

Cost Effective Network Fault Tolerance

By Ed Hanzel, System Specialist, Oceanside Unified School District and Marc Richardson, System Technician, Oceanside Unified School District



Ed Hanzel



Marc Richardson

Oceanside USD identified two areas where fault tolerance would be affordable and practical, those being firewall and Web server. Fault tolerance and cost and throughput were the primary reasons for selecting the SonicWall 300 (www.sonicwall.com) as our firewall.

Initial setup of the SonicWall in our system involved setting up static routes for all of our schools and the district office. Since we also run an R2000 for content filtering, we had to do the same for both in order to get them to play well in the same sandbox. Once that was accomplished, few, if any, problems have occurred.

The SonicWall has three ports, LAN, WAN and DMZ. To configure failover, the two SonicWall devices are simply connected together using a switch and the two DMZ ports. This also allows for adding a proxy filter in the transparent mode without re-addressing all of the machines in the district. There is a tab or button, in the main SonicWall screen that addresses the High Availability mode. This is where configuration of the failover begins. The second, or backup, device should be blank, except for having the same firmware as the primary. You need a block of ip addresses, as you must assign another address at this point, different from the one that is used for general networking. But, communication between the two devices is obtained by entering the serial number of the second device in the block provided, then requesting that the two devices synchronize.

The two devices use a "heartbeat" signal to communicate with one another. As such, a heartbeat interval and failover trigger level must be set. The heartbeat interval is the time, in seconds, between heartbeats, and the failover trigger level is the number of heartbeats that, if missed, will cause a failover condition. Should power fail

to the primary unit, there is a standard 10-second failover. Once the primary firewall is configured, turn on the secondary one. It should be detected and configured by the primary, and the log file will reflect the status.

The Web server is a different story. Once again, the name of the program is "Heartbeat," and while the end result is the same, everything in the middle is different and requires some Linux skills. Like most Linux software, Heartbeat is free. While that's the good news, it is also the bad news, as support and training are minimal. For the full setup, it takes some tinkering skills. The program is available from a Linux High Availability Web site, <http://www.linux-ha.org/>.

One of the articles referenced on the site, "High Availability Systems Under Linux," by Atif Ghaffar, is a very humorous approach to the subject and worth reading (<http://www.linuxfocus.org/English/November2000/article179.shtml>). In it, Atif points out that money can be saved by

using a "fully loaded" machine for the primary, and a "just enough" machine (CPU processor speed and memory) for the backup. The theory here is that the backup machine will not be receiving changes and have many people logging into it for uploads. It is strictly a machine that will continue to put your presence on the Web, while you fix the problems in the main server. This will become evident later, when synchronization techniques are discussed.

While Heartbeat is available for Windows servers, this article focuses on Linux and Apache. To start with, both the master and backup must be configured to work with two network cards in each. These second cards can be configured with private addresses outside of the normal ip range you use on your network. For example, if you use a class A range (10.0.0.0) for your main IP addressing, consider using class C here (192.168.0.0). The purpose of the second card is to talk to the other machine only. A simple

COMPUTER	USE	EXAMPLE
Webserver.1(master)	Maintenance IP Address	10.99.1.5
Webserver.2(backup)	Maintenance IP Address	10.99.1.6
Webserver.1	Communications IP Address	192.168.0.1
Webserver.2	Communications IP Address	192.168.0.2
Both	Alias IP Address Main DNS listing for webserver	10.99.1.100

Table of IP Address Requirements

crossover cable provides connectivity. Also use a backup null-modem cable between the two serial ports of the machines. This provides additional backup against hardware failure.

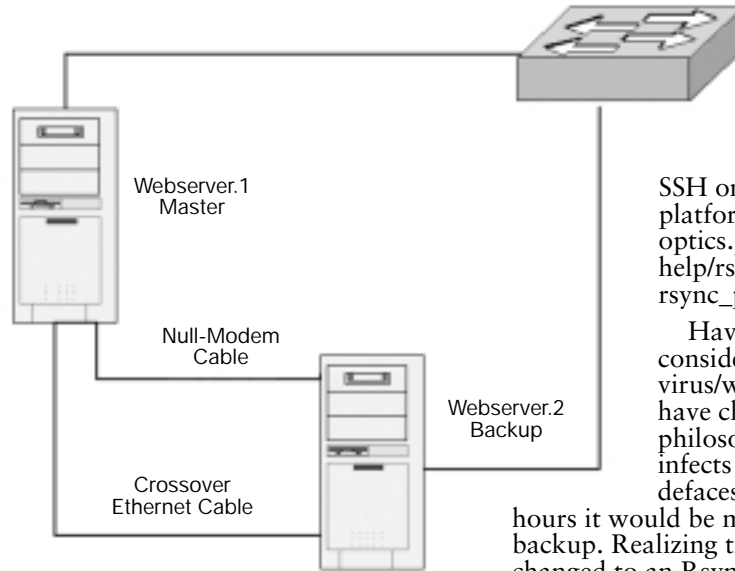
Once this is working, it's time to install Heartbeat. There is an excellent guide available on the site at <http://www.linux-ha.org/download/GettingStarted.html>. The guide includes excellent examples in how to locate and use the configuration files and what all of the settings mean in plain English. These files or scripts, allow naming of the primary and backup servers, the Alias IP address, and are set with defaults for failover times, heartbeat intervals, and so fourth. It is here that you will also name the service that will be started — in this case it's Apache. The trick here is that you must remember to let Heartbeat start and stop the service, and take the service out of auto-startup. Heartbeat will also send an ARPCLEAR signal, which will cause the switches and routers to re-home on the MAC address of the device holding the alias ip address. Anyone that has attempted to exchange boxes on the fly in a live network understands the beauty of this. Until the arpcache in the various switches and routers are cleared and rebuilt, they attempt to send packets to the old device. This happens because all these devices work on layer 2, using MAC addresses.

The chain of events, then, goes something like this:

- System startup
- Recognize Heartbeat
- Recognize position on the network (master or backup)
- Activation of Alias IP
- Send ARPCLEAR
- Initialize Apache (or other service)

In testing, you can ping both the hard coded ip address and the alias. Ensure that shutting down the master causes the backup to capture the alias ip, and send a CLEARARP. You should be able to display the arpcache in your router to determine if the alias ip is, in fact, pointing to a new MAC address. Then, when restarting the master, it should recognize that it is the master, grab the alias ip, and send a CLEARARP again.

So, now that the failover works, the task is to keep them current. We use Rsync, another free Linux-based software package. A good place to start on this software is the author's site at <http://sunsite.dk/info/guides/rsync/rsync-mirroring.html>. He points out



SSH on a windows platform (http://optics.ph.unimelb.edu.au/help/rsync/rsync_pc1.html).

Having given a lot of consideration to the latest virus/worms/Trojans, we have changed our backup philosophy. If something infects the Web site, or defaces it, within two

hours it would be mirrored to the backup. Realizing this, we have changed to an Rsync schedule of every three days. True, the latest changes may not be included, but with all of our school sites, it would give someone an opportunity to notice a difficult, and perhaps repair it, before it is mirrored to the backup server.

where to download it and how to configure it. We started by setting up a differential backup to occur every two hours. This ensured that the backup server would always mirror the master, with very few exceptions. We also configured it to copy only the Web site portion, not users/passwords/groups. It is not intended that the backup be used for uploads of new information. If the master server is down, users will have to wait for updates until the master can be brought up again. We have also set up SSH for the Rsync, but only because we are paranoid. This site addresses how to install Rsync and

These two examples demonstrate what we have done so far with high availability redundant systems for fault tolerance. The same principles could be applied to any service, which runs on a server, such as email, DNS, or critical databases. It took us a while to navigate these waters, but now that we know how, we will be adding other services. The biggest cost is the learning curve. ■

Financial administration - Personnel - Payroll - Kern COE - Demographics - Budgeting - Web access - Customized training - Position Control - individual implementation plans - El Dorado COE - Purchasing - Requisitions - Personal account managers - Salary schedules - Capistrano Unified - Customized reporting - Numeric grades - Sacramento COE - Hourly attendance - Fixed assets - Corona-Norco Unified - Student discipline - SACS compliant - Human resources - Student scheduling - Placer COE - Budget development - On-site training - Y2K ready - California State Office - San Diego Unified - San Diego Unified - San Diego Unified - Payroll - San Juan Unified - Yolo COE - Lucia Mar Unified - web access - customized training - Student records - individual implementation plans - letter grades - Student discipline - Marin COE - 24 hour emergency support - Salary schedules - Customized reporting - SACS compliant - Human resources - Fixed assets - Budget development - San Luis Obispo COE - Large user group - Purchasing requisitions - SACS compliant - Human resources - Positions control - Financial administration - Personnel - Payroll - Kern COE - Demographics - Budgeting - Web access - Customized training - Position Control - individual implementation plans - El Dorado COE - Purchasing - Requisitions - Personal account managers - Salary schedules - Capistrano Unified - Customized reporting - Numeric grades - Fixed assets - Corona-Norco Unified - Student discipline - SACS compliant - Human resources - Student

QSS/OASIS
Software for California's Schools

Finance/HR/Payroll
Student Records

21 COEs, 45+ Independent districts

www.qss.com 650/372-0200