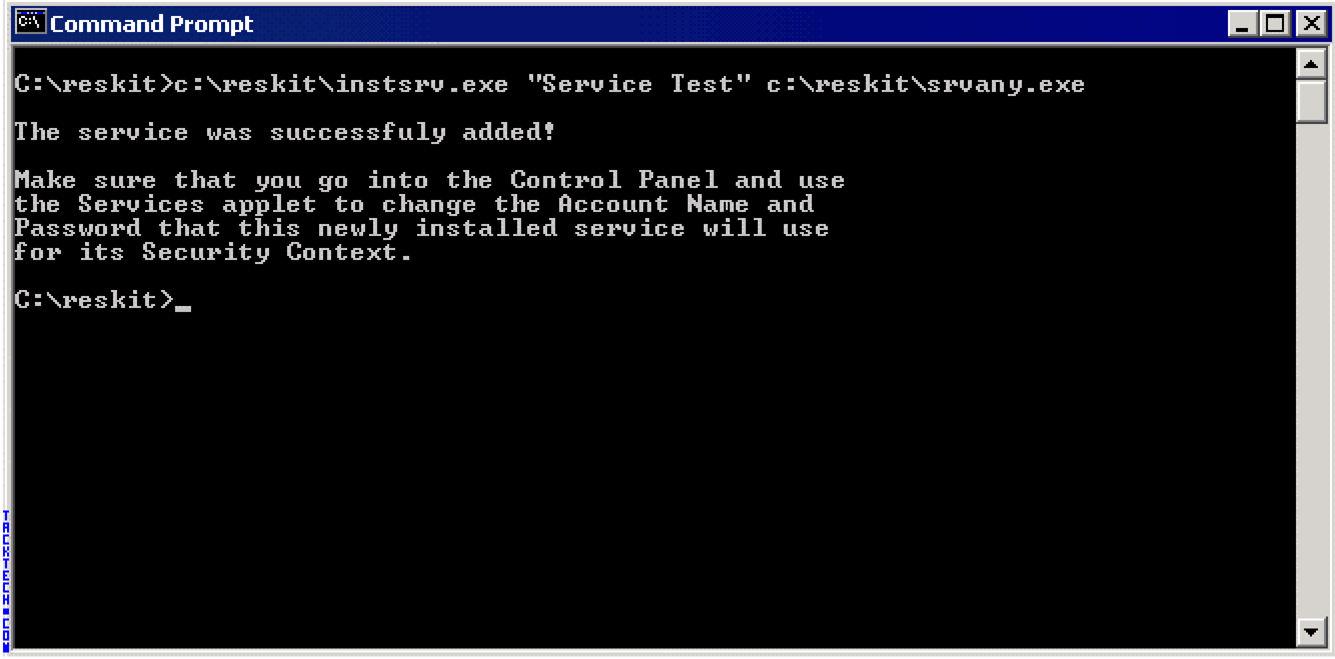
Create your own user-defined services WinNT/2000/XP

The Windows NT/2000/XP Resource Kit provides two utilities that allow you to create a Windows NT user-defined service for Windows NT applications and some 16-bit applications (but not for batch files).

Instrsrv.exe installs and removes system services from Windows NT and Srvany.exe allows any Windows NT application to run as a service. You can download the files here srvany.zip (you can download this file from the following URL: "http://www.tacktech.com/pub/service/srvany.zip" This zip includes three files. The two you need srvany.exe and instsrv.exe to install the services.

Note: Make sure the Services Manager is closed while running the DOS commands.

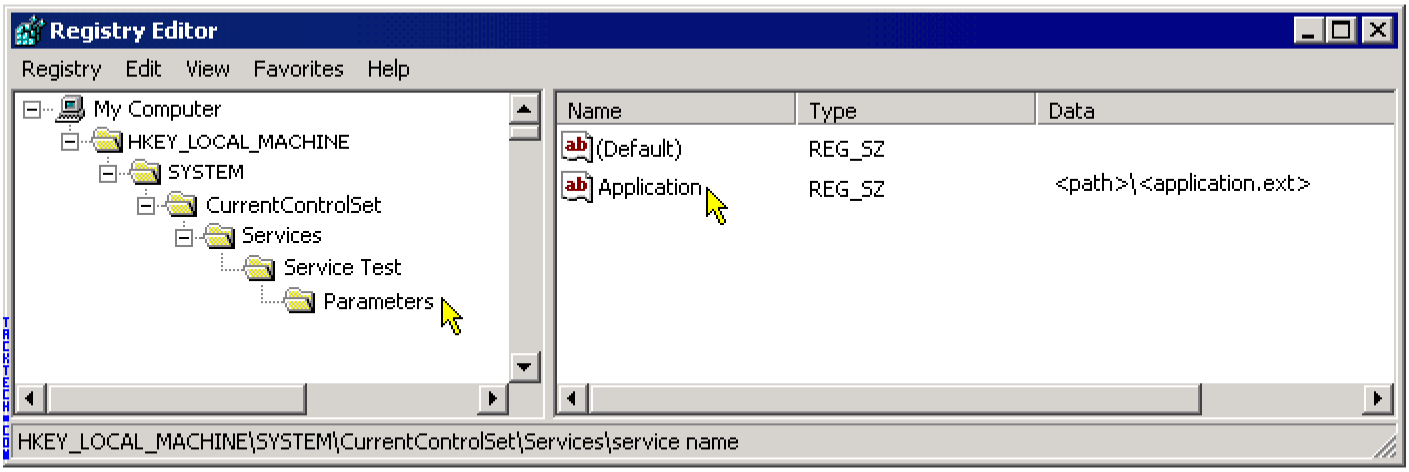
You will need to put these files in a directory called **reskit** At a MS-DOS command prompt(Start | Run | "cmd.exe"), type the following command:
<path>\reskit\INSTSRV.EXE "*Service Name*" <path>\reskit\SRVANY.EXE
This creates the service in the Services manager and the registry keys to setup what program to run.



```
C:\reskit>c:\reskit\instsrv.exe "Service Test" c:\reskit\srvany.exe

The service was successfuly added!

Make sure that you go into the Control Panel and use
the Services applet to change the Account Name and
Password that this newly installed service will use
for its Security Context.

C:\reskit>_
```

Next open regedit.exe **Start | run | regedit.exe**
WARNING: Using Registry Editor incorrectly can cause serious problems that may require you to reinstall your operating system. Microsoft cannot guarantee that problems resulting from the incorrect use of Registry Editor can be solved. Use Registry Editor at your own risk.

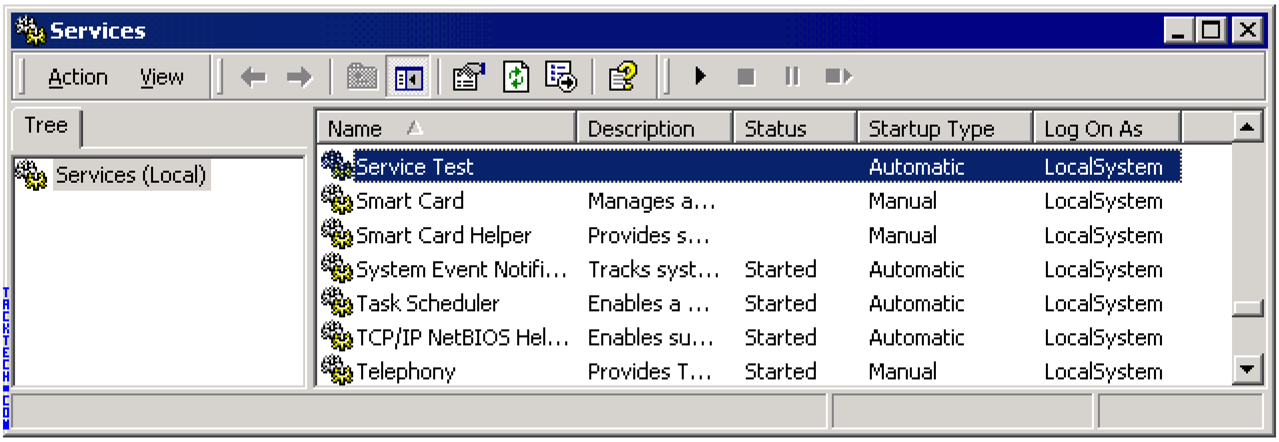
From the Edit menu, click Add Key and name it **Parameters**
Next from the Edit menu, click Add Value and type this information.
Value Name: Application
Data Type : REG_SZ
String : <path>\<application.ext>



Name	Type	Data
(Default)	REG_SZ	
Application	REG_SZ	<path>\<application.ext>

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\service name

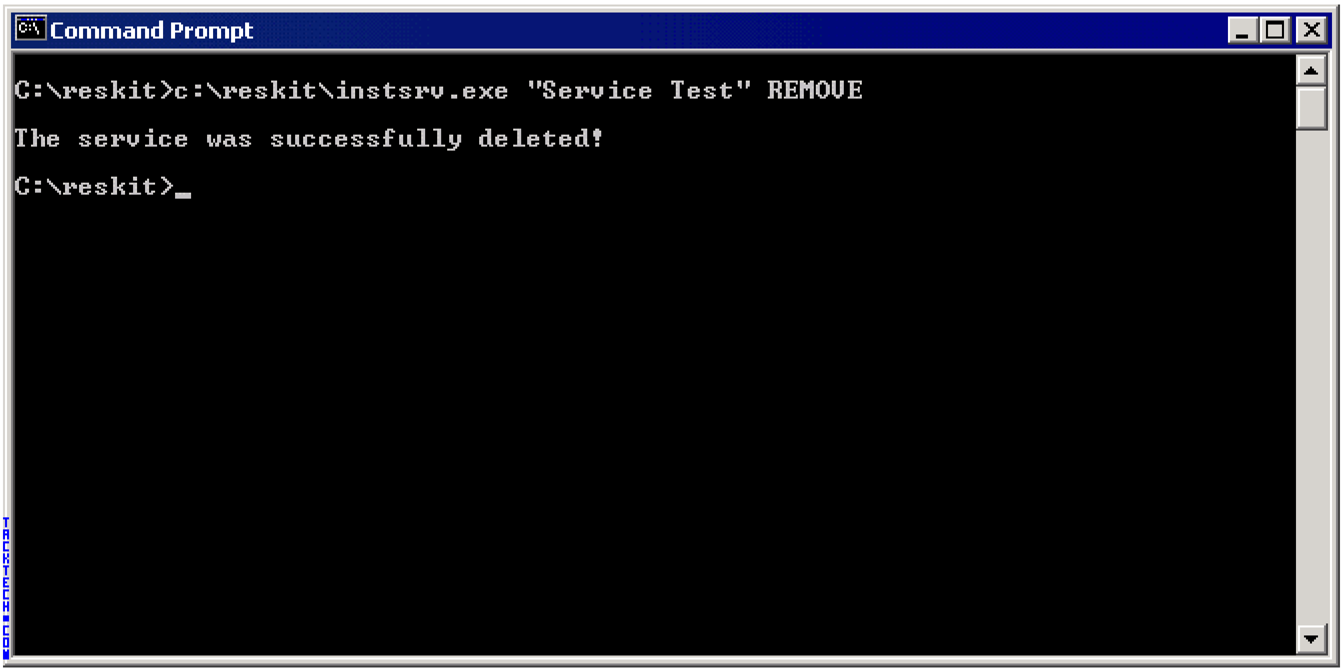
Now you can start your service from the Service Manager



Name	Description	Status	Startup Type	Log On As
Service Test			Automatic	LocalSystem
Smart Card	Manages a...		Manual	LocalSystem
Smart Card Helper	Provides s...		Manual	LocalSystem
System Event Notifi...	Tracks syst...	Started	Automatic	LocalSystem
Task Scheduler	Enables a ...	Started	Automatic	LocalSystem
TCP/IP NetBIOS Hel...	Enables su...	Started	Automatic	LocalSystem
Telephony	Provides T...	Started	Manual	LocalSystem

With this same program you can remove the service also. Just run this command from command prompt.

<path>\reskit\INSTSRV.EXE *"Service Name"* REMOVE



```
C:\reskit>c:\reskit\instsrv.exe "Service Test" REMOVE

The service was successfully deleted!

C:\reskit>_
```

How to use SRVANY to Create Customized

This Document describes the **SRVANY.EXE** utility. This utility allows running **Windows NT™** applications as services. The benefits include:

- allow apps to survive logoff/logon sequences, hence saving the overhead of re-starting them for each new user
- allow server apps to come-up and service requests even when no user is logged-on
- allow apps to run and perform a task in a specific logon account, different from the currently logged-on user

NOTICES:

1- Some applications may terminate upon logoff, even though they were started as a service, if they don't ignore the WM_ENDSESSION message (or TRL_LOGOFF_EVENT). See "Programming Considerations" below for more details.

2- When using **SRVANY.EXE** with **Presentation Manager®** applications, special configuration instructions apply - please read the README.WRI file shipped on Disk#1 of the **Windows NT™ Add-On Subsystem for Presentation Manager®, version 3.51** (a Microsoft add-on product which allows running 16-bit **Presentation Manager®** 1.x applications in addition to the OS/2 character-mode applications supported by the **Windows NT™** OS/2 Subsystem).

Installation:

1-Copy SRVANY.EXE to your system and install it as a **Windows NT™** service, for example:

```
INSTSRV MyService c:\tools\srvany.exe
```

2-configure via the Services applet ("Startup..." dialog) of the Control Panel as manual or automatic, as appropriate.

3- set via the Services applet ("Startup..." dialog) of the Control Panel the account for the service. If you need access to the screen & keyboard, you must choose the LocalSystem account and click the "Allow Service to Interact with Desktop", otherwise choose any account and specify the correct logon password. Note that the LocalSystem account doesn't have network access.

Specifying the application to start & its parameters:

Run the Registry Editor (REGEDT32.EXE):
1-under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MyService: create a 'Parameters' key

2-under the above key, create an 'Application' value of type REG_SZ and specify there the full path of your app executable (including the extension). For example:

```
Application: REG_SZ: D:\TOOLS\VI.EXE
```

3-OPTIONAL: under the above key, create an 'AppParameters' value of type REG_SZ and specify the parameters for your app. For Example:

```
AppParameters: REG_SZ: C:\TMP\FOO
```

4-OPTIONAL: under the above key, create an 'AppDirectory' value of type REG_SZ and specify the current directory to use for the app, eg:

```
        AppDirectory: REG_SZ: C:\TMP
```

The case of all above strings is not important.

There is also another way to specify the service parameters, via the Control Panel. Note that this method applies only to the case where the service is configured to be started manually (i.e. not when it is started automatically with the system):

or Via the Control Panel, Services applet:

1- type in the Startup Parameters box the full command-line, incl. the app name. Example:

```
        D:\\BINP\\B.EXE C:\\TMP\\FOO
```

(note the '\\' is needed to result in '\')

2- if you want to specify the current working directory, use the '/D' switch before the app name, eg:

```
        /D C:\\TMP D:\\TOOLS\\VI.EXE C:\\TMP\\FOO
```

NOTICE: Specifying the target application via the Control Panel is usually inconvenient because the parameters have to be typed every time the service is started. However, this may be convenient in cases where the target application is different every time the service is started.

Starting & stopping the service:

Start:

If the service is configured as 'Automatic', the user doesn't need to start it explicitely: it is started automatically every time when the system is re-booted.
For 'manual' services, the user may start services in several ways:

- from the Services applet of the Control Panel
  or
- using the SC.EXE utility, eg: SC start MyService
  or
- NET START MyService

Stop:

When you stop the service, it will terminate the application it had started. The way to stop the service is (as described above for starting):

- from the Services applet of the Control Panel
  or
- using the SC.EXE utility, eg: SC stop MyService
  or
- NET STOP MyService

WARNING:
When the service is stopped, it terminates the application via the WIN32 TerminateProcess() API: this is a drastic way to end an application. For example, it would not allow the application to prompt the user to save changes. Therefore, it is recommended to close the application BEFORE stopping the service.

De-installation:

If you want to prevent an instance of the SRVANY utility from running until further notice, you should configure it via the Services applet ("Startup..." dialog) of the Control Panel as 'Disabled' (rather than 'manual' or 'automatic').

If you want to remove an instance of the SRVANY utility permanently:
-> If the service is running, stop it (see "Starting and stopping the service" above)
-> Run:

```
INSTSRV MyService remove
```

NOTICE: this procedure removes only one specific instance of SRVANY. Therefore, it is possible that SRVANY.EXE is still being used to start other applications as services.

Programming Considerations:

For WIN32 graphical applications: when the currently logged-in user is logging-off, all WIN32 top-level windows receive WM_QUERYENDSESSION and WM_ENDSESSION messages. Some WIN32 applications choose to terminate upon receipt of such messages. In order for your WIN32 application to survive logoff, it must not do that: instead, your windows procedure should call the default windows procedure on these messages.

For WIN32 Console (i.e. character-mode) applications: when the currently logged-in user is logging-off, all Console applications receive a CTRL_LOGOFF_EVENT event from the Console.
If your Console application has registered a Console event handler (via SetConsoleCtrlHandler), it must ignore CTRL_LOGOFF_EVENT in order to survive the logoff.

Comments/limitations:

You may install SRVANY.EXE several times with different registry parameters (i.e. running different target applications) - just use a distinct service name for each instance (e.g. MyService1, MyService2 etc.)

If SRVANY.EXE fails to start your application, try specifying as current directory the directory where your application is installed (see 'AppDirectory' registry key or '/D' above). SRVANY.EXE may be running under an account different than the currently logged-on user therefore environment variables may be set differently: as a result, for example, the system might have been unable to find a DLL required for your application and running it from the application's directory might help.

Due to a restriction enforced by **Windows NT™** on services, the application can either be interactive (have a Console, read keyboard input etc.) or have network access (not both at the same time).

This limitation is less problematic than it would appear at first glance. Here is why:
If you know which servers your interactive service application needs to access, it is possible to configure these servers (or selected shares on them) to allow your service (running under the LocalSystem account) to access these servers.
a.To allow LocalSystem services on any machine in the domain to access a specific share on a server, use the Registry Editor to add the name of that share to:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters\NullSessionShares
If named pipes on that server also need to be accessed by LocalSystem services, add them to:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters\NullSessionPipes

b.To allow ALL shares & pipes on the server to be accessed by LocalSystem services, add a value:
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters\RestrictNullSessAccess, type DWORD, set to value 0

WARNING: method b) effectively allows everyone in the domain to access that server - make sure this is acceptable for you.