

GetTempPath function

Retrieves the path of the directory designated for temporary files.

Syntax

C++

```
DWORD WINAPI GetTempPath(  
    _In_  DWORD  nBufferLength,  
    _Out_ LPTSTR lpBuffer  
);
```

Parameters

nBufferLength [in]

The size of the string buffer identified by *lpBuffer*, in **TCHARs**.

lpBuffer [out]

A pointer to a string buffer that receives the null-terminated string specifying the temporary file path. The returned string ends with a backslash, for example, "C:\TEMP\".

Return value

If the function succeeds, the return value is the length, in **TCHARs**, of the string copied to *lpBuffer*, not including the terminating null character. If the return value is greater than *nBufferLength*, the return value is the length, in **TCHARs**, of the buffer required to hold the path.

If the function fails, the return value is zero. To get extended error information, call [GetLastError](#).

The maximum possible return value is **MAX_PATH**+1 (261).

Remarks

The **GetTempPath** function checks for the existence of environment variables in the following order and uses the first path found:

1. The path specified by the TMP environment variable.
2. The path specified by the TEMP environment variable.
3. The path specified by the USERPROFILE environment variable.

4. The Windows directory.

Note that the function does not verify that the path exists, nor does it test to see if the current process has any kind of access rights to the path. The **GetTempPath** function returns the properly formatted string that specifies the fully qualified path based on the environment variable search order as previously specified. The application should verify the existence of the path and adequate access rights to the path prior to any use for file I/O operations.

Symbolic link behavior—If the path points to a symbolic link, the temp path name maintains any symbolic links.

In Windows 8 and Windows Server 2012, this function is supported by the following technologies.

Technology	Supported
Server Message Block (SMB) 3.0 protocol	Yes
SMB 3.0 Transparent Failover (TFO)	Yes
SMB 3.0 with Scale-out File Shares (SO)	Yes
Cluster Shared Volume File System (CsvFS)	Yes
Resilient File System (ReFS)	Yes

Examples

For an example, see [Creating and Using a Temporary File](#).

Requirements

Minimum supported client	Windows XP [desktop apps Windows Store apps]
Minimum supported server	Windows Server 2003 [desktop apps Windows Store apps]
Header	FileAPI.h (include Windows.h); WinBase.h on Windows Server 2008 R2, Windows 7, Windows Server 2008, Windows Vista, Windows Server 2003, and Windows XP (include Windows.h)

Library	Kernel32.lib
DLL	Kernel32.dll
Unicode and ANSI names	GetTempPathW (Unicode) and GetTempPathA (ANSI)

See also

[File Management Functions](#)

[GetTempFileName](#)

[Symbolic Links](#)