Configuring the Windows Time Service

This article walks you through the process of setting up an authoritative time server for a Windows Server 2003-based network running Active Directory. The article outlines procedures for syncing to both an internal and external time source, and also lists additional resources for configuring the Windows Time service and troubleshooting time synchronization problems.

Published: Nov 18, 2004 Updated: Oct 12, 2007 Section: Articles & Tutorials :: Windows 2003 Author: Mitch Tulloch

Reprinted from the original full article here: http://www.windowsnetworking.com/articles_tutorials/Configuring-Windows-Time-Service.html?printversion

The Windows Time service (W32Time) is designed to maintain date and time synchronization for computers running Windows 2000XP/2003. The primary use for such time synchronization is to ensure the security of Kerberos authentication within an Active Directory environment. To prevent replay attacks, Kerberos tickets presented to domain controllers by clients are time-stamped. The authenticating domain controller checks to make sure the timestamp is unique and falls within an allowable skew before accepting the ticket and authenticating the client. To ensure this system works properly, both the client and the domain controller clocks must be loosely synchronized within the allowable skew, and W32Time ensures this is the case.

Synching to an Internal Time Source The simplest solution to time synchronization in an Active Directory environment is to let the PDC Emulator in the forest root domain use its own CMOS clock as the source of reliable time for the forest. To do this, you can simply take no action. The only annoying result is that you will occasionally see the following event logged to the System log in Event Viewer:

Event ID: 12

Event source: W32Time

Event description: Time Provider NtpClient: This machine is configured to use the domain hierarchy to determine its time source, but it is the PDC emulator for the domain at the root of the forest, so there is no machine above it in the domain hierarchy to use as a time source. It is recommended that you either configure a reliable time service in the root domain, or manually configure the PDC to synchronize with an external time source. Otherwise, this machine will function as the authoritative time source in the domain hierarchy. If an external time source is not configured or used for this computer, you may choose to disable the NtpClient.

Basically, what this event means is that the PDC Emulator in the forest root domain has not been configured to synchronize its clock with an external stratum 1 time source, and as a result the clocks on all machines in your forest cannot be considered reliable. Now this may be an issue if employees rely upon their workstations' CMOS clocks for signing in and out, but as far as Kerberos is concerned it's a non-issue because Kerberos only requires that clocks on clients and authenticators agree with each other, not that they display accurate time. So if every machine's clock in the forest is one hour late, Kerberos will still work fine and replay attacks will be prevented, which is the purpose of W32Time anyway.

Synching to an External Time Source If you want to ensure that the clocks on your machines are more accurate in terms of absolute (and not just relative) time, you can sync the PDC Emulator in your

forest root domain to one of the reliable time servers available on the Internet. This is a good idea if your company is a large enterprise with sites spanning several countries, or if your organization has two or more forests linked by forest trusts. The procedure for doing this on a PDC Emulator running Windows Server 2003 in the forest root domain is as follows. Open Registry Editor (regedit.exe) and configure the following registry entries:

```
HKLM\SYSTEM\CurrentControlSet\Services\W32Time\Parameters\Type
```

This registry entry determines which peers W32Time will accept synchronization from. Change this REG_SZ value from NT5DS to NTP so the PDC Emulator synchronizes from the list of reliable time servers specified in the NtpServer registry entry described below.

1. -HKLM\SYSTEM\CurrentControlSet\Services\W32Time\Config\AnnounceFlags

This registry entry controls whether the local computer is marked as a reliable time server (which is only possible if the previous registry entry is set to NTP as described above). Change this REG_DWORD value from 10 to 5 here.

1. -HKLM\SYSTEM\CurrentControlSet\Services\W32Time\Parameters\NtpServer

This registry entry specifies a space-delimited list of stratum 1 time servers from which the local computer can obtain reliable time stamps. The list may consist of one or more DNS names or IP addresses (if DNS names are used then you must append ,0x1 to the end of each DNS name). For example, to synchronize the PDC Emulator in your forest root domain with tock.usno.navy.mil, an open-access SNTP time server run by the United States Naval Observatory, change the value of the NtpServer registry entry from time.windows.com,0x1 to tock.usno.navy.mil,0x1 here. Alternatively, you can specify the IP address of this time server, which is 192.5.41.209 instead.

Now stop and restart the Windows Time service using the following commands:

1. -net stop w32time

1. -net start w32time

It may take an hour or so for the PDC Emulator to fully synchronize with the external time server because of the nature of the polling method W32Time uses. Depending on the latency of your Internet connection, the accuracy of the CMOS clock on your forest root PDC Emulator may be within a second or two of UTC. If you need more accurate time however, you can purchase a hardware time source like an atomic clock and connect it to your PDC emulator.

Alternatively, if you don’t want to wait for time convergence to occur between your stratum 2 time server (your forest root PDC Emulator) and the external stratum 1 time server, you can run the following command on your PDC Emulator:

1. -w32tm /resync /rediscover

Tip There are additional registry settings you can configure to ensure external time synchronization operates effectively, see this article in the Microsoft Knowledge Base for details.