How to get private directory

Platform Security introduces the data caging concept. There are some folders on Symbian OS v9 that can only be accessed with certain privileges. The private folder, `\private\<sid>`, is one of them. It is a folder where an application can store sensitive data. Other applications, except the one with AllFiles capability, cannot read/write to the folder.

The <sid> is the Secure ID of the application. It is defined in the .mmp file. The following example shows how to get the path of the private folder of an application. For example, if an application has SID of 0x20001978, the private folder would be \private\20001978. The function that is used to get the private folder is `RFs::PrivatePath()`.

Example

```
const TInt KPathNameLength = 100;
TBuf< KPathNameLength > pathWithoutDrive;
TBuf< KPathNameLength > driveAndPath;
TBuf< 2 > appDrive;

// Get application's private path
//   This is e.g. '\Private\20001978\'
//   Does not contain drive.
iEikonEnv->FsSession().PrivatePath( pathWithoutDrive );

// Extract drive letter into appDrive:
appDrive.Copy(CEikonEnv::Static()->EikAppUi()->Application()->AppFullName().Left(2));

// Combine drive letter and private path
driveAndPath.Copy(appDrive);
driveAndPath.Append(pathWithoutDrive);

// Now driveAndPath contains e.g. 'E:\Private\20001978\'
```

Example 2

```
#include <f32file.h>
TInt GetPrivatePath(TFileName& privatePath)
{
 TFileName KPath;
 RFs fsSession;
 TInt result;
 result = fsSession.Connect();
 if (result != KErrNone)
  return result;
 fsSession.PrivatePath(KPath);
 TFindFile findFile(fsSession);
 privatePath = KPath;
 result = findFile.FindByDir(KPath, KNullDesC);
 if (result == KErrNone)
  privatePath = findFile.File();
 fsSession.Close();
 return result;
}
```


(C) Copyright Nokia 2014. All rights reserved.

The code must be linked with efsrv.lib. The resulting path contains backslash at the end.

(C) Copyright Nokia 2014. All rights reserved.