

## **Appendix A Advanced Utilities**

---

Chapter 8 described the monitoring features available for network users. This appendix explains how to use SiteMeter's advanced utilities.

## About SiteMeter's Utilities

---

The utilities described in this chapter are provided for you if you have decided not to use the SiteMeter Proxy NLM as the metering and file protection method for your LAN.

### Utilities List

The following lists the utilities described in this chapter:

- ☐ SWATCHER
- ☐ DSW
- ☐ SYSMOD
- ☐ SUNLOCK
- ☐ SLOCK

Each of these utilities is fully explained in the following sections.

## Swatcher TSR Method

---

If you have chosen to use the Swatcher TSR method of metering and file protecting your LAN, you may need to become familiar with the following metering utilities:

- ☐ Swatcher
- ☐ DSW

Swatcher is a TSR and must be loaded in order to properly meter and protect the files on your LAN.

### Loading Swatcher

Swatcher must be loaded from the DOS prompt. To load Swatcher type:

```
Swatcher <ENTER>
```

Swatcher must be loaded AFTER loading IPX and NETX shells. It should not be loaded high; it should be in conventional memory.

We recommend loading Swatcher from the AUTOEXEC.BAT before logging in to the network.

**NOTE:**

*DO NOT LOAD Swatcher.com via the NetWare login scripts (either personal or system), as inconsistent and/or inappropriate metering behavior may result.*

*If you try to load Swatcher after login and the Security Scan Interval is set and you have a lengthy login script, you may be disconnected before you get a chance to load Swatcher.*

### Note About Swatcher

Since membership in a group determines local drive access, SiteMeter must know who you are when you load Swatcher. This can only be accomplished if Swatcher is loaded AFTER you log in. For security reasons, however, you may prefer that Swatcher is loaded BEFORE login time. To address this, we have created DSW.COM.

Users can load Swatcher before logging in, but at this time SiteMeter cannot identify the users and therefore will not restrict access to local drives.

ONCE a user is logged in, you must run DSW.COM; it reads the CURRENT server's information about how it should handle the user's local drives and then changes Swatcher accordingly. DSW is only able to update Swatcher ONE TIME-- the FIRST time it is run.

Further attempts to run DSW do not change Swatcher, although you receive a message on the screen indicating that it has updated Swatcher.

### Running DSW

DSW is run through the system Login Script. Here is a sample login script:

```
MAP F:=FS/SYS:LOGIN
DRIVE F:
#DSW
```

### What DSW Does

DSW reads the information the server holds about a user, including whether to disable his or her local drives completely, disable only his or her local .EXE and .COM files, or give the him or her FULL rights.

This information is sent to Swatcher.

Swatcher receives it and updates its values accordingly.

### Potential DSW Problems

If more than one person uses the same PC and these people have different local drive rights, a problem could arise.

*SCENARIO 1:* Person A (in local drives disabled group) loads Swatcher, logs in, and Swatcher is updated.

Person B (who has full rights) logs in to the same PC that Person A was using.

Because Swatcher is NOT updated again, Person B does not get access to the drives.

Person B must reboot and reload Swatcher to get the appropriate rights.

*SCENARIO 2:* Person A (with Full Rights) logs out.

Person B (with no rights) logs into the same PC and now has full rights to the local drives. To prevent this problem, the PC should be re-booted between users.

## SYSMOD

### What is SYSMOD?

SYSMOD is a McAfee utility designed to help network administrators edit files. With SYSMOD, you can edit users' files without going from workstation to workstation to do so.

This utility is installed in the directories SITEMETR or BWORKS and PUBLIC. You can use SYSMOD if you prefer to edit your users' files manually.

### Using SYSMOD

Use the following procedure to implement the SYSMOD program:

**Make sure that the path to the .INI files is included as either a search drive in the case of users' Windows residing on the network or in the users' path statements located in their AUTOEXEC.BAT files for local Windows on the C: drive.**

**Example lines for login script:**

```
Map ins sl6:=SYS:\USER\%LOGIN_NAME\WINDOWS
```

```
INCLUDE SYS:PUBLIC\SMRUSER.BAT
```

**Example line for the SMRUSER.BAT using the INCLUDE command:**

```
SYSMOD WIN.INI REPLACEKEY LOAD SWATCHER.COM  
SWATCHER.COM
```

### Functionality

This section documents the SYSMOD program and its functionality.

All SYSMOD functions follow a basic format:

```
SYSMOD <filename.ext>functionName parameter1...parameterN
```

Below is a list of functions and their associated parameters. An example is also provided for each function. Note that when a parameter can be Null, it must still be entered, but use the NULL keyword. For example,

```
SYSMOD CONFIG.SYS ADDDEVICE C:\HIMEM.SYS NULL BEFORE
```

This would place the device= line at the beginning of the file. (See ADDDEVICE for details.)

The filename specified must exist; SYSMOD, however, will intelligently determine the location of the file for you. If you specify AUTOEXEC.BAT or CONFIG.SYS (with no path), SYSMOD will automatically look on your boot drive for the file. If you specify a filename with no path (e.g., SYSTEM.INI), SYSMOD will look in your path for the file. If you specify a full path to the file, SYSMOD will look in the explicit path for the file. If the file cannot be found, SYSMOD aborts with an error.

### **ADDDEVICE [DeviceToAdd][Key][Option]**

This function adds a new DEVICE= line to a system file (typically the DOS CONFIG.SYS). Specify the path and driver name in DeviceToAdd (e.g., C:\WINDOWS\EMM386.EXE). Specify a key value to search for in Key (e.g., HIMEM.SYS), and where to place DeviceToAdd BEFORE or AFTER Key in Option. For example:

```
SYSMOD CONFIG.SYS ADDDEVICE C:\WINDOWS\EMM386.EXE
HIMEMSYS AFTER
```

would place a “device=c:\windows\emm386.exe” after the “device=himem.sys” line in the CONFIG.SYS. If Key is NULL or the key specified is not found, the device line is added at the beginning or end of the file.

### **ADDLINE [LineToAdd][Key][Option]**

This function adds a line of text to a system file. Specify the entire line of text you wish to add in LineToAdd. Specify a reference key value in Key that you would like to position LineToAdd in reference to. Option (BEFORE or AFTER) specifies whether LineToAdd is placed BEFORE or AFTER Key. Again, if Key is NULL, then LineToAdd is placed at the beginning or the end of the file. For example,

```
SYSMOD AUTOEXEC.BAT ADDLINE C:\DOS\SMARTDRV NETX AFTER
```

would place a new line (c:\dos\smartdrv) after the line specifying netx (if found, otherwise it would go at the end of the file).

**ADDPATH [PathKey][DirectoryKey][Option]**

This function adds a sub directory to a path environment variable. Specify the name of the path environment variable to edit (PATH for DOS, or DPATH for OS/2) in PathKey and the subdirectory to add in PathToAdd. DirectoryKey specifies the sub directory that PathToAdd will be placed BEFORE or AFTER. Option is either BEFORE or AFTER. If DirectoryKey is a null string, ADDPATH will place PathToAdd at the beginning or end (respectively) of the path statement. If the key specified in PathKey is not found, then a new one is added, with a "SET" prepended. This allows for adding path like environment variables such as "SET TEMP=," and so on.

**NOTE:**

*This function can also be used to edit other lines, such as a TEMP environment variable, or any other line that does something like "SET envvar>=d:\path." For example,*

```
SYSMOD AUTOEXEC.BAT PATH C:\WINDOWS C:\DOS BEFORE
```

*would place c:\windows in the path statement before c:\dos. If c:\dos isn't in the path, then c:\windows would be placed at the beginning of the path statement.*

**CFGSETSTRONG [Key][NewValue]**

This function sets a strong variable in a system file (e.g., STACKS 9,256, DOS=HIGH etc.). Specify the variable in Key and a string value to set in NewValue. For example,

```
SYSMOD CONFIG.SYS CFGSETSTRING STACKS 9,256
```

would set the value of the STACKS statement to 9,256.

**REPLACEKEY [Key][DelKey][NewItem]**

This function works like REPLACELINE, except that it replaces a key value rather than the entire line. Use Key to locate the line, DelKey is the key value to replace, and NewItem is the new value. The new value can be NULL, in which case DelKey will be removed. For instance:

```
SYSMOD WIN.INI REPLACEKEY LOAD SWINAPP.EXE SWATCHER.COM
```

would replace the swinapp.exe reference in the load=line of the WIN.INI with swatcher.com.

### **REPLACELINE [Key][NewLine]**

This function replaces an existing line in a system file with a new line. Specify the key value of the line you wish to replace, such as PATH or COMSPEC (or DEVICE, for that matter) in Key. Specify the new value of the entire line in NewLine. If Parameter 2 is a null string, then the line will be deleted. For instance:

```
SYSMOD AUTOEXEC.BAT REPLACELINE COMSPEC NULL
```

would delete the existing COMSPEC line.

### **WRITEINISTR [SectionName][KeyName][NewValue]**

This function mimics the Windows API function WritePrivateProfileString0. It expects the file you're modifying to be in the Windows INI file format. This function writes NewValue to the INI file under SectionName in KeyName. However, if SectionName doesn't exist, then a whole new section is added to the end of the INI file, with a new key=value added in that section. If the section was found, but the KeyName wasn't found, the new key value is added directly after SectionName. For instance:

```
SYSMOD SYSTEM.INI BOOT KEYBOARD.DRV BDIKBD.DRV
```

would change the existing keyboard.drv= entry in the [boot] section of the system.ini to read keyboard.drv-bdikbd.drv.

### **Halting SYSMOD Updates**

This section describes how to stop SYSMOD from updating a user's files every time he or she logs in.

To instruct SYSMOD to update a user's .ini file only once, add similar lines to the system login script:

```
MAP F:=SYS:USER\%LOGIN_NAME
```

```
#COMMAND /C SITE.BAT
```



```
MAP DEL F:
```

where site.bat is a .bat file with the following lines:

```
if exist f:sitedone.doc goto end

call smruser.bat

copy sitedone.doc f:

:end
```

where sitedone.doc is a dummy text file that = 'sysmod has already run.'

This code creates a flag that indicates if sysmod has already run. SYSMOD will see this and not run.

## SLOCK/SUNLOCK

This section explains how to use SiteMeter's SLOCK/SUNLOCK utilities for metering from a batch file. In this section, "LockSet" is a McAfee term for a metered application.

**NOTE:**

*You must have Read, Write, Open, Create, Delete and Search rights in the directory where the SITEDATA file is stored if you wish to use SLOCK and SUNLOCK. The default home directory is SYS:SYSTEM\SITEMETR.*

SLOCK and SUNLOCK allow a user to secure or lock a position in the desired application and then free or unlock that position for another user. These commands should be included in batch files or other means such as a menu system setup.

The following batch file consists of the appropriate SiteMeter commands to lock and unlock positions into the LockSet SYSCON. The batch file looks like this:

```
SLOCK SYSCON* (where SYSCON is metered APP.Name)
```

```
If errorlevel = 100 SYSCON
```

```
SUNLOCK SYSCON * (where SYSCON is metered App. Name)
```

**NOTE:**

*The errorlevel = 100 tests the return code from SLOCK. All return codes for SLOCK and SUNLOCK are listed later in this appendix. The asterisk (\*) will display SLOCK messages.*

SYSCON is the metered application name as it appears in the metering definition. By requiring people to use a batch file to run SYSCON, you are modifying the way users operate that application, resulting in increased control over the network through SiteMeter, even without SWATCHER loaded.

### Using SLOCK & SUNLOCK

This section describes how to use SLOCK and SUNLOCK.

Use the following procedure:

1. **Create a LockSet for SYSCON.**

**2. Use your text editor to create a batch file called TEST.BAT.**

This batch file should contain the three lines from the previous section.

**3. At the DOS prompt, type TEST <ENTER>.**

The first command in the batch file will now be executed. Once you have secured a position in the LockSet, the following message appears:

```
Position has been obtained in LockSet SYSCON.
```

At this point, the screen for SYSCON displays. You have now successfully gained access to SYSCON.

**4. Exit the application.**

The command “SUNLOCK SYSCON” is executed as soon as you exit SYSCON. The following message appears:

```
Your position has been released in LockSet SYSCON.
```

The following batch file shows examples of using errorlevels in conjunction with SLOCK and SUNLOCK. Here, if the errorlevel returns a successful code, the software package Multimate will be accessed. If not successful, a message will appear on the screen telling the user that all the positions in the LockSet for Multimate are taken. At the end of each course, SUNLOCK will be executed, releasing the position in the LockSet.

```
echo off

SLOCK MULTIMATE

if errorlevel= 100 goto go

goto Again

:go

MMS.EXE (Multimate)
```

```

goto end

:Again

echo 'Try later, Multimate is licensed out...' pause

:end

SUNLOCK MULTIMATE (metered application name)

```

### Command Line Reference

This section displays the correct syntax for SLOCK and SUNLOCK and explains the command line options.

The correct syntax for SLOCK is:

```
SLOCK LockSet [*] [S=fileserver]
```

**NOTE:**

*The brackets indicate optional entries. Do NOT type the brackets when using these options.*

OPTION	DESCRIPTION
LOCKSET	This is a required entry. The name of the LockSet must be entered. The LockSet is the metered application.
*	Displays message flag. Designating this option provides the user with visual confirmation that the commands have been executed.
S=fileserver	This option denotes the name of the fileserver that contains the LockSet.

The correct syntax for SUNLOCK is:

```
SUNLOCK LockSet [*] [S=fileserver]
```

**NOTE:**

*The brackets indicate optional entries. Do NOT type the brackets when using these options.*

OPTION	DESCRIPTION
LOCKSET	This is a required entry. The name of the LockSet must be entered. The LockSet is the metered application.
*	Displays message flag. Designating this option provides the user with visual confirmation that the commands have been executed.
S=fileserver	This option denotes the name of the fileserver that contains the LockSet.

## Error Levels

SiteMeter has two types of error reporting that can be printed to the screen: text messages and numeric codes. Not all programs contain both types of errors. This section explains the numeric codes.

All programs will report with an on-screen message when there is an attempt to:

- ☐ Execute the utility while not using Advanced NetWare
- ☐ Use any of the utilities while not logged into the network (other than SWATCHER)

Numeric code returns are offered for both SLOCK and SUNLOCK. The following sections list these codes.

## SLOCK Errorlevels

### 100

User has successfully obtained a position in the desired LockSet.

### 99

Bad LockSet name or lock set name is too long.

### 88

Invalid LockSet name or name does not exist.

### 77

LockSet is in use by someone using SiteMeter to change some parameter about that set.

### 66

After testing the LockSet 30 times, it is still locked (not by the Supervisor--77)

### 58

No slots available in LockSet, you have been queued.

### 56

No queue slots available (maximum of 8); you were not queued.

**55**  
No slots available in LockSet...try later.

**50**  
Not attached to specified server.

## **SUNLOCK Errorlevels**

**100**  
User's slot has been successfully found and released.

**99**  
Bad LockSet name or LockSet name is too long.

**88**  
Invalid LockSet name; name does not exist.

**77**  
LockSet is in user by someone using SiteMeter to change a parameter about that LockSet.

**66**  
After testing the LockSet 30 times, it is still locked (not by Supervisor--error 77).

**55**  
No slot in use.

**50**  
Not attached to specified server.

**33**  
No rights to SYS:SYSTEM\ITEMETR, or it does not exist. The name (directory) is defined by the SiteMeter Administrator.

**22**  
No network drives are mapped. SUNLOCK 'borrows' one of the current station's network drive letters to update the activity log, but if none exist the activity log will not be updated.