Bluetooth Web API

Bluetooth is a standard for short-range wireless communication between devices. Bluetooth "Classic" defines a set of binary protocols and supports speeds up to about 24Mbps. Bluetooth 4.0 introduced a new "Low Energy" mode known as "Bluetooth Smart", BLE, or just LE which is limited to about 1Mbps but allows devices to leave their transmitters off most of the time. BLE provides most of its functionality through key/value pairs provided by the Generic Attribute Profile (GATT).

BLE defines multiple roles that devices can play. The Broadcaster and Observer roles are for transmitter- and receiver-only applications, respectively. Devices acting in the Peripheral role can receive connections, and devices acting in the Central role can connect to Peripheral devices. A device acting in either the Peripheral or Central role can host a GATT Server, which exposes a hierarchy of Services, Characteristics, and Descriptors.

**Requirements: -**

* Primary requirement is we need to have two devices that has Bluetooth support in it and Bluetooth version should be 4.0 which supports BLE.
* Next we need to have chrome browser in those devices with a version not lower than 53, because these Bluetooth web api is supported only in chrome with version more than 53 as of now.
* The chrome://flags/#enable-experimental-web-platform-features flag must be enabled on Linux and Windows. In Chrome OS, Android, and Mac, the GATT Communication API is available without any flag.

**OS compatibility: -**

* Android: Requires Android 6.0 Marshmallow or later.
* Mac: Requires OS X Yosemite or later.
  + Some MacBooks may not work: Check "About this Mac" / "System Report" / "Bluetooth" and verify that Low Energy is supported.
* Linux: Requires Kernel 3.19+ and BlueZ 5.41+ installed. Read How to get Chrome Web Bluetooth working on Linux.
  + Note that Bluetooth daemon needs to run with experimental interfaces if BlueZ version is lower than 5.43: sudo /usr/sbin/bluetoothd -E
* Windows: Requires Windows 8.1 or later.
  + To discover devices the user hasn't yet manually paired, requires Windows 10.
  + A third-party Windows 10 Polyfill (Chrome Extension + Native Executable) has been created while Windows support is not implemented yet.

**Unsupported platforms**

* Android WebView: Will be supported in the future.
* iOS: Uses the web exposed APIs as provided by the WKWebView, no implementation planned in the Chromium codebase.

Note: WebBLE is an app for iOS that supports the GATT Communication API. It was created initially for the Puck.js project.

For more info on OS compatibility, browser support refer the below link

[https://github.com/WebBluetoothCG/web-bluetooth/blob/gh-pages/implementation-status.md#chrome](https://github.com/WebBluetoothCG/web-bluetooth/blob/gh-pages/implementation-status.md%23chrome)

Initially started working with sample examples from web and tried to execute it, but we were not able to discover devices. Then noticed that in code they have applied filters so it is searching for devices that match the given filters so tried removing filters, made it to discover all devices and found devices. When we select a device and click on pair button in the dialog box it was showing error as **connectGATT ()** was not a function. Then we resolved this error by changing it to **device. gatt. connect ()** function.

Then we thought GATT server needs be started manually and started to work on it. But we found that every Bluetooth device with version 4.0 will have BLE support and no need to start gatt server manually.

When tried on laptop we got list of available devices and when we select a device and click on pair button it shows device not supported error.

Now we are working on why it showing as device not supported error.

🡺The above error was due to compatibility regarding OS version that was installed in our system, we need to have Windows 10 Creators Update. It should display: "Version 1703 (OS Build 15063.413)" or higher.

🡺We tried installing Windows 10 Creators Update OS that was compatible using VMware in windows and Ubuntu but we were not able to do it.

🡺Then we choose Ubuntu as our platform for working on it because it was also having wire shark a tool through which we can capture packets that are being transferred and analyse them.

🡺We installed Ubuntu and then installed Wireshark in it and started to work on Web Bluetooth API. When we tried our examples on Ubuntu machine, we were not able get the values displayed on browser but Wireshark was able to capture the packets.

🡺Then we got to know to that we need to update Bluetooth version to make it support all the gatt features, so we updated Bluetooth version and found it to be working fine.