Invalidating User Reference Numbers

It is your responsibility to invalidate all user reference numbers obtained with the **StartSecureSession** function before your application quits. However, while your application remains open, you may want to keep track of a user reference number to start a session with a port, end it, and then later start another session with the same port.

Use the **DeleteUserIdentity** function to invalidate the user reference number for a particular user.

The **DeleteUserIdentity** function removes a user by invalidating the specified user reference number. Note that you cannot invalidate the guest reference number (0) and, in most cases, you should not dispose of the default user reference number.

The following program illustrates how you use the **DeleteUserIdentity** function to invalidate a user reference number obtained from a **StartSecureSession** function. The sample code does not invalidate the user reference number if it is either the default user reference number or the guest reference number (0).

```
// Using the DeleteUserIdentity function to invalidate a user identity

// Assuming inclusion of MacHeaders
#include <PPCToolBox.h>

// Prototype your routine like this prior to calling it
OSErr DeleteNewUserRefNum(long);

OSErr  DeleteNewUserRefNum(long newUserRef)

{
    OSErr  err;
    long    defUserRef;
    Str32  defUserName;

    if ( newUserRef ) {
        // user reference number passed was not the guest
        // first arguement is expected to be unsigned, thus cast it
        err = GetDefaultUser((unsigned long *)&defUserRef, defUserName);
        if ( !err ) {
                // there is a default user
        if ( newUserRef != defUserRef )
            // it's not the default, so delete it
            err = DeleteUserIdentity(newUserRef);
        }
        else
            // there is no default, so delete it
            err = DeleteUserIdentity(newUserRef);
            return err;
        }
        else
            // user reference number passed was the guest
            return noErr;
```

```
}
```