Building Your Own Internet of Things with JavaScript: From Constrained Device to Secure Gateway

Ziran Sun
ziran.sun@samsung.com
Samsung Research, UK
October, 2018.
1. A Simple IoT demo using JavaScript

2. On Constrained Device
   a. Ultra-Light JS Engine - JerryScript
   b. IoT Platforms on JerryScript

3. On Secure Gateway
   a. The Things Gateway by Mozilla
   b. Web Thing API

4. There are more you can do...
1. A simple IoT demo using JavaScript
A simple IoT demo using JavaScript
Demo video

https://vimeo.com/user90242752/review/293325534/5d519c4bc7
Why JavaScript?

- Popularity
- Maturity
- Productivity

“Any application that can be written in JavaScript will eventually be written in JavaScript.” - Jeff Atwood
2. On Constrained Device
## IoT Hardware [1]

<table>
<thead>
<tr>
<th>IoT Hardware</th>
<th>Memory spec</th>
<th>IoT Hardware</th>
<th>Memory spec</th>
</tr>
</thead>
<tbody>
<tr>
<td>PanStamps</td>
<td>-RAM: 2KB</td>
<td>Nanode</td>
<td>-RAM: 32KB</td>
</tr>
<tr>
<td></td>
<td>-ROM: 33KB</td>
<td></td>
<td>-ROM</td>
</tr>
<tr>
<td></td>
<td>(Flash: 32KB, EEPROM: 1KB)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TinyDuino</td>
<td>-RAM: 2KB</td>
<td>Arduino Yun</td>
<td>-RAM: 2.5KB</td>
</tr>
<tr>
<td></td>
<td>-ROM: 33KB</td>
<td></td>
<td>-ROM: 33KB</td>
</tr>
<tr>
<td></td>
<td>(Flash: 32KB, EEPROM: 1KB)</td>
<td></td>
<td>(Flash: 32KB, EEPROM: 1KB)</td>
</tr>
<tr>
<td>Arduino Uno</td>
<td><strong>-RAM: 2KB</strong></td>
<td>mbed – LPC1768</td>
<td>-RAM: 32KB</td>
</tr>
<tr>
<td></td>
<td><strong>-ROM: 33KB</strong></td>
<td></td>
<td>-ROM: 512KB</td>
</tr>
<tr>
<td></td>
<td>(Flash: 32KB, EEPROM: 1KB)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RFduino</td>
<td>-RAM</td>
<td>Wi-Go Module</td>
<td>-RAM</td>
</tr>
<tr>
<td></td>
<td>-ROM</td>
<td></td>
<td>-ROM: 2MB</td>
</tr>
<tr>
<td>XinoRF</td>
<td>-RAM: 2KB</td>
<td>pcDuino</td>
<td>-RAM: 1GB</td>
</tr>
<tr>
<td></td>
<td>-ROM: 33KB</td>
<td></td>
<td>-ROM: 2GB</td>
</tr>
<tr>
<td></td>
<td>(Flash: 32KB, EEPROM: 1KB)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OpenKontrol Gateway</td>
<td>-RAM: 32KB</td>
<td>OpenPicus Flyport WiFi</td>
<td>-RAM</td>
</tr>
<tr>
<td></td>
<td>-ROM</td>
<td></td>
<td>-ROM: 16MB</td>
</tr>
<tr>
<td>Pinoccio</td>
<td>-RAM: 32KB</td>
<td>Hackberry</td>
<td>-RAM: 512MB/1GB</td>
</tr>
<tr>
<td></td>
<td>-ROM</td>
<td></td>
<td>-ROM: 4GB</td>
</tr>
<tr>
<td>Raspberry Pi</td>
<td><strong>-RAM: 512MB</strong></td>
<td>UDOO</td>
<td>-RAM: 1GB</td>
</tr>
<tr>
<td></td>
<td>-ROM</td>
<td></td>
<td>-ROM</td>
</tr>
<tr>
<td>BeagleBone Black</td>
<td>-RAM: 512MB</td>
<td>Libelium Wasmote</td>
<td>-RAM</td>
</tr>
<tr>
<td></td>
<td>-ROM</td>
<td></td>
<td>-ROM</td>
</tr>
<tr>
<td>CubieBoard</td>
<td>-RAM: 512MB/1GB</td>
<td>The Rascal</td>
<td>-RAM: 64MB</td>
</tr>
<tr>
<td></td>
<td>-ROM</td>
<td></td>
<td>-ROM</td>
</tr>
</tbody>
</table>

(Table credit to Tilmann Scheller)
Device

Blinkt Light

GPIO

IoT.js
JerryScript
Raspbian
Raspberry Pi Zero W
2a. Ultra-light JS Engine
-JerryScript by Samsung
JerryScript

- An ultra lightweight JavaScript engine (<64KB RAM, <200 KB ROM, 160KB binary size compiled for ARM Thumb-2)
- Originally developed from scratch by Samsung
- Transferred to JS Foundation in 2016
- Self-contained and extremely portable
- Mature C API, easy to embed in applications
JerryScript

- Optimization and performance have been top priorities
- Adding new features:
  - Jerry-debugger
  - ES6 related features
  - More hardware and OS supports
- Fast growing community
Hardware and OS Supports (JerryScript Github)

- ESP8266
- ARM MBED
- STM32F4 discovery Board with Nuttx, RIOT
- TizenRT on ARTIK 05x
- OpenWrt
- Particle firmware on Photon board
- Zephyr OS on Arduino 101
- Raspberry Pi 2
JerryScript in Smartwatch [2]
2b. IoT platforms based on JerryScript
IoT JavaScript Platforms based on JerryScript

- **IoT.js by Samsung**
  a. Lightweight version of node.js for constrained devices
  b. Hardware and OS supports are well inline with JerryScript.

- **Zephyr.js by Intel**
  a. JavaScript* Runtime for Zephyr OS
  b. Zephyr.js so far officially supports just 2 boards: Arduino 101 and FRDM-K64F.
  c. Node.js* like APIs
IoT.js Architecture [3]

(Diagram credit to Zoltan Herczeg)
3. On Secure Gateway
3a. Things Gateway
By Mozilla
**Things Gateway Architecture [4]**

Diagram credit to Ben Francis

**Things Cloud**

**Things Gateway**

**Front End** (HTML, CSS & JavaScript)

**Web Thing API** (HTTPS+WS)

**Back End** (NodeJS)

**Adapter API** (IPC)

**Adapter (Rust)**

**Adapter (NodeJS)**

**Serial**

**Linux**

**Project Things**

(Diagram credit to Ben Francis)
Things Gateway - Security

- **HTTPS via mozilla-iot.org tunnelling service**
  a. Allow setting up a secure subdomains with LetsEncrypt TLS certificates
  b. TCP tunnel uses PageKite from Mozilla cloud server to Gateway
- JSON Web Tokens (JWT) used for authentication
- OAuth to authorise third party apps & services
3b. Web Thing API
By Mozilla
Web of Things

- Creates Standards to Reduce IoT Fragmentation and make things **interoperable**.
- Addresses Things via URLs on the web to make things **linkable** and **discoverable**
- Reuses existing and well-known web standards and web technologies
Web Thing API

- Web Thing Description
- Web Things REST API
- Web Thing WebSocket
Mozilla Web Thing API

- Things Description

```json
{
    "name": "Blinkt! light",
    "type": "thing",
    "Description": "A Blinkt! light",
    "properties": {
        "on": {
            "type": "boolean",
            "description": "Whether the Blinkt! light is turned on",
            "href": "/things/blinkt/properties/on"
        }
    },
    "actions": {
        "toggle": {
            "description": "Turn the Blinkt! Light on and off"
        }
    }
}
```
Web Thing REST API (Get/Put/Delete/Post)

Request

```
PUT
/things/blinkt/properties/on
{
  "on": true
}
```

Response

```
200 OK
{
  "on": true
}
```
4. There are more you can do...
Example 1: An End to End Demo Built with WoT [5]

https://sosg.mozilla-iot.org

......................................................

<table>
<thead>
<tr>
<th>Fan</th>
<th>○</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual On/Off Switch</td>
<td>1</td>
</tr>
<tr>
<td>Virtual Binary Sensor</td>
<td>○</td>
</tr>
<tr>
<td>Virtual Multi-level Sensor</td>
<td>0</td>
</tr>
<tr>
<td>Ambient Light Sensor</td>
<td>ON</td>
</tr>
<tr>
<td>Temperature Sensor</td>
<td>31</td>
</tr>
<tr>
<td>GpioSensorExample</td>
<td>○</td>
</tr>
</tbody>
</table>

PWA
References:

References:


Thank You!