It's Good to be on Time

By Ivan Pepelnjak

1. 4. 2008

The importance of having accurate time on distributed servers (and even personal workstations) has been recognized long time ago by the IT managers, but it hasn't been applied consistently to the networking devices. In this article, I'll describe the importance of time synchronization for networking devices, the basics of Network Time Protocol (NTP) that is commonly used to synchronize IP hosts and routers, how to use it on Cisco routers and IOS-based switches and how to implement it in a highly scalable way.

The Need for Accurate Time

Years ago, the only environment where people would care about accurate time on their networking gear would be academic environments (in many cases simply because it's fun having very precise time on the device that should do nothing more than forward IP packets), but the introduction of encryption-based Virtual Private Networks (VPNs) implemented with IPSec and the public key infrastructure (PKI) based on X.509 certificates required that all certificate users (including routers and VPN concentrators) have

approximately correct time (all X.509 certificates have embedded timestamps defining certificates' validity). This requirement was easily met on high-end routers that have internal real-time clock backed up with a battery. The low-end routers (for example, the 800-series routers) are a different story; unless you synchronize them to an external time source after the reload, they will not establish a VPN tunnel.

The PKI certificates require time that is accurate to a few hours. On the other hand, if you want to perform a distributed analysis of events happening in your network (for example, break-ins, denial-of-service attacks or routing instabilities) or correlate logging printouts stored locally on various devices, the devices participating in the analysis have to be almost perfectly synchronized.

Don't Forget

Even if you have the most accurate time on your routers, it won't be very helpful unless you use it in syslog messages (configured with the service timestamps global configuration command).

Last but not least, if you decide to offload various network services to routers, you could use them as local NTP servers (together with being DHCP- and DNS proxy servers).

NTP Basics

The Network Time Protocol (NTP, RFC 1305) is a simple protocol using UDP port 123. The RFC describes the protocol itself as well as the architectural framework and in-depth implementation recommendations.

The NTP architectural framework specifies a hierarchy of time servers using *stratum* values from one to sixteen to indicate their relative accuracy. The most accurate servers using external clocks (GPS receivers are commonly used due to their low cost) are *stratum one* servers and any server synchronizing itself to a *stratum X* server advertises itself as *stratum X+1* server.

Two NTP servers communicate in client-server or peer-to-peer mode (the desired peering mode is configured manually on the servers and indicated in the outgoing NTP packets). The fundamental difference is the

synchronization behavior: an NTP server can synchronize to a peer with better stratum, whereas it will never synchronize to its client (regardless of the client's stratum).

The upstream NTP *servers* have to be defined manually in the *client* NTP server (or the NTP client); there is no auto-discovery mechanism in the client-server relationships. If you use NTP on the Cisco IOS, the NTP peers have to be defined only on one side; the IOS implementation of the NTP server automatically creates a peer-to-peer association when it receives an incoming NTP packet indicating the remote IP host wants to establish a peering relationship.

Warning: This behavior is a potential threat in security-conscious environments that can easily lead to denial-of-service attacks. It's thus highly recommended that you protect the IOS NTP servers with NTP authentication or access lists; both mechanisms will be described in an upcoming IP Corner article.

NTP clients are no different from the NTP servers (from the protocol or implementation perspective). The typical NTP client implementations can synchronize to multiple NTP servers, select the best server and synchronize with it or even set the local clock to the averaged value returned by the servers.

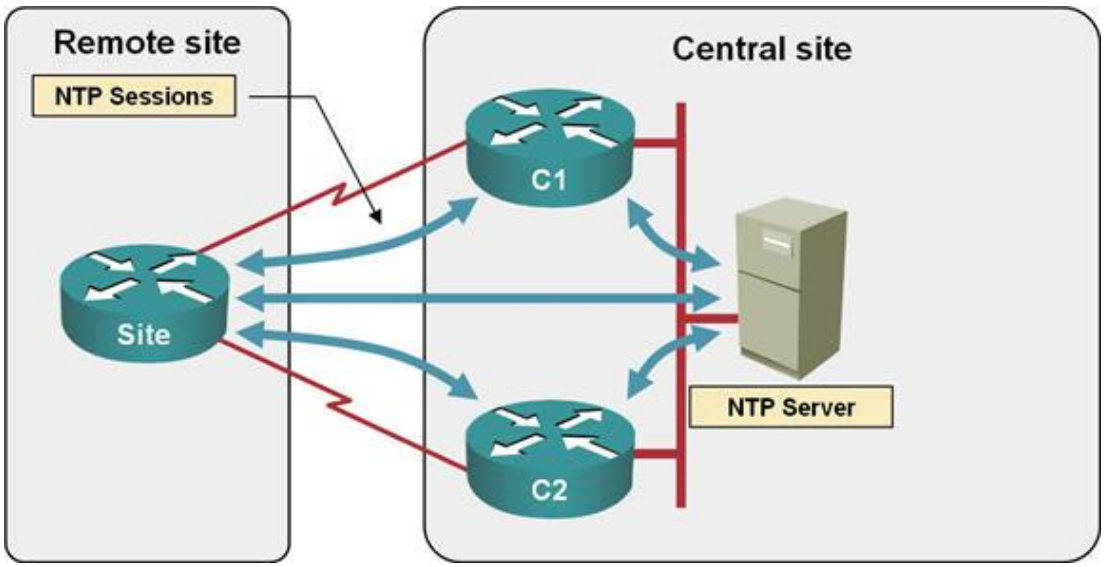
Low-end time synchronization implementations (for example, Windows 2000) typically choose to use Simple Network Time Protocol (SNTP) defined in RFC 4330. SNTP is a small subset of NTP. It interoperates with NTPv3 servers (there is no need to deploy another time synchronization infrastructure to use SNTP) but forgoes the complex time synchronization algorithms of RFC 1305 and replaces them with a simple stateless request-reply protocol, resulting in lower accuracy.

Technical Details: Some low-end Cisco routers support only SNTP client. Mid-range routers can be configured to use either NTP or SNTP. If you want to use a router as an NTP server, it has to synchronize with upstream routers via NTP; if you only need pretty accurate local time, SNTP is a good choice for remote locations.

Design Guidelines

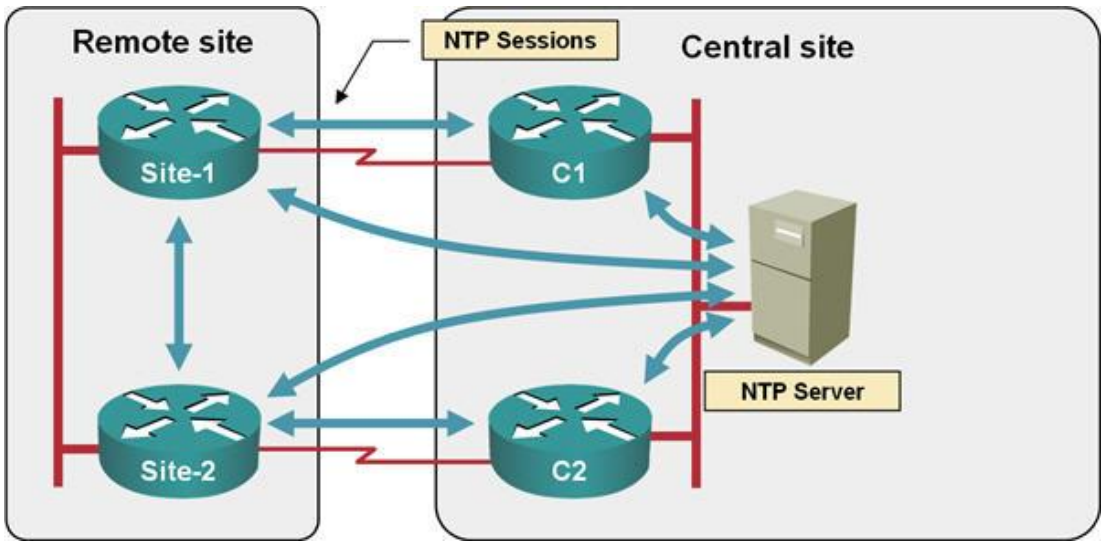
Typical NTP design guidelines recommend that every NTP server uses at least two upstream servers and peers with at least one more server of the same stratum. Of course, these recommendations have to be tailored to the actual network design. For example, if you have remote sites with a single router, it makes no sense for the router to peer with anyone but the upstream routers (or central NTP servers); if the upstream connectivity is gone, it has no reasonable peers anyway (Figure 1).

Figure 1: NTP sessions on a non-redundant site



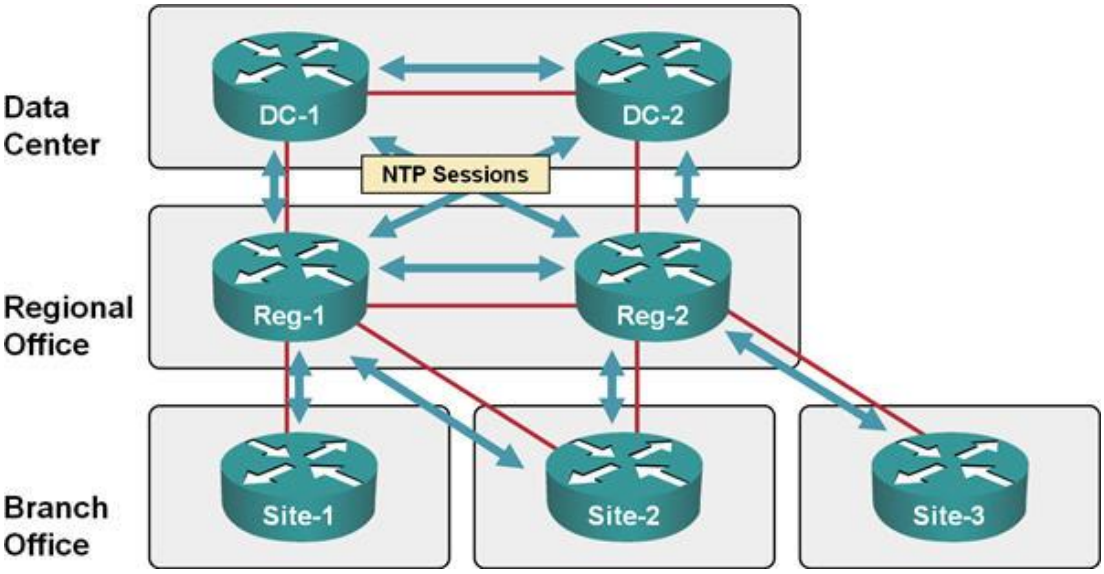
On the other hand, if you have a redundant network design that has two routers on each remote site, it's advisable that you configure them as NTP peers; even if one of them loses upstream connectivity, it can still synchronize to the other one (Figure 2).

Figure 2: NTP sessions on a redundant remote site



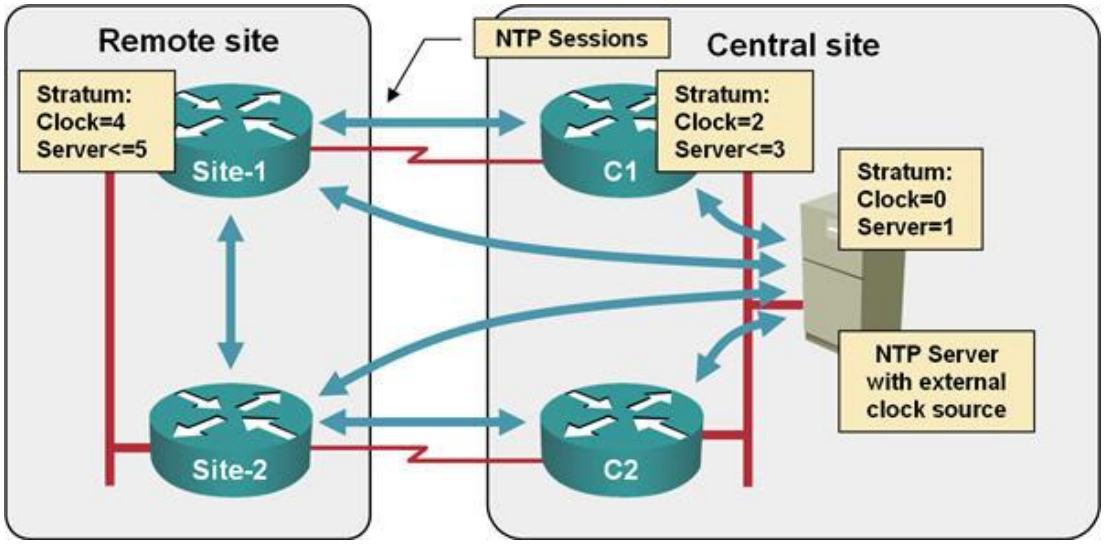
The NTP protocol uses little bandwidth and CPU resources on the routers. A Cisco router sends only a few NTP packets after the reload, trying to achieve fast synchronization with the upstream NTP servers and peers. After the initial synchronization attempt, NTP packets are sent every 64 seconds to unstable peers or newly configured servers. This interval is gradually increased until the packets are sent every 1024 seconds in the steady-state conditions. It's therefore totally harmless to implement NTP topology that follows the actual physical topology of your network. The proposed NTP topology in a typical hierarchical network design is shown in Figure 3.

Figure 3: NTP sessions in a typical hierarchical network



If you want to have a robust distributed NTP architecture in your network, you should allow every NTP server to use its own local clock as one of the time sources ensuring that it will continue to serve meaningful time to its client even if it loses connectivity to all upstream servers and peers. The stratum of the local clock should be set to the worst possible stratum of all the upstream servers increased by one, as shown in Figure 4 (setting strata of local clocks to lower values might prevent time synchronization if the central NTP server fails).

Figure 4: Strata of local clocks in a multi-level NTP network



Warning: If you want to set the stratum of router's local clock to X, you should configure the router to be stratum X+1 server with the ***ntp master*** *configuration command.*

NTP Configuration on Cisco IOS

NTP configuration in Cisco IOS uses global configuration commands that start with the keyword **ntp.** The upstream NTP servers in Cisco IOS are defined with the **ntp server** ***ip-address*** configuration command and the peers are defined with the **ntp peer** ***ip-address*** configuration command.

Note: The router's behavior will not change if you define all neighboring servers with the ***ntp peer*** *configuration commands, but since the router will send out NTP packets indicating peer-to-peer mode, an unprotected upstream server could decide to synchronize to them.*

Once the router synchronizes with its NTP neighbors, it will insert the **ntp clock-period** *some-number* command in the running configuration. This command specifies an estimate of the actual frequency of the local clock (the internal router's clock is believed to be highly precise, but not necessarily running at the correct frequency). Don't change it and make sure it's stored in the NVRAM after the router's time synchronization reaches steady state (upstream NTP servers are polled every 1024 seconds).

It's highly recommended that you track the status of your time synchronization with *syslog* commands; the **ntp logging** command is available in IOS releases 12.3T and 12.4. Some people complain that it generates too much output, but the repeating messages about NTP synchronization loss and subsequent resynchronization usually indicate a real problem somewhere in your network:

- The **ntp clock-period** command could be a bad estimate of the drift of the router's internal clock;
- The NTP server is intermittently unreachable;
- The NTP server is heavily loaded and does not respond to NTP queries;

Technical Detail: A heavily-loaded router is not the best choice for an authoritative NTP server in your network. The replies to the incoming NTP requests are sent from the NTP process, which is a medium-priority process in Cisco IOS. All packet switching activities as well as high-priority processes will be executed before the router is able to reply to an NTP request. In most UNIX implementations (including Linux) you could run the NTP processing in the kernel to get microsecond-level accuracy.

If you use access lists to protect your NTP server, you should ensure that the NTP packets sent by a router are always sent from the same IP address (usually from the router's loopback interface). To specify the source IP address globally, use the **ntp source** *interface* configuration command. To specify the source IP address for a specific NTP peer or upstream server, use the **ntp peer|server** *ip-address* **source** *interface* configuration command.

If you're concerned about the memory consumption of a core NTP server, you can limit the number of associations it supports with the **ntp max-associations** *number* configuration command.

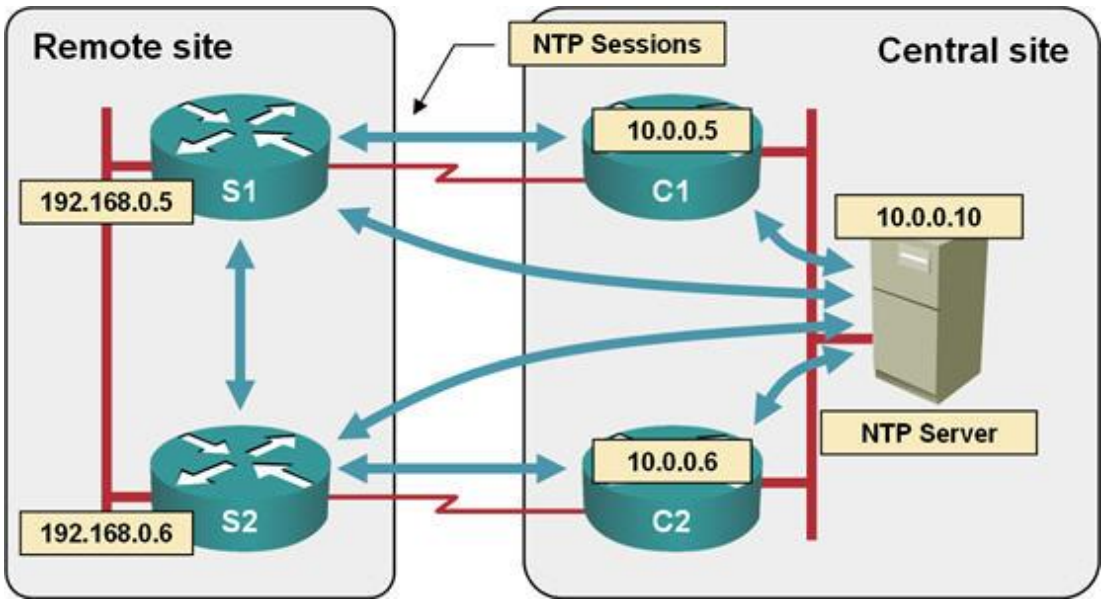
And finally, a router will not be able to act as a standalone NTP server (in case all upstream servers and peers are lost) unless you:

- Store the NTP-synchronized time value in the internal battery backed-up clock with the **ntp update-calendar** configuration command;
- Configure the router to become a stratum X server with the **ntp master** *stratum* configuration command. The *stratum* value you use in the **ntp master** configuration command should be at least the maximum possible stratum value of all upstream NTP servers increased by two.

To configure the NTP server on a remote site S1 router from Figure 5 you could use the configuration commands in Listing 1 that:

- Configure the central server (NTP Server) and the upstream router (C1) to be NTP servers;
- Configure the redundant router (S2) on the same remote site to be NTP peer;
- Configure the router to act as a standalone NTP server with stratum 10;
- Specify that the NTP packets toward the NTP servers should be sent with the source address of the Loopback interface whereas the NTP synchronization with S2 should use the IP address of the Fast Ethernet interface;
- Perform periodic updates of the internal clock.

Figure 5: Sample network using NTP synchronization



Listing 1: NTP configuration on S1

```
ntp logging
ntp source Loopback0
ntp master 10
ntp update-calendar
ntp server NTP-Server
ntp peer S2 source FastEthernet0/0
ntp server C1
```

*Note: The hostnames used in the **ntp server** and **ntp peer** commands are resolved immediately and stored as IP addresses.*

Monitoring NTP

Cisco IOS provides two commands to monitor the status of the embedded NTP server. The **show ntp status** command displays the state of the internal clock (Listing 2). This printout indicates the synchronization status, the stratum of the local NTP server, the internal clock frequency, the current time (reference time) and the offset and dispersion to the NTP server to which the router has synchronized (the mentioned fields are highlighted in Listing 2).

Listing 2: Time synchronization status on S1

```
S1#show ntp status
Clock is synchronized, stratum 7, reference is 10.0.0.10
nominal freq is 250.0000 Hz, actual freq is 250.0005 Hz, precision is 2**18
reference time is CB6D3483.720B1C23 (12:35:15.445 UTC Mon Feb 25 2008)
clock offset is 0.7809 msec, root delay is 1.51 msec
root dispersion is 41.38 msec, peer dispersion is 29.30 msec
```

Note: When the synchronization process with an upstream server reachable over high-speed links has completed, the clock offset and the dispersion should both be not more than 100 milliseconds.

The **show ntp associations** command displays all configured and dynamically acquired NTP servers and peers, their stratum values, reference clocks (IP addresses of the upstream NTP servers), polling intervals, reachability information, delays and offsets (Listing 3).

Listing 3: NTP associations on S1

```
S1#show ntp associations
    address       ref clock    st when  poll reach delay offset   disp
+~10.0.0.5      10.0.0.10       4   33   512  377   45.9  18.60    4.1
 ~192.168.0.6   0.0.0.0        16    -  1024    0    0.0   0.00  6000.
 ~127.127.7.1   127.127.7.1     9   22    64  377    0.0   0.00    0.0
*~10.0.0.10     127.127.1.0     3  436   512  377   45.6 -22.51    6.0
* master (synced), # master (unsynced), + selected, -
 candidate, ~ configured
```

A single NTP neighbor is selected as the NTP master. If the local clock is synchronized to the NTP master, its status is *master (synced*), otherwise it's *master (unsynced).* Other NTP neighbors could be *selected* for potential synchronization should the current master fail or be *candidates* for synchronization.

In-depth NTP association information can be displayed with the [show ntp associations detail] command. Unfortunately, this command does not accept the IP address of the NTP neighbor; the only means of reducing its output is to use the output filters as illustrated in Listing 4.

Listing 4: Detail of NTP association between S2 and NTP server 10.0.0.5

```
S1#show ntp associations detail | begin ^10.0.0.5
10.0.0.5 configured, selected, sane, valid, stratum 4
ref ID 10.0.0.10, time CB6D36DE.B61F1589 (12:45:18.711 UTC Mon Feb 25 2008)
our mode active, peer mode active, our poll intvl 256, peer poll intvl 256
root delay 2.55 msec, root disp 53.92, reach 276, sync dist 59.280
delay 0.66 msec, offset -7.9848 msec, dispersion 2.87
precision 2**18, version 3
org time CB6D3716.B5B90617 (12:46:14.709 UTC Mon Feb 25 2008)
rcv time CB6D3716.B7DA45D6 (12:46:14.718 UTC Mon Feb 25 2008)
xmt time CB6D375F.710E25CA (12:47:27.441 UTC Mon Feb 25 2008)
filtdelay =     0.66    1.80   -0.34    2.30    1.98    2.49    6.68   12.79
filtoffset =   -7.98   -8.56  -10.49  -12.60  -11.89  -10.21   -4.59   13.11
filterror =     0.85    2.81    4.76    6.71    7.69    9.09   10.07   11.05
10.0.0.10 configured, our_master, sane, valid, stratum 3
ref ID 127.127.1.0, time CB6D36E7.BDA843E2 (12:45:27.740 UTC Mon Feb 25 2008
)
… rest of printout deleted …
```

SNTP Configuration

The SNTP configuration in Cisco IOS is (as one would expect) much simpler than the NTP configuration:

- You can configure the SNTP-related logging with the **sntp logging** configuration command;
- Upstream NTP server is configured with the **sntp server** *ip-address* configuration command. You can configure multiple servers for redundancy purposes.

Note: You cannot configure the SNTP process to update the internal clock, as SNTP is supposed to be used solely on the low-end models with no battery backed-up clock.

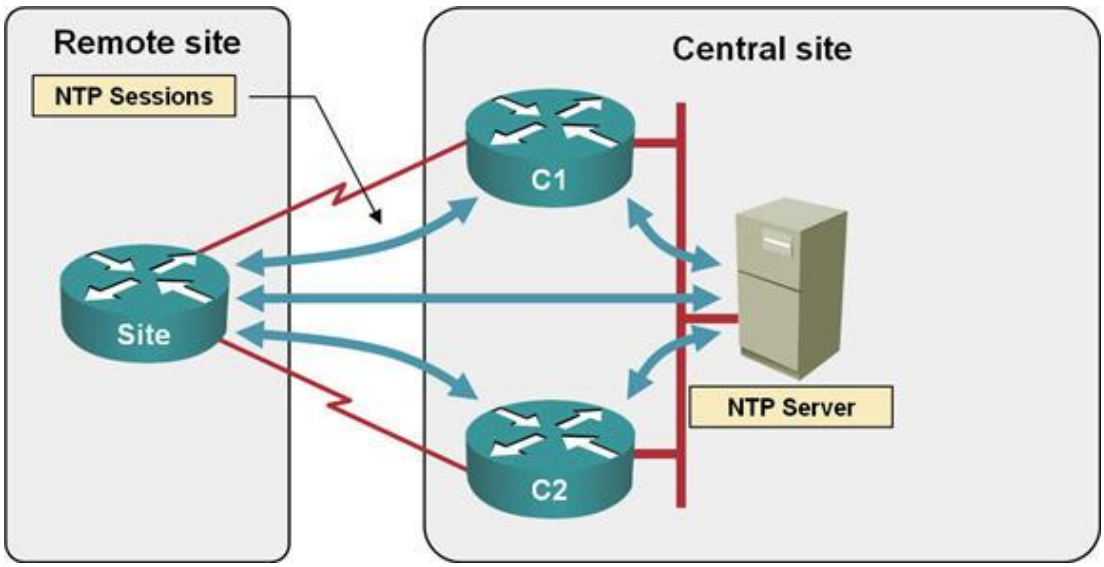
The SNTP process will not synchronize to the configured SNTP servers if you've previously entered any NTP-related configuration commands on the

router (**ntp logging** is enough), as the *NTP* process in Cisco IOS receives replies that should be received by the *SNTP* process (remember: NTP and SNTP use the same UDP port number). The only way to fix this problem is to reload the router.

Warning: A router using SNTP synchronization cannot be used to provide NTP services to downstream clients, as these services would require the NTP process to run, thus blocking the SNTP synchronization.

If you want to configure SNTP on the Site router in Figure 6, you could use the configuration commands displayed in Listing 5.

Figure 6: Simple remote site using SNTP synchronization



Listing 5: Caption

```
sntp logging

sntp server C1

sntp server C2

sntp server NTP-Server
```

The SNTP process should quickly acquire the correct time from the NTP servers and generate *syslog* messages as it synchronizes with the servers (Listing 6). Multiple synchronizations might occur if the SNTP process reaches a high-stratum NTP server before a low-stratum one.

Listing 6: SNTP synchronizations on the Site router

```
00:00:56: %SYS-6-
CLOCKUPDATE: System clock has been updated from 14:09:51 UTC Mon Feb 25 2008
```

```
 to
13:29:59 UTC Mon Feb 25 2008, configured from SNTP by 10.0.0.5.

00:01:59: %SYS-6-
CLOCKUPDATE: System clock has been updated from 13:31:02 UTC Mon Feb 25 2008
 to
13:31:02 UTC Mon Feb 25 2008, configured from SNTP by 10.0.0.10
```

The **show sntp** command can be used to display the current synchronization status (Listing 7).

Listing 7: SNTP status on the Site router

```
Site#show sntp

SNTP server     Stratum   Version    Last Receive

10.0.0.5           7         1        00:00:56

10.0.0.10          6         1        00:01:03    Synced

10.0.0.6           7         1        00:00:15
```

Summary

Having accurate time on the network devices is very important if you use public key infrastructure (where the time is needed to check the certificate validity) or if you want to perform a distributed analysis of a security incident or a routing problem.

Cisco IOS devices (router and switches) can be configured to use the Network Time Protocol (NTP) to synchronize with a reliable time source. They can also act as NTP servers allowing you to build a hierarchical time synchronization infrastructure. Any NTP client can synchronize with a router providing NTP services; you can thus minimize the impact of workstations' time synchronization from remote sites and increase the time accuracy on these sites by providing the time services locally.

Low-end routers that do not support NTP can be configured to use SNTP (a stripped-down version of NTP that provides only the basic client functionality). SNTP client can also be configured on some mid-range routers (even if they support NTP) in the recent IOS releases.

NIL – More Than Just a Training Company

NIL Learning delivers the leading-edge Cisco training to IT professionals and companies around the globe. Through field-proven experts — each both active engineer and instructor — NIL Learning enhances the standard learning curriculum with real-life experience and helps clients to maximize their training investment.

NIL Learning is part of NIL, a leading global IT solutions provider. Since 1992, NIL has been at the forefront of advanced contributors to strategic partner Cisco's technologies, learning curriculum and value-added solutions deployed to clients around the globe. Today, NIL has earned the highest certifications offered by Cisco, VMware, EMC, HP, IBM, Microsoft, F5, Jive, MobileIron, RSA, VCE and others. Their portfolio of solutions consists of managed services, professional services and learning services.

NIL is headquartered in Slovenia, with regional offices in Croatia, Serbia, Saudi Arabia, the U.S., Turkey, South Africa, Morocco, Nigeria, Kenya and Botswana.

Why learn at NIL LEARNING?

- All NIL LEARNING instructors are **field-proven experts** - each both active engineer and instructor.
- **75% of NIL LEARNING engineers hold CCSI** certifica- tions, and **18 have already achieved the respected CCIE rank.**
- NIL LEARNING **enhances the standard learning curriculum** with real-life experience and helps clients to maximize their training investment.
- NIL has been a Cisco Training Partner for many years; it became a **Cisco Learning Partner in 1993**, and has been a **Cisco Gold Partner since 1995.**
- NIL was awarded the **Cisco Most Business Relevant Learning Partner in MEA in 2010 and the most innova- tive learning partner in MEA.**

- NIL received the **Innovation Award for its Technology Led Training and its extensive contribution to Cisco learning solutions** at the Cisco EMEAR Learning Partner Summit in 2012.
- NIL received the **Innovation Award for its Technology Led Training and Advanced Engineer Program** at the Cisco Global Learning Partner Summit in 2013.
- NIL LEARNING runs a **centralized training schedule across the whole EMEAR region.**

More Info

Slovenia

T: +386 1 4746 500

E: sales-support@nil.com

Botswana

T: +267 318 1684

E: training@it-iq.bw

Croatia

T: +385 (0)51 583 255

E: info-nilcroatia@nil.com

Kenya

T: +27 (0)11 575 4637

E: mea_sales@nil.com

Morocco

T: +212(0) 660 808 394

E: info-nilmorocco@nil.com

Nigeria

T: +27 (0)11 575 4637

E: mea_sales@nil.com

Saudi Arabia

T: +966 1 465 4641

E: info.nilme@nil.com

Serbia

T: +381 11 2282 818

E: info-nilserbia@nil.co.rs

South Africa

T: +27 (0)11 575 4637

E: mea_sales@nil.com

Turkey

T: +902 123 81 8639

E: info-nilturkey@nil.com

USA

T: +1 612 886 3900

E: info-nilusa@nil.com

www.learning.nil.com