Archived:Authorization requires authentication in Bluetooth security manager on S60 2nd Edition (Known Issue)

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The article is believed to be still valid for the original topic scope.

Overview

Using authorization mandates using authentication when the TBTServiceSecurity class is used in certain devices. If there is a need for an authorization feature, also authentication has to be set.

Description

The Bluetooth Security Manager faced some changes in Symbian OS v8.0a. In contrast to earlier platforms, it is not possible to use authorization (user confirmation for the inbound connection) without authentication (bonding between two devices).

How to reproduce

Set TBTServiceSecurity's authentication OFF (false) and authorization ON (true). Then connect the devices. The devices get connected without any dialog for the user to accept/decline the connection request.

Solution

You have to set both authorization and authentication ON (=true) to be able to use authorization (which is actually also recommended to ensure better security).
An example of how to set authorization ON:

```
...
// Get the port and set it to the listening socket
TInt channel;
User::LeaveIfError
( iListeningSocket.GetOpt( RFCOMMGetAvailableServerChannel,
 KSolBtRFCOMM, channel ));
TBTSockAddr listeningAddress;
listeningAddress.SetPort( channel );
// Define security for the application (unique service ID,
// authentication ON, encryption ON; authorization ON)
TBTServiceSecurity serviceSecurity;
serviceSecurity.SetUid ( KUidBTPointToPointApp );
serviceSecurity.SetAuthentication ( ETrue );
serviceSecurity.SetEncryption ( ETrue );
serviceSecurity.SetAuthorisation ( ETrue );
serviceSecurity.SetDenied( EFalse );
// Attach the defined security policy to the socket
listeningAddress.SetSecurity(serviceSecurity);
// Bind the socket
User::LeaveIfError( iListeningSocket.Bind( listeningAddress )
);
User::LeaveIfError( iListeningSocket.Listen( KListeningQueSize
) );
...
```


(cc) Copyright Nokia (2013) All rights reserved.

(C) Copyright Nokia (2013) All rights reserved.