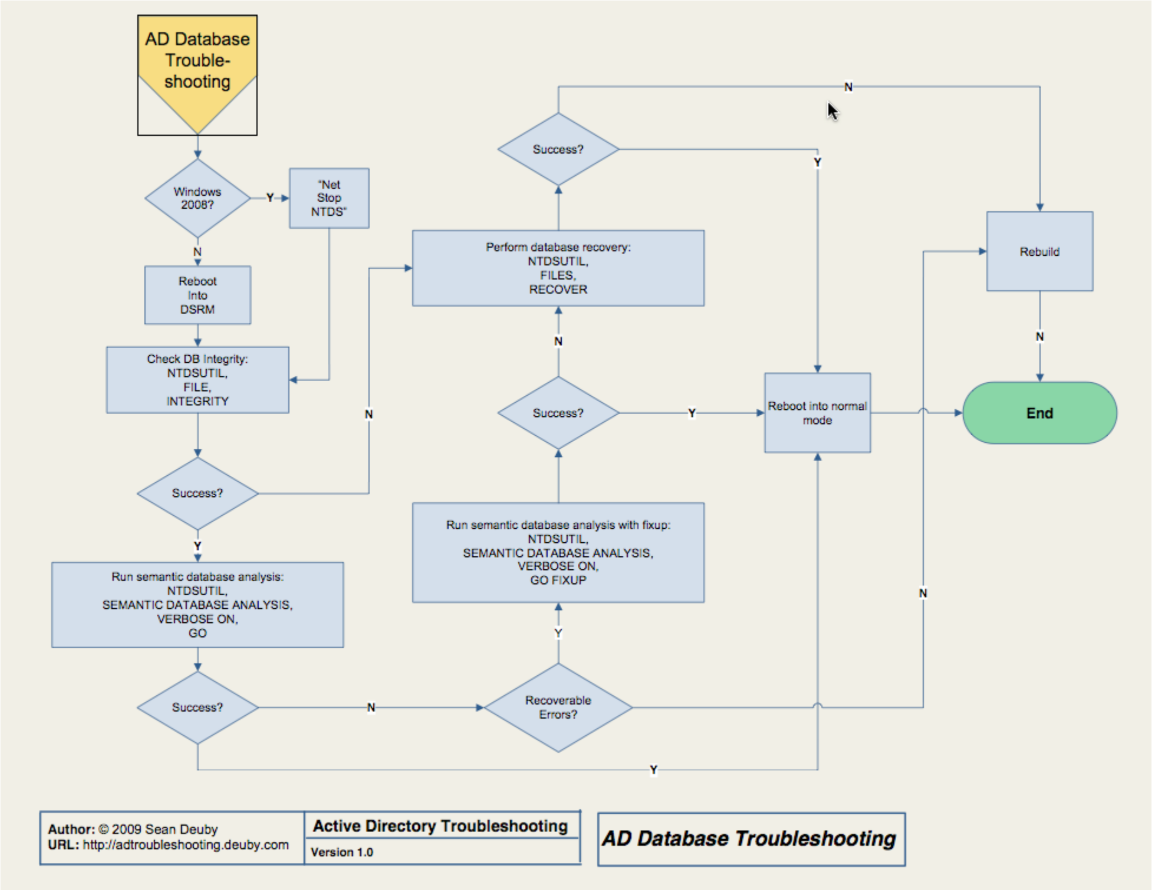
AD Database Trouble-shooting
Windows 2008?
Y
"Net Stop NTDS"
N
Reboot Into DSRM
Check DB Integrity: NTDSUTIL, FILE, INTEGRITY
Success?
N
Y
Run semantic database analysis: NTDSUTIL, SEMANTIC DATABASE ANALYSIS, VERBOSE ON, GO
Success?
N
Y
Recoverable Errors?
Y
N
Run semantic database analysis with fixup: NTDSUTIL, SEMANTIC DATABASE ANALYSIS, VERBOSE ON, GO FIXUP
Success?
N
Y
Perform database recovery: NTDSUTIL, FILES, RECOVER
Success?
Y
N
Reboot into normal mode
Rebuild
N
End
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
AD Database Troubleshooting

Figure 3.7: AD database troubleshooting.

Here, you'll probably be taking a domain controller offline so that you can reboot into Directory Services Restore Mode (DSRM)—make sure you know the DSRM password for whatever domain controller you're dealing with. You'll use NTDSUTIL to check the file integrity of the AD database itself because, at this point, we're starting to suspect corruption of some kind. If you find it, you'll be doing a database restore. If you don't have a backup, you're probably looking at demoting and re-promoting the domain controller, if not rebuilding the serve entirely. Sorry.

Again, this is where third-party tools can help. You may have thought that the "AD Recycle Bin" feature of Windows Server 2008 R2 was a great feature, but it isn't designed to deal with a total database failure. Third-party recovery tools (which are available from numerous vendors) can get you out of a jam here. Make sure you're not using too old a backup; ideally, domain controller backups shouldn't be older than a few days. Older backups will require the domain controller to perform a lot more replication when it comes back online, and a *very* old backup can re-introduce tombstoned (deleted) objects to the domain, which would be a Bad Thing.

Group Policy Issues

If you've made it this far, AD's most complex components are working, and you're on to troubleshooting one of the easier elements. First, recognize that there are two broad classes of problem with Group Policy: no settings from a Group Policy object are being applied or the wrong settings are being applied. This chapter, as shown in the flowchart in Figure 3.8, is concerned only with the former. If you're getting settings but not the right ones, you need to dive into the GPOs, Resultant Set of Policy (RSoP), and other tools to discover where the wrong settings are being defined.

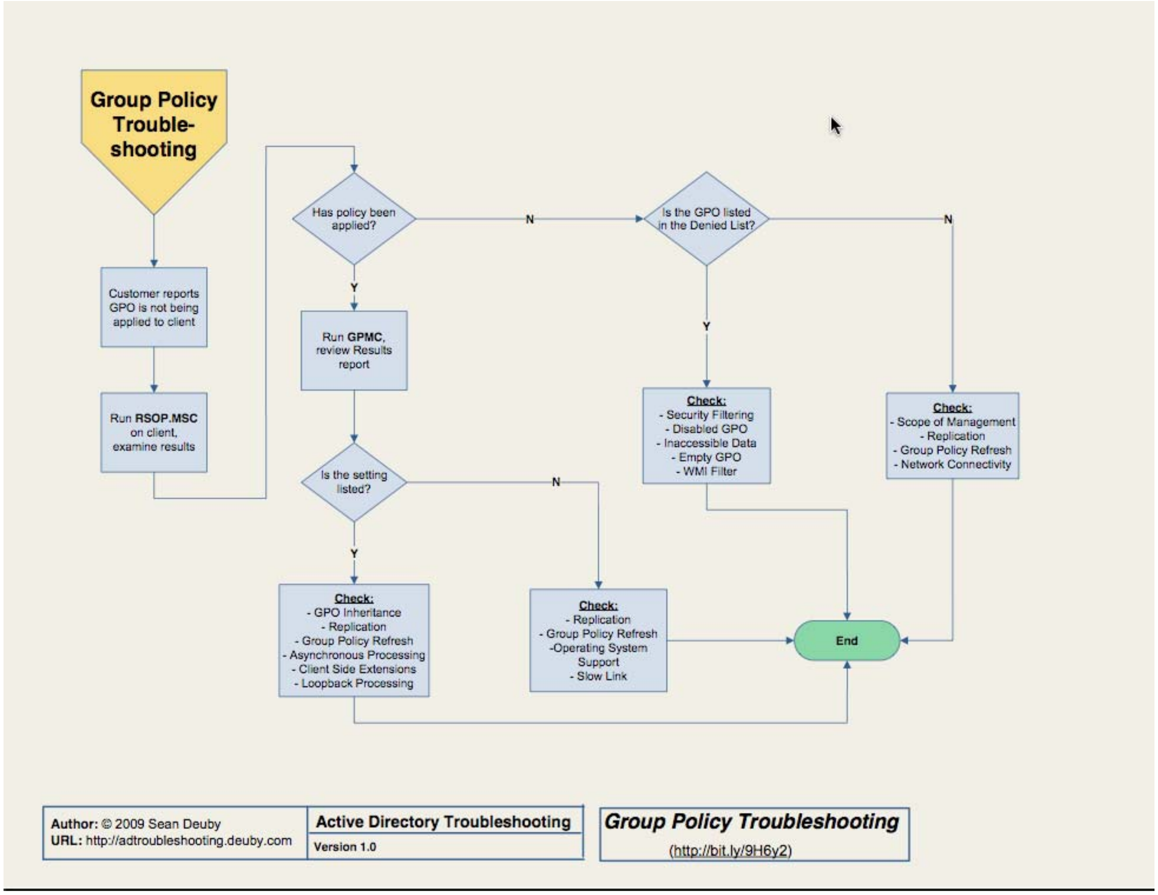

Figure 3.8: Group Policy troubleshooting.

Troubleshooting GPOs is pretty much about verifying their configuration. If a user isn't getting a specific GPO, the problem will be due to replication, inheritance, asynchronous processing (which means they're getting the GPO, just not as quickly as you expected), and so forth. Group Policy is complicated, and knowing all the little tricks and gotchas is key to solving problems. I recommend buying Jeremy Moskowitz' latest book on the subject; he's pretty much the industry expert on Group Policy and his books comes with great explanations and flowcharts to help you troubleshoot these problems.

Unraveling "what's changed" is also the easiest way to fix GPO problems. Unfortunately, most tools that track AD configuration changes don't touch GPOs because GPOs aren't stored in AD itself. There are tools that can place GPOs under version-control, and can help track the changes related to GPOs that do live in AD (such as where the GPOs are linked). Quest, NetWrix, Blackbird Group, and NetIQ all offer various solutions in these spaces.

Kerberos Issues

Finally, the last area we'll cover is Kerberos. Figure 3.9 shows the last page in the flowchart.

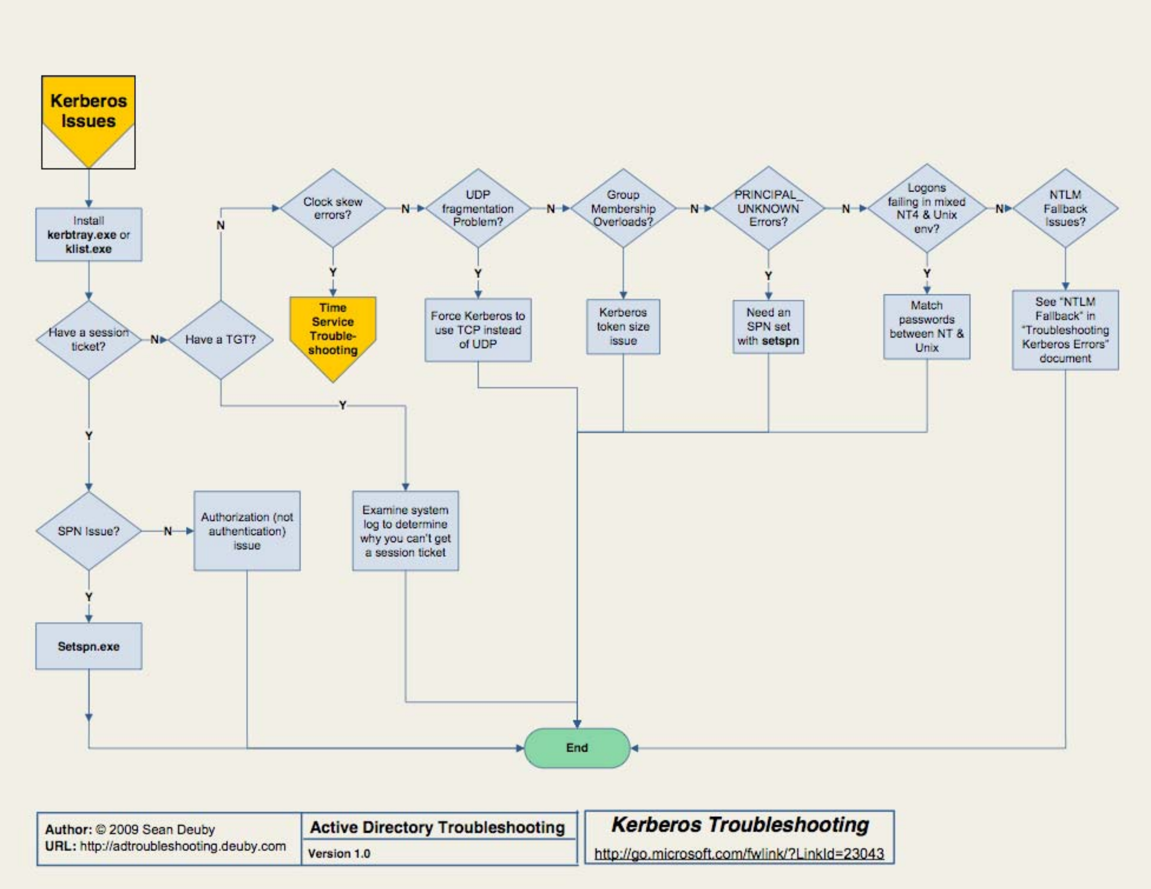
Kerberos Issues
Install kerbtray.exe or klist.exe
Have a session ticket?
N
Have a TGT?
N
Clock skew errors?
N
UDP fragmentation Problem?
N
Group Membership Overloads?
N
PRINCIPAL_ UNKNOWN Errors?
N
Logons failing in mixed NT4 & Unix env?
N
NTLM Fallback Issues?
Y
Time Service Trouble-shooting
Y
Force Kerberos to use TCP instead of UDP
Kerberos token size issue
Need an SPN set with setspn
Match passwords between NT & Unix
See "NTLM Fallback" in "Troubleshooting Kerberos Errors" document
Y
Examine system log to determine why you can't get a session ticket
SPN Issue?
N
Authorization (not authentication) issue
Y
Setspn.exe
End
Author: © 2009 Sean Deuby
URL: http://adtroubleshooting.deuby.com
Active Directory Troubleshooting
Version 1.0
Kerberos Troubleshooting
http://go.microsoft.com/fwlink/?LinkId=23043

Figure 3.9: Kerberos issues.

Here, you'll need to install resource kit tools, preferably Kerbtray.exe, so that you can get a peek inside Kerberos. You'll also need a strong understanding of how Kerberos works. Here's a brief breakdown:

- When you log on, you get a Ticket-Granting Ticket (TGT) from your authenticating domain controller. This enables you to get Kerberos tickets, which provide access to a specific server's resources. Each server you access will require you to have a ticket for that server. So each time you access a new server every day, you'll have to first contact a domain controller to get that ticket.
- Ticket validity is controlled by time stamps. Every machine in the domain needs to have roughly the same idea of what time it is, which is why Windows automatically synchronizes time within the domain. A skew of about 5 minutes is allowed by default.

- Tickets are a bit sensitive to UDP fragmentation, meaning you need to look at your network infrastructure and make sure it isn't hacking UDP packets into fragments. You can also force Kerberos to use TCP, which is designed to handle fragmentation.

There are a few other uncommon issues also covered by the flowchart.

Coming Up Next

With this troubleshooting guidance under your belt, it's time to move on to our next AD topic: security. I've seen an incredible amount of confusion and misinformation with regard to AD security over the past few years, so we're going to start by stepping back to basics and looking at AD's security architecture. We'll spell out AD's *real* role in securing your organization's resources, and look at reasons you might want to re-think your current security design. We'll even peek at DNS security. It's all coming up in Chapter 4.

Download Additional eBooks from Realtime Nexus!