

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

Abstract

- Troubleshoot Microsoft Active Directory (AD) replication.
- Verify that the target Microsoft Active Directory (AD) domain controller's (DC's) operating system (OS) and directory service are working properly.
- Verify that the Domain Name System (DNS) directory service is working properly on both the target and source Microsoft Active Directory (AD) domain controllers (DCs).
- Verify that the target Microsoft Active Directory (AD) domain controller (DC) can resolve the source Microsoft Active Directory (AD) domain controller (DC).
- Check the Kerberos authentication protocol and the services Kerberos depends on.
- Check firewall configurations, because some firewall configuration changes might block replication.

Active Directory (AD) replication is the method by which directory changes made at one domain controller (DC) pass to other DCs. AD is a very robust and fault-tolerant service. Because AD is distributed across many DCs, losing parts of the whole doesn't cripple the overall directory service. In an AD forest you must monitor not only the DCs' basic health but also replication between the DCs. In my experience, replication in an unmonitored forest tends to fall apart over time, even if you configured the DCs carefully. Monitoring and repairing replication problems when they occur is much easier than fixing a forest with accumulated problems. But regardless of whether or not you monitor your AD replication, you'll inevitably need to troubleshoot it. In this article I explain some basic replication principles, and I present a straightforward methodology for troubleshooting AD replication problems. AD replication troubleshooting can be confusing; following my steps will help remove the "black art" feel from this task.

The Basics

Determining the best approach for troubleshooting replication can be difficult. Certainly you want to use the method that resolves the problem as quickly as possible; however, you don't want to skip a step that ultimately drags out the troubleshooting process. I've always found that a logical, inside-out approach works the best. Wanting to immediately dive into the fancy troubleshooting tools is only natural, but you should first use a logical approach to verify that the basics are working correctly. As you get more comfortable troubleshooting replication problems, you might breeze through many of the steps. But first you need the experience of carefully performing each step, to ensure that you don't jump to the wrong conclusion or have to go back to a previous step. Start by checking the health of the OS on the DC itself, then check the health of the directory service. Next, check the DC's basic communications with its fellow DCs. Finally, verify the protocol that the directory service uses and determine whether the DCs are authenticating correctly with one another. Following these steps will help you resolve 90 percent of your replication problems.

Check the Foundation

Your first step is to verify that the DC's OS is working correctly. Replication errors can be caused by various local errors on a DC, so you need to ensure that the server's foundation is sound. If you haven't already done so, install the latest Windows Server Support Tools on all your production systems.

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

Resolving all the issues in a large environment might be impractical; however, you can use an Internet search engine and the Microsoft Knowledge Base (<http://support.microsoft.com/search/?adv=1>) to weed out your most significant problems.

After you install the Windows Server Support Tools, look at the event logs. First, check the system log for warnings and errors. If you encounter errors in the system log, try running NetDiag. Even without using any of its command-line options, NetDiag runs 23 tests related to the system's network configuration. Some of the useful tests it runs are domain membership, DNS, client configuration, trust relationships, Kerberos, and LDAP functionality. If you find a problem area, rerun NetDiag with the /test:testname switch and the /v option to get a detailed test analysis of the area. An important NetDiag option that I refer to later in the article is the /fix switch, which reregisters the server's DNS entries.

If the directory service log has errors, run DCDiag. DCDiag is a comprehensive test utility for DCs. Even without using any of its command-line options, DCDiag runs 27 DC-related tests. As for NetDiag, if you find a problem area, rerun DCDiag with the /test: testname switch and the /v option to get a detailed test analysis of the area. Don't get too hung up over errors in the system log test; any recent errors in the system log will cause this test to fail.

If everything looks good on the home front—that is, if NetDiag and DCDiag didn't reveal any OS or directory service-related errors—it's time to start looking at replication. The best place to start is to check your DCDiag test results, because DCDiag runs extensive replication tests.

Let's use the domain that Figure 1 shows to see how to troubleshoot a common error. This domain, called Deuby.net, has three DCs. The DCs named Godan and Kohai are in the Hub site. The DC named Sandan is in the Branch site, connected to the Hub site by a site link with a replication interval of 15 minutes. Suppose that updates aren't replicating from Kohai to Godan. Replication is always an inbound operation. Thus, even though replication in a site is triggered by change notifications from a DC that has been updated, you need to think of updates as being pulled in by the target DC from another DC. Start your troubleshooting efforts with the DC that should be receiving the updates. In my example, this DC is Godan.

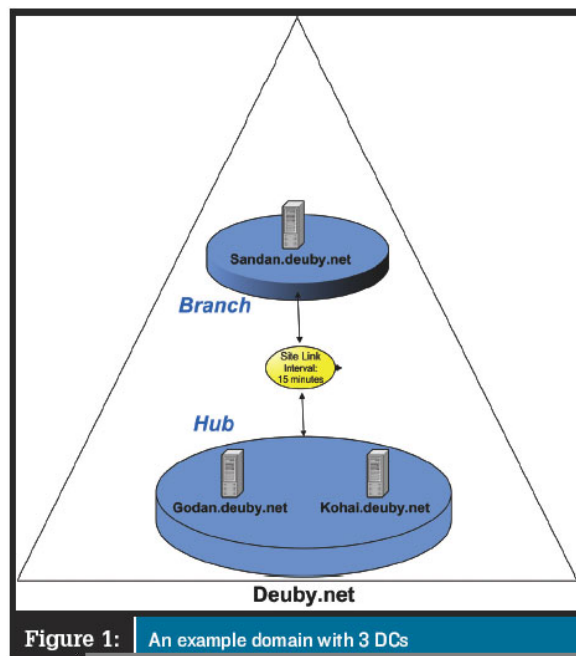


Figure 1: An example domain with 3 DCs

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

Figure 2 shows the partial output from running DCDiag on Godan. Notice that the Replications test failed because of the error "[KOHAI] DsBindWithSpnEx() failed with error 1722, The RPC server is unavailable." Although this error message is dense, we can work through the message to get a good idea of the problem. The problem obviously has something to do with the DC Kohai. But what does "DsBindWithSpnEx()" mean? "BindWithSpn" tells us that the error occurred when Godan attempted to bind (i.e., connect and authenticate) to Kohai. Therefore the problem appears to be related to Godan unsuccessfully communicating with Kohai.

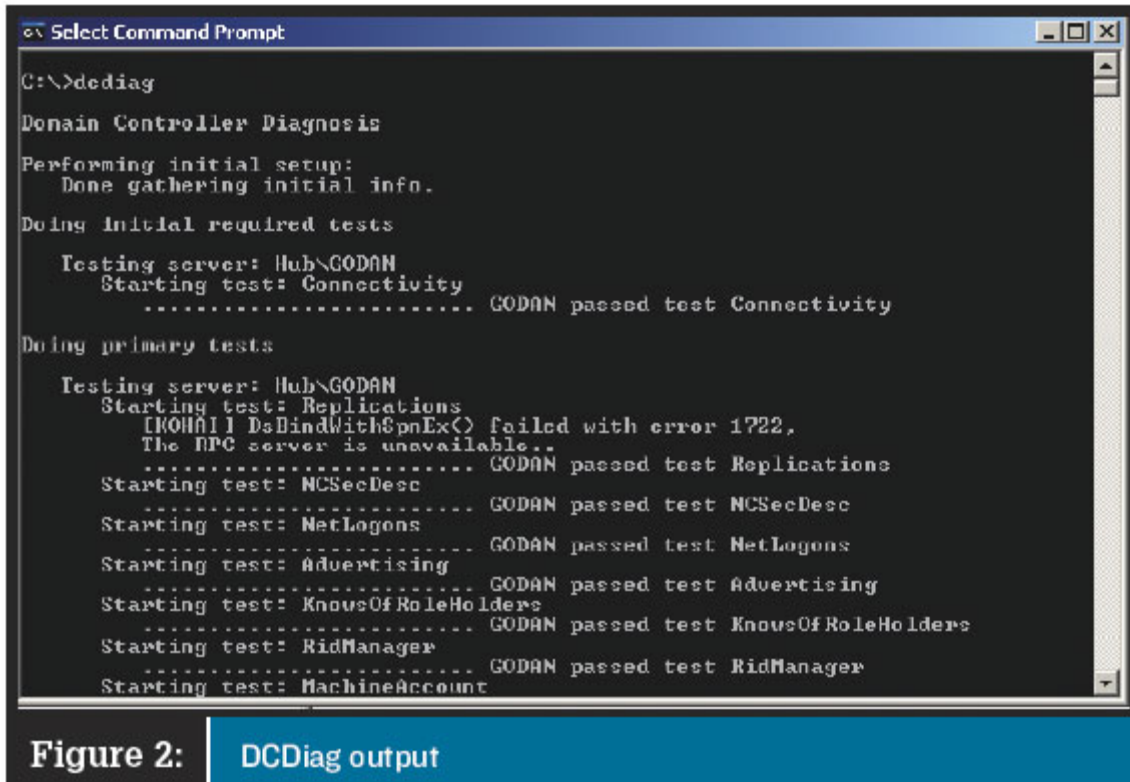


Figure 2: DCDiag output

We need to determine whether Godan can even locate its replication partner Kohai. One of the first tests is to ping Kohai's IP address to check basic network connectivity. If this test works, you could ping Kohai by name (i.e., by its DNS A record). However, this method isn't a conclusive test for replication because a DC finds its replication partners not by resolving their A records (e.g., dc1.mycompany.com), but by resolving a special DNS Canonical Name (CNAME—i.e., alias) guaranteed to be unique in the forest.

Each DC in the forest must register its CNAME record for the name DsaGuid._msdcs.ForestName; this CNAME identifies the DC to the replication system as a DC. The CNAME record maps this string to the DC's A record, which contains its IP address. For example, the DNS CNAME of dc1.mycompany.com might be d40c01da-23fa-46e6-8bf3798503e2590f._msdcs.mycompany.com.

The CNAME record would be d40c01da-23fa-46e6-8bf3798503e2590f._msdcs.mycompany.com CNAME dc1.mycompany.com. Note that the directory service agent (DSA) globally unique identifier (GUID) that comprises the first part of the DC's CNAME isn't the GUID (specifically, the objectGUID attribute) of the DC's computer object, as you might expect. Instead, it's the GUID of the NTDS Settings object under the DC in the Sites container. For example, if DC1 were in the Hub site, its

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

distinguished name (DN) would be CN=NTDS Settings, CN=DC1,CN=Servers, CN=Hub,CN=sites, CN=configuration, DC=mycompany,DC= com.

If the troubled DC can't resolve its replication partner's CNAME, it won't be able to receive updates from it. So, first you must determine Kohai's GUID-based DNS CNAME; then you need to see if Godan can resolve the CNAME. Probably the simplest way to do this is to launch Active Directory Sites and Services (dssite.msc), drill down into Godan's site (i.e., Hub, Servers container, KOHAI computer object), then right-click the NTDS Settings container and select Properties. Kohai's CNAME is in the DNS Alias field, as Figure 3 shows. Copy and paste this string into a Ping command in a command prompt on Godan, as Figure 4 shows, to determine whether the replication engine can resolve Kohai. Because DNS can't resolve Kohai via its CNAME, replication can't occur. Note that a similar query using Godan's CNAME resolves correctly.

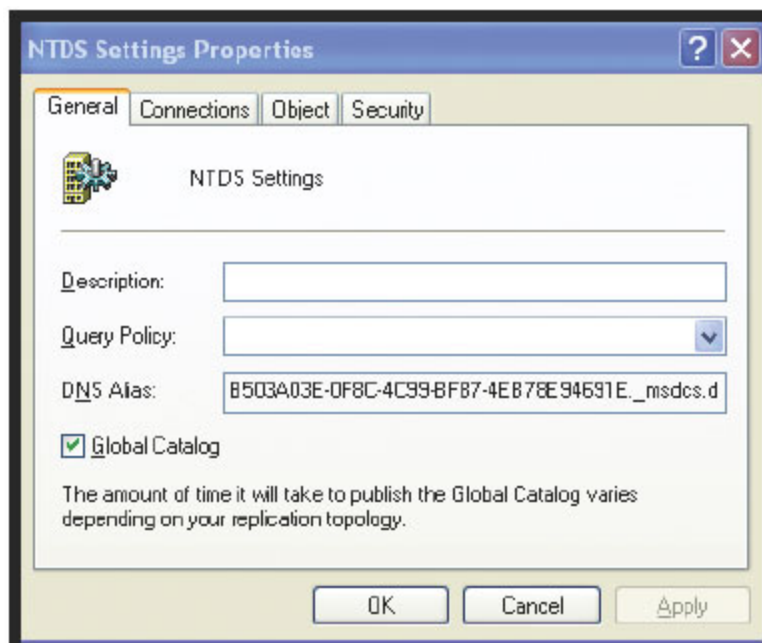


Figure 3: Viewing Kohai's CNAME

We've determined the problem to be that Kohai's DNS CNAME is missing. Two methods exist for reregistering it. A straightforward solution is to stop and start the Netlogon service; however, this method will also temporarily disrupt communication between Kohai and its active users. Just because a DC is having replication problems doesn't necessarily mean it isn't servicing its users. A less intrusive solution is to run NetDiag /fix. The /fix parameter is specifically to reregister all necessary DNS records for a DC. After this step takes place, NetDiag runs cleanly but with a warning that the newly added CNAME will take a while to replicate to the DNS server that's specified as secondary in the network card's DNS configuration.

Solution Steps

1. Make sure the target DC's OS and directory service are working properly.
2. Make sure DNS is working properly on both the target and source DCs.
3. Make sure the target DC can resolve the source DC.
4. Check Kerberos and the services it depends on.

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

5. Check firewall configurations.

DNS Misconfiguration

My example shows why DNS misconfiguration is the most common root cause for AD problems. DNS configuration is complex and tightly integrated into AD's functionality; many ways exist to misconfigure DNS. If you're getting DNS lookup failures or other "can't locate" type errors for a DC, you need to check the following settings.

Verify that the IP addresses in the DC's client settings (TCP/IP properties of the local area connection) are correct. If the DC is also a DNS server, the recommended configuration is for the primary DNS entry to point to itself. (Longhorn Server will automatically configure the DNS entry to loopback—127.0.0.1—when Dcpromo runs.) The secondary DNS server should point to another DC in the same domain. For more information about the pros and cons of different kinds of DNS client configurations for DCs, see the Microsoft article "Best practices for DNS client settings in Windows 2000 Server and in Windows Server 2003" (<http://support.microsoft.com/kb/825036>).

A DC's primary DNS server is its only means of locating other resources on the network. Thus, you can control a DC's knowledge of other servers and domains by controlling its primary DNS entry. If you're wondering whether the DC's own DNS server is working, point the primary DNS entry to a DNS server you know is working correctly.

Make sure the DC has already registered the resource records it needs to function. Three records relate to replication. Two of these are the CNAME (discussed previously) and its A record (i.e., host name to IP address translation). You can run `DCDiag /test:connectivity` to confirm that these records are registered in the DC's primary DNS server, and you can use the `NetDiag` command to reregister the records if necessary. If the records still won't register, run `DCDiag /test:Registerindomain /Dns Domain:dnsdomainname` to verify that the DC is configured correctly to be able to perform the registration. The A record must also map to the correct IP address, and remember that these registrations must replicate to the DNS servers its partners use before they can find it as well. (Note that the `IPConfig /RegisterDNS` command doesn't register all the DNS records a DC requires.)

For child domain DCs in a domain tree configuration, you need to check a third record: the glue record. Glue records are A records of DNS servers (in other words, your DCs) for the forest's child domains, kept in the root domain's forward lookup zone. Glue records help solve a sort of catch-22 circular reference dilemma: To find a host in a child domain from outside that domain, you need to talk to a DNS server that's authoritative for the domain; however, you can't resolve that authoritative server, because its A record is in the very domain you're trying to get DNS information about! Putting a second set of A records for the child domain's DNS servers in the root domain solves this reference problem and thus "glues" the child domains to the root. The `DCDiag DNS` test with the `/DnsDelegation` option (`DCDiag /test:DNS /DnsDelegation`) tests for correct registration of a DC's glue records.

Access Denied

The second most common group of errors revolves around a DC's denial of access to its replication partner. Under normal circumstances access problems don't occur because all DCs' machine accounts are members of the Enterprise Domain Controllers built-in group. Thus, an Access Denied error means something has happened to invalidate the security between replication partners. One of the most common root causes is incorrect time synchronization on one of the DCs. Replication itself doesn't depend on time—but Kerberos does. Kerberos demands tight time synchronization between DCs; if their internal clocks differ by more than five minutes (by default), Kerberos will fail and you'll

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

receive an error message that says access to the source DC was denied. The system log will have Kerberos and probably W32Time errors.

Use the Net Time /QuerySNTP command to see which time servers are configured for the DC in question. A DC is a member of a domain by definition; if a DC isn't the PDC emulator of the root domain, its time server configuration should be empty, because the default Network Time Protocol (NTP) server for a non-PDC DC is the PDC of its domain. If the DC in question is in a child domain, the PDC looks to a DC in the root domain as a time source, and these DCs in turn look to the PDC of their parent domain (usually the root domain) as the authoritative time source for the entire forest. Use the Net Time /SetSNTP: command to remove references to an explicit time server. You can then use the handy W32tm command to control the DC's NTP settings. To force the DC to locate a time source and synchronize with it, run the W32tm /Resync /Rediscover command. You could also run the W32tm /Config /Syncfromflags:DOMHIER command to sync from a DC in the domain hierarchy. To check the NTP settings on all DCs in the domain, run the W32tm /Monitor command. Watch the clock in the system tray to tell when the time changes take effect. (For more information about how Windows Time works, see the Microsoft articles "How Windows Time Service Works," <http://technet2.microsoft.com/windowsserver/f/?en/library/71e7658728f4-4272-a3d7-7f44ca50c0181033.msp>, and "How to configure an authoritative time server in Windows Server 2003," <http://support.microsoft.com/kb/816042>)

In circumstances in which the time has been out of sync for so long that the DC's own Kerberos tickets have expired, you must disable the Key Distribution Center (KDC) on the DC and reboot. (To disable the KDC, stop it in the Control Panel Services applet and set the startup to Disabled.) Taking this step clears out the Kerberos tickets and forces the DC to get new tickets from one of the remaining functional DCs.

Other Tips

First, be patient. Replication in an enterprise takes a while to complete, as well as to correct itself when something goes wrong. For example, when a DC doesn't respond, the Knowledge Consistency Checker (KCC) waits 90 minutes to recalculate connection objects around the DC.

In this era of greater security, consider the possibility that firewall configuration changes might block replication. Look for servers that won't respond to pings even though they're perfectly healthy, or for servers that respond to some protocols but not others. For details about DC port requirements for firewalls, see the Microsoft article "Active Directory Replication over Firewalls" (<http://www.microsoft.com/technet/prodtechnol/windows2000serv/technologies/activedirectory/deploy/confeat/adrepfir.msp>). For a list of port requirements for various Windows Server products, see the Microsoft article "Service overview and network port requirements for the Windows Server system" (<http://support.microsoft.com/kb/832017>).

If you don't think you're getting enough detail from the directory log, go to the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Ntds\Diagnostics subkey and enable NTDS logging. Which key value to increase logging on depends on the area you're investigating (e.g., Knowledge Consistency Checker, Name Resolution, Replication Events). The default value is 0, with a maximum value of 5. A value of 3 is typically the highest you'd need. Monitor the effect that increased logging has on your directory log, and disable the logging when you no longer need it.

Trust the KCC. Resist the temptation to outguess the KCC when it doesn't seem to be creating the topology you planned. If the KCC isn't doing what you expect, something in your site topology

Troubleshoot AD Replication

Sean Deuby

(Reprinted from WindowsItPro Magazine)

probably isn't configured like you think it is. For example, you need to ensure that the DC's IP address corresponds to subnets associated with the site the DC belongs to.

And finally, if you receive errors that indicate the DC hasn't replicated for a period longer than the tombstone lifetime, you can stop trying to troubleshoot. You must rebuild the DC, remove its metadata from AD, and repromote the DC. As an MVP colleague once said, "DCs are like little tin soldiers; you can knock one down and put another just like it in its place."

Put Down Your Wand

Replication is a key function of AD, but troubleshooting replication is often regarded as a black art. To remove the mystical aspect of replication, first use a logical approach to verify that the basics are working; then, verify that the DC's OS is working correctly, check its directory service, check its DNS configuration, check inter-DC communications, check Kerberos and its dependencies (e.g., the Windows Time Service), and check firewall configurations. Following the tips I present in this article will transform AD replication troubleshooting from voodoo into tried-and-true.