Die Bluetoothverbindung zwischen Brille

Und Tablet und Smartwatch und Tablet wird auf zwei unterschiedliche Weisen.

Die Verbindung zwischen Brille und Tablet stellt ein unsichers Vefahren ohne Koppelung bereit. Weder die Datenbrille, noch der Programmteil, welcher auf der Datenbrille laufen, sind in der Smartwatchapplikation hinterlegt.

Diese Art der Verbindung stellt etwa keine Kontrollmechanismen über Angreifer wie zwischengeschaltete Instanzen zur Verfügung. Bei dem genutzen Bluetooth Protokol handelt es sich um RFCOMM, über die Methoden der Android Programmiersprache wird eine Socket Verbindung eingerichtet: <https://developer.android.com/guide/topics/connectivity/bluetooth.html#ConnectingDevices>. Das Tablet stellt als Bluetoothdevcice einen Bluetooth Socket zur Verfügung durch Aufruf der Android Methode [createRfcommSocketToServiceRecord(UUID)](https://developer.android.com/reference/android/bluetooth/BluetoothDevice.html#createRfcommSocketToServiceRecord(java.util.UUID)): „ The UUID passed here must match the UUID used by the server device when it called [listenUsingRfcommWithServiceRecord(String, UUID)](https://developer.android.com/reference/android/bluetooth/BluetoothAdapter.html" \l "listenUsingRfcommWithServiceRecord(java.lang.String, java.util.UUID)) to open its [BluetoothServerSocket](https://developer.android.com/reference/android/bluetooth/BluetoothServerSocket.html). To use a matching UUID, hard-code the UUID string into your application, and then reference it from both the server and client code.“

Initiate the connection by calling [connect()](https://developer.android.com/reference/android/bluetooth/BluetoothSocket.html#connect()). Note that this method is a blocking call.

After a client calls this method, the system performs an SDP lookup to find the remote device with the matching UUID. If the lookup is successful and the remote device accepts the connection, it shares the RFCOMM channel to use during the connection, and the [connect()](https://developer.android.com/reference/android/bluetooth/BluetoothSocket.html#connect()) method returns. If the connection fails, or if the [connect()](https://developer.android.com/reference/android/bluetooth/BluetoothSocket.html#connect()) method times out (after about 12 seconds), then the method throws an {java.io.IOException}.

Because [connect()](https://developer.android.com/reference/android/bluetooth/BluetoothSocket.html#connect()) is a blocking call, you should always perform this connection procedure in a thread that is separate from the main activity (UI) thread.

**Note:**You should always call [cancelDiscovery()](https://developer.android.com/reference/android/bluetooth/BluetoothAdapter.html" \l "cancelDiscovery()) to ensure that the device is not performing device discovery before you call [connect()](https://developer.android.com/reference/android/bluetooth/BluetoothSocket.html#connect()). If discovery is in progress, then the connection attempt is significantly slowed, and it's more likely to fail.

Die Einrichtung benötigt eine Bestätigung der Clientinstanz. Details zum Management der Verbindung finden sich unter <https://developer.android.com/guide/topics/connectivity/bluetooth.html#ManagingAConnection>. Im wesentlichen werden die Daten über einen Input Stream empfangen InputStream und einen Output Stream versendet OutpuStream

Bei der Verbindung zwischen Tablet und Smartwatch handelt es sich um eine sichere Verbindung. Die Applikation der Smartwatch wird im Programmcode des Hauptprogramms auf dem Tablet registriert. Zum Übertragen von Nachrichten koppelt eine Applikation zum Koppeln von Bluetoothgeräten die Smartwatch vor Programmbeginn. Bei Programmablauf stellt dieses Vorgehen bessere Gewährleistung dar, dass es sich bei dem Empfangsgerät tatsächlich um die gekoppelte Smartwatch handelt.

Das Programm verwendet als Framework das Open Source Projekt mit Titel Copyright 2016 The Android Open Source Project, Inc. mit dem Titel Android Bluetooth Chat welches eine Bluetooth Verbindung einstellt, nach anderen Bluetooth Devices sucht, nach gekoppelten Geräten fragt, eine Bluetoothverbindung über das RFCOMM Protokoll <https://de.wikipedia.org/wiki/RFCOMM> einrichtet, eine Verbindung mit einem Gerät einrichtet und den Datenversand vornimmt.

*Die Open Source Lizenz ist hier erhältlich und besteht aus einem Verweis auf das Apache Projekt und sollte in die Klassen eingebunden werden.*