Optimizing Login Scripts for Remote Users

With more and more remote users accessing the network via cable, Digital Subscriber Line (DSL), and Code Division Multiple Access (CDMA) modems, optimizing login scripts for remote access can be a challenge. Login scripts must allow remote users to log in quickly while still providing the basic services users require to access resources on the network. This article provides a quick review of the different kinds of login scripts available in NetWare 5 and 4, explains how these login scripts interact, and lists some options for handling remote users. This article also touches on remote authentication options since authentication is a key factor in enabling remote users to access a network.

LOGIN SCRIPTS 101

If you usually manage NetWare 3 or 2 networks or single-server, single-container environments, you may need a brief review of NetWare 5 and 4 login scripts. NetWare 5 and 4 have four types of login scripts: container, profile, user, and default.

The container login script is the most commonly used login script. As the name suggests, the container login script sets the network environment for all users in a particular container. This login script executes first.

The profile login script is used primarily for group-specific login requirements. (Of course, the container login script can also be used to handle group-specific login requirements.) This login script executes after the container login script.

The user login script is exactly the same as the user login script included in previous versions of NetWare. The user login script sets the network environment for a particular user. This login script executes after the profile login script.

Of the four types of login scripts, the default is the only login script you cannot change. This login script contains only essential commands, such as drive mappings to important utilities. The default login script runs the first time you log in as the ADMIN user. This script also runs if a user does not have a user login script.

As you can see, you can set up multiple login scripts to run when a user logs in to the network. You can even have more than one container login script execute when users log in.

By default, a user runs the login script for the container in which his or her User object resides. In addition to having the container login script for the user's context execute, you can chain, or jump, to other container login scripts: Simply use the INCLUDE command followed by the context of the login script that you want to process next.

Despite the fact that you can have multiple login scripts execute, I recommend that you keep login scripts as simple as possible. The more login scripts you use, the more time you will have to spend troubleshooting if problems arise. You will also have to spend more time making changes, and you will have to ensure that the changes you make do not impact your complicated login scripts.

The last line in the login script that you set up should end with a NO_DEFAULT statement. This statement prevents the default login script from running and overwriting other login scripts or causing problems.

Although most remote users will probably access the network via high-speed connections such as cable modems and DSL connections, you should plan login scripts for remote users as if they will be using a slow dial-in connection. As a result, you should take a minimalist approach to reduce the amount of time spent processing the login script. For remote users, you should execute only drive mappings in login scripts; you should not try to execute other commands such as the command for the Novell Delivered Applications window. (This window is part of the ZENworks Application Launcher.)

You should not run the ZENworks Application Launcher across a remote connection because loading it across a slow dial-in connection could take several minutes. Not only must you load the executable, but you must also load all of the associated DLLs that that ZENworks Application Launcher requires. In addition, depending on the refresh interval you have set in the ZENworks Application Launcher's properties, more traffic will be generated when NDS eDirectory checks what changes have occurred on each remote user's workstation.

Fortunately, you can use the new remote capabilities in ZENworks for Desktops 3. Using these new capabilities, you can create an application CD and configure ZENworks Application Launcher to run the applications from this CD.
If users have a high-speed remote connection, you can use the ZENworks Force Cache option. ZENworks for Desktops 3 will copy the application to a hidden cache on the remote workstation’s hard drive. The application can then be installed as needed on the remote workstation. (For more information about using these remote capabilities, see “ZENworks for Desktops 3: Tips for Delivering Applications to Remote Users,” Novell Connection, Nov. 2000, pp. 40–41. You can download this article from www.ncmag.com/past.)

If you are using services such the single sign-on service in BorderManager 3.5, you should copy the CLNTRUST.EXE and DWNTRUST.EXE files to the workstation hard drive instead of copying them across a WAN connection. (Do not confuse this single sign-on service with Novell Single Sign-on, which is a separate product.)

The best way to set up login scripts for your company will depend in part on how many Organizational (O) or Organization Unit (OU) containers your company’s NDS tree has. Because most medium- to large-sized companies have multiple O and OU containers, you have probably already experienced the challenge of maintaining identical login scripts for each container. If your company has one O container and multiple OU containers, you can put all of your login script commands in the O login scripts and include the following command in the OU login script:

```
include organization_container_name
```

Replace `organization_container_name` with the name of the O container. This ensures that everyone runs the same login script and reduces the work you must do on login scripts for the OU container.

**HOW TO HANDLE REMOTE USERS**

You can handle remote users’ access to the network in several ways:

- **Option 1.** You can create a second username for each remote user and have remote users enter this username when they log in from a remote location.
- **Option 2.** You can place remote users in a special group (called Remote, for example) that changes how the login script executes when these users log in from a remote location.
- **Option 3.** You can create a special container for remote users and put an alias for each remote user in this container.

**Option 1**

The first option requires quite a bit of setup on the front end because you must create a second username for each remote user and ensure that the second username has the same rights as the user’s regular username. Each remote user can then access the files he or she needs no matter how that user logs in.

However, if your company’s network includes Windows NT and NetWare, this option has additional impact: If you are running the Windows NT login script and the NetWare login script, you must complete some additional setup to ensure that the NT login script continues to work.

Using the User Manager for Domains, you must look at the user’s local login ID and see what login script batch file is configured to run. You must then review this batch file to see if any of the commands require files to be copied to the user’s workstation. You can then determine if you need to copy these files to the users’ workstation before the user logs in remotely. After you make any necessary adjustments, you can associate this login script with the user’s remote username.

After you complete the setup process, remote users must remember what username to use when logging in locally versus remotely. Depending on users’ level of PC knowledge, users may try to log in with the wrong username and end up calling the help desk for assistance.

Handling the login script for this second option requires some planning: Look at your login script and determine the point at which you could stop the processing of the login script and still deliver the drive mappings and services remote users require.

Depending on how your login script has evolved, you may want to do a little housecleaning before trying to implement a remote-access solution. For example, I suggest placing the drive mappings required by all users at the top of the login script and placing anything that requires a group membership below. This initial step helps you get a good idea of what the login script does and how to handle exceptions for particular groups and/or departments.

Whether remote or local, users require one or more MAP commands. Place these essential MAP commands first (even though some drive letters may be mapped out of order from the way the volumes are mounted when the server boots up).

If remote users do not need all of the MAP commands listed in the login script, stopping the other MAP commands from executing may be a simple process: You may be able to include a command, such as the following, after the last drive mapping remote users do need:

```
"IF MEMBER OF ".REMOTE_CONTEXT" THEN EXIT"
```

If a remote user requires one or more drive mappings that are unique to a particular department, you can change the THEN EXIT statement in this command to THEN BEGIN. You can then add each of the commands on a separate line below. After you list the commands, you can add a new line and type ENDIF.

Ideally, you should be able to create just one remote group. Creating more groups will obviously create more administrative work for you.

**Option 3**

The third option involves the most initial setup. This option also requires that users know when and how to change their context as they log in to the network. Users also have to remember more than one password. From a security standpoint, you may want to require users to use a different password when logging in using the remote-access account, or you may want to implement Novell Modular Authentication Service (NMAS). If you implement
NMAS, you can assign the user a hardware token that handles the password process, making the login more secure.

By using a different context for remote users, you can create a unique login script for remote users. Because this login script is customized for remote users, it can be a basic login script. You can do this by using the IF MEMBER OF “Group_name” statement to embed the commands unique to remote users or to stop processing the commands not needed by remote users.

USE A VPN

Regardless of which option you choose, I strongly recommend that you use a Virtual Private Network (VPN) solution for remote users. This VPN solution should support Lightweight Directory Access Protocol (LDAP) or Remote Authentication Dial-In User Service (RADIUS). If your VPN solution supports LDAP or RADIUS, you will not have to maintain yet another authentication method. In addition, if a user leaves the company or no longer needs remote access, you will not have to delete the user from yet another database.

If you can find an Internet service provider (ISP) that will help you implement RADIUS, this solution has one advantage over LDAP. With RADIUS, you have more control: When a user dials in to an ISP, the ISP can forward all login requests to you before allowing the user to connect to the Internet. As a result, you do not have to keep a list of active user accounts at the ISP, and you do not have to assign each user a separate ISP password. In addition, you can quickly terminate a user’s access if that user leaves the company.

ADDITIONAL TIPS

Designing login scripts for remote-user access is not difficult. You can minimize the administrative work by using the following suggestions:

Give Me Some Space

Keep login scripts simple and neat. Do not bunch all the lines together. For example, grouping together all of the lines that map drive letters to the root of a volume is a good idea; however, you should leave at least one blank line between sections that are unique to a group. You can then follow the logic of what is happening within each section of the login script.

Depending on the number of lines executed in a particular section of a login script, you may want to leave some space in that section to make it easier to read. For example, if a section starts with a then-begin statement and ends several lines later with an end statement, you may want to leave two or three lines after the end statement. These blank lines won’t increase the processing time, but they will make the login script easier to read.

Know Your Audience

Determine if remote access is practical for users. For example, I recently helped a large company implement a remote-access solution and quickly found the Accounting department would not be happy with remote access. These employees had to access large spreadsheets (more than 1 MB in size) that included references to other spreadsheets. As a result, just loading the files took 15 to 20 minutes at a minimum.

Get Users’ Attention

Use the WRITE command in the login script to remind users that they are using the remote-access version of their user account. You can also include a short explanation of the differences users should expect when they use their remote-access username. This short explanation may prevent a call or two to the help desk.

To make the WRITE statement effective, you should enable the option to display the results in a window on a user’s screen. You should also use a PAUSE command to keep the message displayed until they press a key to continue executing the login script.

Depending on how often you change this message, you can use the DISPLAY or FDISPLAY command to print a text file in the Login Script Results window that appears on the client workstation when the user is logging in. You can then have a user edit the file (rather than you having to edit the login script). You can assign any user to edit the file, eliminating one more administrative task from your list.

Because some users may not be paying attention to their workstation while the login script is running, you may need to use the fire phaser command to get their attention. Of course, you should not use the fire phaser command with a count of 10 or 15 because executing the command may be slow on a dial-up connection.

With NetWare 5, you can use the fire phaser command with a slight variation. By using a command such as fire 5 sample.wav, you can get users’ attention by launching a special WAV file that sounds unlike anything else on their workstation.

Keep It Simple

Be careful when you use # or @ with external commands in the login script. As mentioned throughout this article, you should not run programs across slow dial-up connections. This restriction includes what can seem like small applications that users run as a part of the login script when users log in locally. (For example, you have to use the CLNTRUSTEXE file if you are using the single sign-on function in BorderManager 3.5.) If users must use small applications such as these, copy them to the hard drive of every workstation.

Double-check login scripts for users who access multiple platforms. If you manage multiple platforms, you are probably already familiar with the problems of handling dual login scripts. In some cases, you will need to evaluate what each user actually needs for remote access.

For example, because the Windows NT login script process is a batch file process, most of the concepts that have been discussed on the Novell side should work, although some minor changes may be required. To copy an icon on the desktop for the remote access may be easier. Users can then simply double-click the icon when they need to access the resource.

Document Your Changes

While you set up and change login scripts, you should use REM, *, or ; to include comments in the login script. These comments can help you remember what a particular piece of the login script does and why it is there.

The more changes you make to a login script to accommodate a particular user, group, or application (also known as management by exception), the more important documenting those changes becomes. Knowing why a particular line is in the login script may save you grief later when you are trying to figure out if you need a particular section of the login script or not.

CONCLUSION

Working with login scripts on your network is not a one-time process. The only thing constant about networks is that they change. Because you will need to make changes, keeping things simple and documenting what you do and why will prevent problems down the road.

Ronald Nutter is a Master CNE and a Microsoft Certified Systems Engineer. •