

White Paper: Easy Remote Program Installations Using the Task Scheduler Service

Rev 4 – June 1, 2006

Lieberman Software Corporation
<http://www.liebsoft.com>

Abstract

In this paper you will see how simple software installations can be easily accomplished using the built-in Task Scheduler service already installed in every Windows® NT, 2000, XP and Server 2003 machine.

Contents

1. Introduction	3
2. Requirements and Methods of Software Distribution	3
3. Packaged Distribution	3
4. Remote Execution of the Installation Package	4
5. Other Free Alternatives to the AT Command	5
6. Going Further	6
7. A Simple Step-up from Scripts	6
8. Summary	8

1. Introduction

The periodic distribution of new programs, patches and updates represent a large portion of an administrator's duties. The challenge is to find the right tool to do the distribution with a minimum of fuss and muss. It turns out that you can do a pretty good job at simple software distribution with minimal effort by just doing remote file copies combined with creating jobs in each remote system to install the just copied file(s). The rest of the paper will walk you through the details.

For really complex installations you have a wealth of options. Products such as Microsoft's SMS, HP's OpenView, Novell's ZenWorks and IBM's Tivoli do an excellent job with even the most complex installations. Just remember that these tools do take a significant amount of time and money to setup and manage, but they can do just about anything you can conceive of when it comes to program installation.

2. Requirements and Methods of Software Distribution

To push a new software package, the following minimal steps are required:

- Distributing the package – this is just copying the file(s) (easy)
- Remote Execution of the Package – how do you remotely run the program?

3. Package Distribution

One of the nice things about administering Windows NT and later operating systems is that they are chock full of remote management services. In this case, the distribution of a software package can be done easily since each hard drive letter is automatically shared out over the network (C\$, D\$, E\$...). Even the directory used by the operating system has its own share (admin\$) that is available to read/write across the network. This directory sharing (as well as other management features) is accomplished by the Server service running on every NT machine.

Copying to remote destination directories on each machine can be accomplished by either using the built-in command-line Copy command or the standalone XCOPY.EXE program (the resource kit Robocopy program is another excellent choice). Copying is a matter of selecting the source directory/files (usually stored locally) and then setting the destination path. To handle a destination path on a remote system, the path must be encoded using UNC (Universal Naming Convention) in the format:

```
\\Server\ShareName\DestinationPath
```

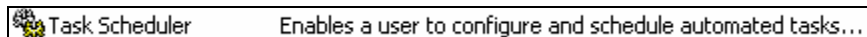
For the copy operation to succeed, the copy command (or copy program) must be executed while running under an account that is seen by all target systems as an administrator¹. The common administrator account requirement can be quite a challenge if the distribution must be done across multiple domains without trusts, or to multiple workgroups. Another challenge is the handling of off-line machines via some kind of auto-retry mechanism.

¹ Technically, the account only must be able to access the share and write files to the destination directory on the remote machine through the share. The term "administrator" is used to indicate an account with these rights in this context.

4. Remote Execution of the Installation Package

The secure remote execution of the installation package is the primary reason most administrators buy a third party software management package.

A free secure remote execution agent is already installed in all Microsoft Windows NT, 2000, XP and Server 2003 machines and it is known as the "Task Scheduler" service. The Task Scheduler can be remotely controlled to run any program either immediately or at a specific time and date. It even produces a log of the success or failure of the remote launch.



The least expensive way to manage the jobs on remote systems is via the command-line accessible AT command. To see all of the options of the AT command, just type:

AT /?

Here is the output of the command:

```
The AT command schedules commands and programs to run on a computer at
a specified time and date. The Schedule service must be running to use
the AT command.

AT [[\computername] [ [id] [/DELETE] | /DELETE [/YES]]
AT [[\computername] time [/INTERACTIVE]
    [ /EVERY:date[,...]] | /NEXT:date[,...]] "command"

\computername    Specifies a remote computer. Commands are scheduled on the
                  local computer if this parameter is omitted.
id                Is an identification number assigned to a scheduled
                  command.
/delete           Cancels a scheduled command. If id is omitted, all the
                  scheduled commands on the computer are canceled.
/yes              Used with cancel all jobs command when no further
                  confirmation is desired.
time              Specifies the time when command is to run.
/interactive      Allows the job to interact with the desktop of the user
                  who is logged on at the time the job runs.
/every:date[,...] Runs the command on each specified day(s) of the week or
                  month. If date is omitted, the current day of the month
                  is assumed.
/next:date[,...]  Runs the specified command on the next occurrence of the
                  day (for example, next Thursday). If date is omitted, the
                  current day of the month is assumed.
"command"        Is the Windows NT command, or batch program to be run.
```

To run a just copied installation package, the administrator would issue the command:

```
AT \\SERVER 01:00 "C:\program files\my program\setup.exe  
-s:data.inf"
```

In this case we would be running the setup.exe program at 1:00 AM in our just created directory and the installation program would get its installation parameters from the data.inf file. Of course, each installation program has its own syntax and method of passing the setup parameters to an installation program.

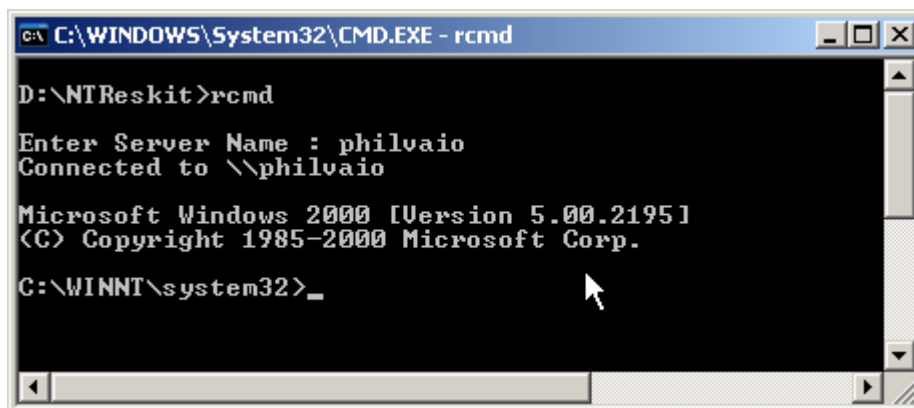
The log of the Task Scheduler can be obtained by browsing to each machine², opening the Scheduled Tasks folder, and then using the menu option:

Advanced | View Log

5. Other Free Alternatives to the AT Command

When you need to execute a remote installation, there are plenty of other free programs in the Microsoft Server Resource Kit that perform remote execution. However, all of the following programs require the installation of a remote service or the starting of a remote program on each remote system.

The Remote Command Service consists of the client program: RCMD.EXE and its remote matching service RCMDSVC.EXE. The RCMDSVC service needs to be copied to every machine and installed. Remote installation of the service can be accomplished by the SC (for information, use the command: SC /?) command or en masse to all machines via Lieberman Software's Service Account Manager. This program performs secure authentication before any transactions are allowed. Below is an example of the Remote Command Service:



² This only works if you're browsing to the machine through the Network interface, and the version of the operating system you're using shows the Scheduled Tasks folder under the machine in the Network browser. If you browse to the machine's task folder using a network share, such as \\SERVER\Admin\$\Tasks, and try to open the log, you will get your local Task Scheduler log file.

The Remote Command Line consists of a single file: REMOTE.EXE (also in the Server Resource Kit). This program is not recommended since there is no security provided and anyone can perform remote commands to any machine where the REMOTE.EXE program is running. Manual installation and starting is required on all remote machines.

Another alternative is the Remote Console utility in the Resource Kit. This utility operates like a Telnet session in Unix. It is located in the Resource Kit directory under \RCONSOLE. This program consists of a remote network service that must be installed manually on every machine, and a remote control called RCLIENT.EXE. This utility is more suited for a one-by-one remote installation.

Finally, the TCP/IP Remote Shell Service operates like the Unix Remote Shell Service (rsh). This is a service-based package also provided in the Server Resource Kit. This tool installed using the same methods as the Remote Command Service. Its security depends on the creation of a custom RHOSTS file that specifies all machines and users that are allowed remote access via this service.

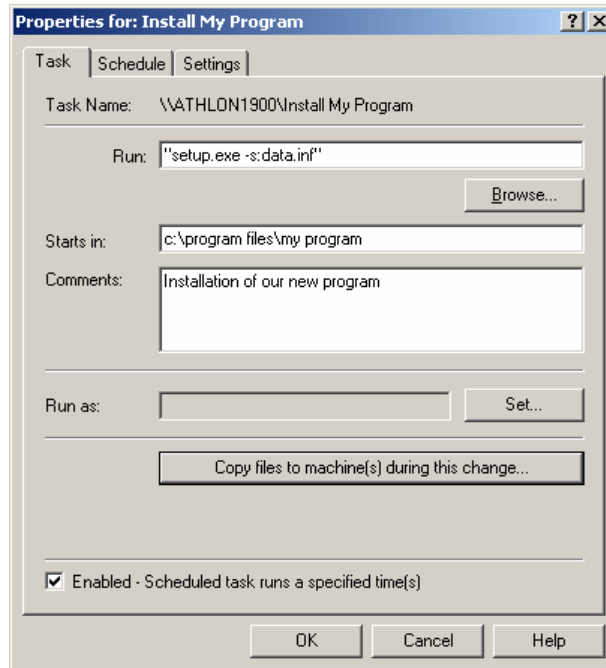
6. Going Further

Using scripts to drive the copy and AT commands (or remote control programs/services) allows software installations at virtually zero cost other than your development time. Some of the goodies you generally miss with scripts include:

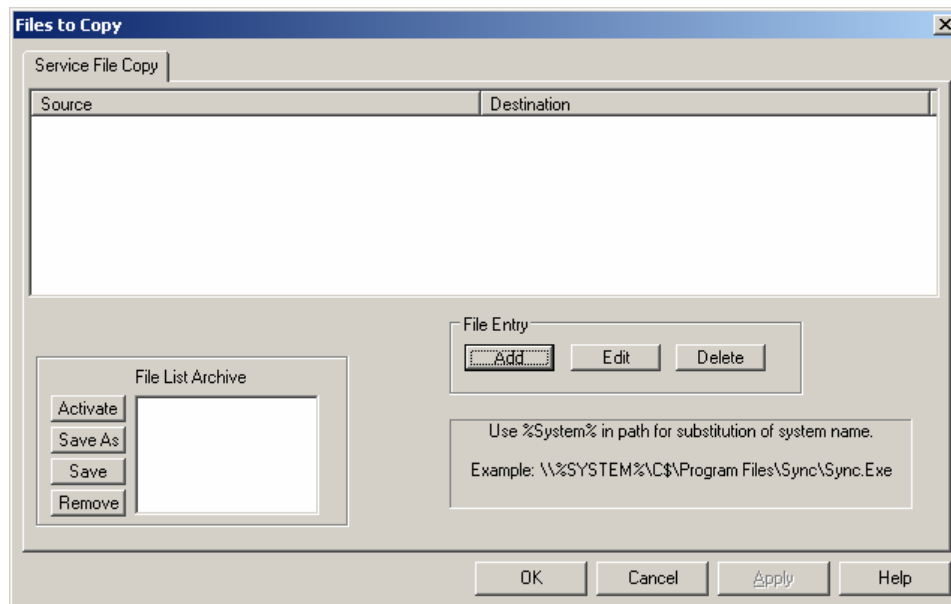
- Easy management of a dynamically changing machine list
- No auto-retry of off-line machines
- Multi-threaded operation for concurrent installation where speed is important
- Comprehensive logging of operations (auditing)
- Ability to manage the distribution of more than one package
- Extensive reporting of installation status
- May support impersonation to allow cross domain/workgroup installation
- You have to document and maintain that script you create (not a lot of fun)

7. A Simple Step-up from Scripts

A step up from writing scripts and a step down from using a full blown software distribution package is the use of a specialized tool to do mass management of the scheduled tasks on all your remote machines. One such tool is Task Scheduler Pro™ from Lieberman Software (<http://www.liebsoft.com>). Although the product is not a software distribution product, it has an integrated file copying option included in its scheduled task management function. By combining task management with built-in concurrent file copying, the program can be used to quickly install simple software packages in minutes. The program interfaces are nearly identical to those in Microsoft's own tools, except for the addition of the file copying option:



When this copy button is pressed, the program provides a general purpose file copy editor:



As an added bonus, the program can also simultaneously retrieve all of the Task Scheduler logs from all of your systems for immediate review. It also has automatic retry, so if any of the target systems are off-line the program will keep trying to do the installation until it is completed.

Task Scheduler Pro is fully multithreaded for fast simultaneous operation, logs all transactions to a permanent text file, and supports unlimited impersonation so that cross-domain and multiple workgroups are supported without having to log off and on multiple times.

For those extremely familiar with the AT command, the account AT uses is normally hard coded per machine and requires a personal visit to each machine to change the account. Task Scheduler Pro can display the accounts used by the AT command on each system and allow you to change them all remotely!

8. Summary

For complex software installations, full-featured third party packages are the ideal choice for administrators. For those without a budget, simple scripts combining copying and the AT command can fit the bill nicely. And finally, for those that need to do simple installations and don't have a lot of time to create scripts, and are looking for more functionality, there is the Task Scheduler Pro product from Lieberman Software.

Our support staff is available to answer your technical questions whether you are a customer or not.

Voice: 800.829.6263 (USA/Canada) Voice: (01) 310.550.8575 (Worldwide) Fax: (01) 310.550.1152 (Worldwide)
Web: www.liebsoft.com Email: support@liebsoft.com

