



BY GUY C. YOST

Streamlining Management of WinFrame Servers: Part III

This concluding article explores how to streamline the creation of WinFrame servers for initial roll-out, maintenance or disaster recovery and introduces the logic for incorporating a WinFrame user bulletin system.

LAST month, Part II of this series looked at automating routine maintenance tasks on WinFrame and NetWare servers. We examined four major tasks:

- ◆ using WINNAT on NT/WinFrame and CRON on NetWare platforms
- ◆ automating unattended reboot process for NT and NetWare Servers
- ◆ automating unattended print spool purge across multiple servers
- ◆ logging client log in and log out events for analyzing access patterns

This concluding article will explore how to streamline the creation of WinFrame servers for initial roll-out, maintenance or disaster recovery. The logic for incorporating a WinFrame user bulletin system will also be introduced. The strategies presented can also be applied to the Citrix MetaFrame environment, which uses NT 4 as the base OS instead of NT 3.51.

WINFRAME SERVER CREATION AND RECOVERY

Administrators of large-scale multiple-server WinFrame installations will be faced with installing and rolling out multiple WinFrame servers quickly as well as keeping the disk “image” on each of the servers current and up-to-date with each other. Especially where load balancing is used, server disk contents will need to be identical to provide consistent and reliable sessions across the server farm. As discussed in Part I, it greatly reduces the amount of work needed to manage multiple “drone” servers if you avoid distributing application and data files across servers, and you centralize file resources as you would in a classic client/server environment. This still leaves

the ongoing task of keeping all the drone images (used as multi-user clients rather than servers) synchronized, however. Administrators will also want a way to quickly restore or rebuild a server in the event that a disk system or other critical server component fails or the OS becomes corrupt or unusable.

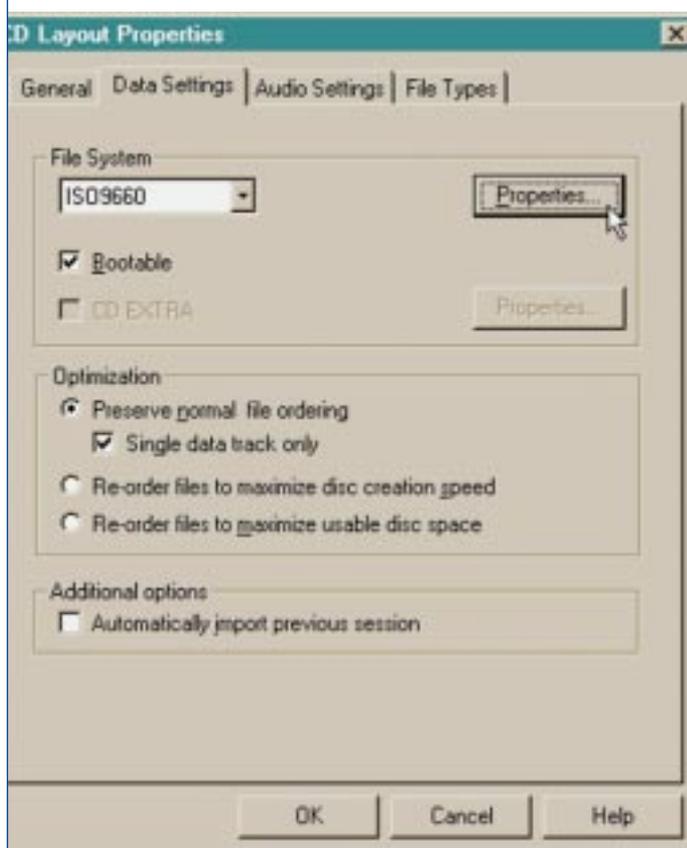
In Part I, a simple batch routine to synchronize drone files across servers was introduced. The suggested technique is ideal for distributing moderate amounts of data, but it is not designed for complete server disk image reproduction. I’ve worked with a few third-party products that are designed to synchronize a single “master” server to one or few “secondary” servers; however, these products are not designed to synchronize large numbers of servers due to the network and CPU overhead associated with constant “real-time” updates.

There are a few ways to provide solutions for each of these issues; however, I’ll examine a single technique that that can be used to accomplish all of them. The idea is to use a homemade bootable CD that contains a “master” server disk image and startup scripts to semi-automate the process of rapidly creating servers for initial server builds, disaster recovery and image synchronization.

First, I’ll outline the steps for building the CD:

1. Configure a master WinFrame server and then create a master server drive image using a disk imaging product such as Ghost from Ghost Binary Research or Drive Image Pro from PowerQuest Corporation. Functional evaluations of these products can be downloaded from www.ghost.com

Figure 1: Making a Bootable CD



and www.powerquest.com, respectively. You will need to think about how to prepare a very generic and compact image that can be easily customized once installed. You'll want to keep the image itself under 600MB to fit on a 650MB recordable CD and have room for other utilities. Here are some tips on preparing a master drone for efficient imaging:

- ◆ Reduce the swap file to only 25MB to make the server image smaller – it can be increased to normal size once installed on a server.
 - ◆ Only include disk partitions that contain important files — partitions used for swap, temp or generated files can easily be built using Disk Manager once the base OS image is in place.
 - ◆ Remove server-specific licensing just before imaging so you don't propagate the same license across several servers.
 - ◆ Use the compression options that come with the imaging software. Both products mentioned earlier support "fast" and "high" compression options. We found the "fast" option to be sufficient and the "high" option to be much slower but with not much space gain.
2. After creating a generic WinFrame server image, copy the disk image file to a file server and then copy it back down to the hard drive on a workstation that has a recordable CD drive and corresponding CD creation software. Using the CD creation

Figure 2: CONFIG.SYS

```
[MENU]
menuitem ght, Restore disk from Ghost image on CD
menuitem nwk, Connect to network

[ght]
[nwk]

[COMMON]
device = a:\dos\aspi8dos.sys /d
device = a:\dos\aspicd.sys /d:cdrom01
buffers = 30
files = 30
lastdrive = z
```

software, specify inclusion of the master image file to be written to the CD. We purchased a HP Surewriter that came with Adaptec's popular Easy CD Creator software. Current releases of this software support creating bootable CDs using the ISO 9660 standard. (You can obtain the software and its updates from www.adaptec.com/support/files/upgrades.html.) From the program's main File menu choose CD Layout Properties, then click on the Data Settings tab. Configure the CD layout to use the settings shown in Figure 1.

3. You will be asked to provide a floppy disk containing a bootable DOS image. When bootable CDs are created, the system makes the bootable DOS portion of the CD (only using 1.44MB) the A: drive, and the rest of the CD capacity will be accessed using whatever drive letter you specify in the MSCDEX command. For example, we created a bootable DOS 6.22 floppy containing the CONFIG.SYS and AUTOEXEC.BAT files, as shown in Figures 2 and 3.

Note that the CD partition containing the server image (master.img) is the K: drive, as specified by the MSCDEX command. The syntax used in this example is specific to the Ghost software product, specifying the size of each logical partition. We included an option to attach to the file server using a standard 16-bit VLM client so we could pull a server image down from the network if the image on the CD became outdated. The example implies the existence of DOS and GHOST5 directories on the floppy. The DOS directory contains the SCSI device drivers specified in the CONFIG.SYS as well as the MSCDEX.EXE command. The GHOST5 directory contains the GHOST.EXE software.

Using this approach, we can walk up to a completely scratch server, drop in the bootable CD, choose a menu option from the CONFIG.SYS to load the server image from the CD or network, and GHOST loads using our predefined parameters. After about 15 minutes, we have a newly installed WinFrame server, identical to the generic master server.

Your servers can't remain completely identical, however, so there are some additional tasks required to identify each server. After creating the base image, boot the server off of the network so its name, IP address, license (if any) and SID won't conflict with the master or any other server on the network. Once the server is booted, log in as Administrator and do the following:

1. Run the SID editor utility to uniquely identify the new server. Each NT installation uses an inherent System ID to uniquely

Figure 3: AUTOEXEC.BAT

```
@echo off
prompt $p$g
a:\dos\mscdex /d:cdrom01 /m:12 /l:k
goto %config%

:ght
a:\ghost5\ghost.exe -clone, mode=load, src=k:\master.img, dst=1, sze1=502m,
sze2=2047m, sze3=2047m
goto end

:nwk
cd\nwclient
@call startnet
f:
login drone
map root k:=fs01\data:images\wfmaster
a:\ghost5\ghost.exe -clone, mode=load, src=k:\master.img, dst=1, sze1=502m,
sze2=2047m, sze3=2047m

:end
```

Figure 4: Directory Structure

```
DATA:
    REGISTRY
        WFSERVER01
        WFSERVER02
        WFSERVER03
        ...etc...
```

identify itself. A freeware utility called NewSID from www.sysinternals.com will automatically parse through system files and apply a unique system ID. As a convenience, be sure to include this utility as part of your master image.

2. Activate unique base and load balance licenses using the WinFrame Licensing utility in the Administration program group.
3. In Control Panel, redefine the SWAP file to a normal (larger) size using the System applet.
4. While still in Control Panel, rename the server and set a unique IP Address (if applicable) using the Network applet. You may need to “reset” the NIC driver so it sees the network adapter. Simply go into the adapter configuration, verify settings, exit the Network applet, and then exit Control panel.
5. Reboot the server as prompted. When the newly configured server reboots, you’ll have a uniquely identified production-ready server.

Once the server is configured for production, make a backup copy of its registry using the REGBACK utility found in the NT

resource kit in the \i386\registry\directory. The WinFrame registry holds all the information that uniquely identifies that server (name, IP address, WinFrame user and load balancing licenses). We created a directory structure on our file server that holds registry backups of all our production servers, as shown in Figure 4.

Especially where load balancing is used, server disk contents will need to be identical to provide consistent and reliable sessions across the server farm.

After all of the servers were built, we had room to include a copy of the registries on the master CD. We have since had to format and reinstall a few test servers and the efforts to rebuild a server couldn’t be any easier. We simply installed the base WinFrame OS image as described previously and then ran the NEWSID and REGREST commands (also found in the NT resource kit in the \i386\registry\ directory) to finish the process.

We demonstrated this simple rebuild process to an IT executive management team who was thrilled with the 15-minute demonstration as proof of a solid and easy disaster recovery strategy.

CLIENT BULLETIN SYSTEM

The examples used in this series are based on work done for a large WinFrame application. Because this installation services thousands of users all across the state, it became necessary to devise a way to easily communicate technical bulletins or daily informational “news” flashes that let users know of maintenance dates, expected down times, partial functionality, events, training dates, etc. We examined messaging options and came up with a simple yet functional approach using (yes, again) batch files.

The need dictated a system that supports multiple bulletins displayed in a queued fashion. That is, if there are 10 active bulletins, users would view each bulletin document only once in succession, and the system must verify when a user has read each bulletin. Optionally, specific bulletins can be treated as “constant reminders” and will display every time a user logs in until turned off.

Users receive bulletins as part of the login process. After the user ID and password is entered, the login script executes and at the end calls a batch file that includes the code block shown in Figure 5. The bulletins themselves are stored in a central directory (SYS:\PUBLIC\BULLETIN) on the NetWare file server with the file names BULLET.001, BULLET.002, etc. This naming convention differentiates and saves the bulletins for historical review. To check if a user has received a bulletin, a file is created on the K: drive (mapped in Figure 5) using his login id and bulletin number as the file name and ext, respectively. All check files are stored in the same directory off of the server’s DATA volume called BULLCHK. If a bulletin is to be viewed persistently, then a file with the name “const” and the bulletin version as the extension is created in the BULLCHK directory. The BULLDISP.BAT is shown in Figure 6. Optionally, you could include a statement to print out the bulletins after the “PAUSE” statement by simply printing the file BULLET.%2 to the local LPT port for the user’s records.

Figure 5: Bulletin Display Main Batch Code

```

REM This next section checks to see if there is an active bulletin.
REM Each user should see a bulletin one time, and not on subsequent logins
REM unless the bulletin is indicated as a 'constant' reminder.
REM the P: drive is mapped to the PUBLIC directory on the file server in
REM the LOGIN SCRIPT. Also, %1 is set to the NDS user ID by the LOGIN SCRIPT
Set BULLETSET="01 02 03 04 05 06 07 08 09 10"
map root k:=nwfs01\data:bullchk>nul
for %%i in (%BULLETSET%) do call bulldisp.bat %1 %%i
map del K:>>%LOGFILE%

```

Figure 6: BULLDISP.BAT

```

REM First see if user has already read the bulletin being called by the main code
REM If so, skip this bulletin and exit this batch file
IF EXIST K:\bullchk\%1.%2 GOTO END
REM Else, display the bulletin not yet seen by this user
CLS
TYPE P:\bulletin\bullet.%2
REM Append information about this session to the on screen message
echo *****
echo Winframe Client ID: %CLIENTNAME%, User Login ID: %1 , Bulletin Number: %2
echo *****
PAUSE
REM Now create a little file using the userid.bulletin# in the K drive.
REM to indicate that the user has seen the bulletin. Don't create the file
REM if this bulletin is a "constant" reminder. A repeated bulletin will have a
REM corresponding blank file called "const.bulletin#" in the bullchk directory
if not exist k:\bullchk\const.%2 echo %1 got bulletin number:%2>k:\bullchk\%1.%2
:END

```

SUMMARY

This three-part series highlighted some of the most effective methods and tools that allow management of multiple WinFrame servers with the least amount of effort. Of course, there are many more tricks and behind-the-scene processes that will further simplify WinFrame server-farm management, so if more of this type of article series is of interest to you, please let Editor Amy Novotny know via email at editor@naspa.net. 



NaSPA member Guy C. Yost is the owner of Redstone Consulting, an IT management consulting firm in New York. He has authored several books on networking

for Que Publishing, including Learning NetWare 4.1, and NetWare 4.1 SmartStart, and contributes to Technical Support magazine as an author, columnist and technical editor. Guy also develops and conducts seminars on networking with Windows NT, UNIX, NetWare and Internet/intranet technologies across the United States and Canada. He can be reached at (518) 674-5606 or gyost@logical.net.

©1998 Technical Enterprises, Inc. For reprints of this document contact sales@naspa.net.