



Distributed Security Risks and Opportunities in the W3C Web of Things

Michael McCool (presenting) and Elena Reshetova

NDSS DISS 2018
February 18, 2018
San Diego

Outline

Goals

W3C Web of Things

Risks and Opportunities

1. Local Links
2. Vulnerability Analysis
3. Endpoint Adaptation
4. Secure Discovery
5. Distributed Security

Summary and Conclusions

Goals

Why this paper?

- Necessary to perform security review of standards under development
- Paper lists a number of **problems** with the proposed W3C Web of Things standard under development that need to be addressed
- The paper does **not**, *generally*, propose **solutions**

Desired outcome:

Discussion, collaboration, and research to find solutions to these problems.

W3C[®] Web of Things

Working Group within W3C chartered December 2016

- Based on ongoing work in an Interest Group by the same name

Target date of December 2018 to deliver specifications for

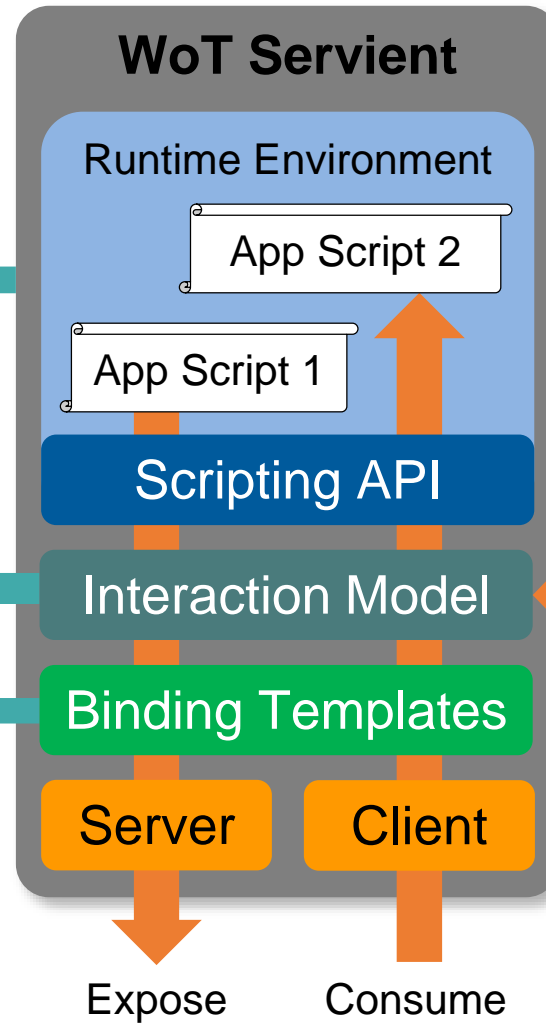
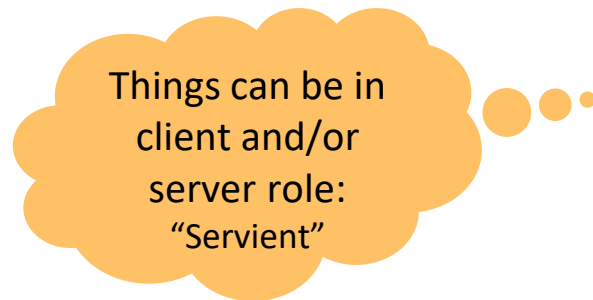
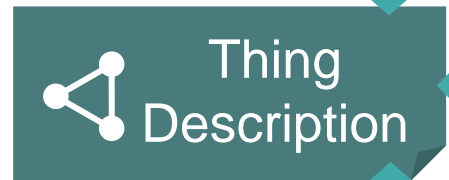
- **Thing Description:** metadata for WoT Things
- **Scripting API:** standardized mechanism to consume and expose Thing Descriptions and program the behaviour of Things
- **Protocol Bindings:** mappings of WoT architecture to various concrete protocols: HTTP, CoAP, MQTT, etc.

W3C[®] Web of Things: Resources and Links

- W3C: World Wide Web Consortium: <https://www.w3.org>
- Web of Things Interest Group: <https://www.w3.org/WoT/IG/>
 - Charter: Leverage web standards and technology to enable IoT interoperation
 - Web architecture: <https://www.w3.org/standards/webarch/>
- Web of Things Working Group in the W3C to develop standard recommendations:
 - <https://www.w3.org/2016/09/wot-wg-charter.html>
 - Co-chairs: Matthias Kovatsch (Siemens), Kazuo Kajimoto (Panasonic), Michael McCool (Intel)
 - White paper on WoT architecture: <http://w3c.github.io/wot/charters/wot-white-paper-2016.html>
- WoT current practices: <http://w3c.github.io/wot/current-practices/wot-practices.html>

W3C[®] WoT: Deliverables/Architecture

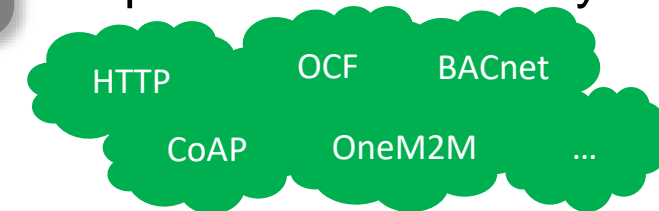
1. WoT Thing Description (TD)
with simple interaction model



3. WoT Scripting API
for a browser-like
runtime environment

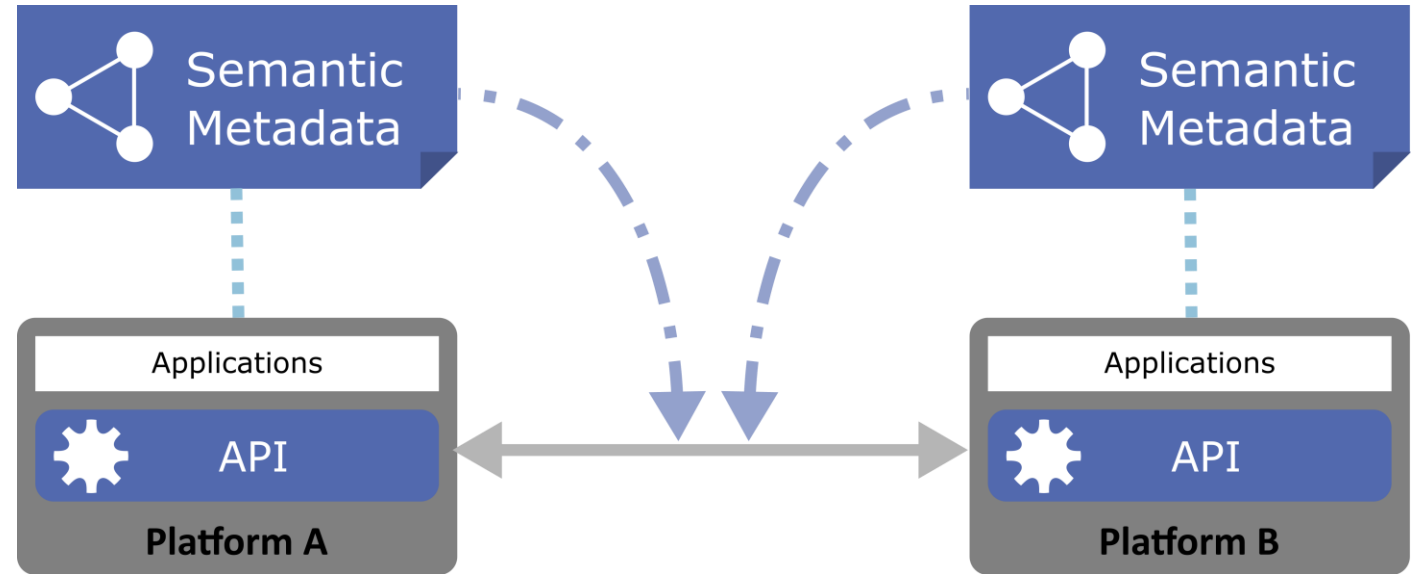
Local Hardware

2. WoT Binding Templates
to connect to different
platforms and ecosystems



Standardized Metadata

- Protocol-independent description of network APIs
- Communication and security requirements
- Data models and constraints
- Semantic annotation



Thing Description Example

```
{
  "@context": [
    "http://w3c.github.io/wot/w3c-wot-td-context.jsonld",
    { "domain": "http://example.org/actuator#" }
  ],
  "@type": "Thing",
  "name": "MyLEDThing",
  "base": "coap://myled.example.com:5683/",
  "security": {
    "cat": "token:jwt",
    "alg": "HS256",
    "as": "https://authority-issuing.example.org"
  },
  "interaction": [
    {
      "@type": ["Property", "domain:onOffStatus"],
      "name": "status",
      "outputData": {"valueType": {"type": "boolean"}},
      "writable": true,
      "links": [
        {
          "href": "nwn"
```

JSON-LD
(Linked Data)

W3C WoT TD
vocabulary

domain-specific
vocabulary

JSON Schema


```

"interaction": [
  {
    "@type": ["Property", "domain:onOffStatus"],
    "name": "status",
    "outputData": {"valueType": {"type": "boolean"}},
    "writable": true,
    "links": [
      {
        "href": "pwr",
        "mediaType": "application/exi"
      },
      {
        "href": "http://mytemp.example.com:8080/status",
        "mediaType": "application/json"
      }
    ]
  },
  {
    "@type": ["Action", "domain:fadeIn"],
    "name": "fadeIn",
    "inputData": {
      "valueType": {"type": "integer"},
      "domain:unit": "domain:ms"
    },
    "links": [
      {
        "href": "in",
        "mediaType": "application/exi"
      }
    ]
  }
]

```

Property

Action

```

    inputData : {
      "valueType": {"type": "integer"},
      "domain:unit": "domain:ms"
    },
    "links": [
      {
        "href": "out",
        "mediaType": "application/exi"
      },
      {
        "href": "http://mytemp.example.com:8080/out",
        "mediaType": "application/json"
      }
    ]
  },
  {
    "@type": ["Event", "domain:alert"],
    "name": "criticalCondition",
    "outputData": {"valueType": {"type": "string"}},
    "links": [
      {
        "href": "ev",
        "mediaType": "application/exi"
      }
    ]
  }
]
}

```

} Event
(sources, sinks, ...)

Problem 1: Local Links

Risks

- WoT is predicated on Web standards being useful for IoT
- However, the Web is oriented towards browsers and human-readable information, whereas the IoT includes many machine-to-machine communications
- Web technologies generally also assume an active full internet connection. IoT devices may only have local network connectivity

Major pain point:

- Browser assumptions about certificate revocation checking under HTTPS
- Primarily affects use of HTTPS for “local” user interfaces

Problem 2: Vulnerability Analysis

Risks

- Pervasive metadata allows attacker to analyze a system in detail to find vulnerabilities and plan an attack

Opportunity

- Pervasive metadata allows a system owner to analyze a system in detail to find vulnerabilities and prevent attacks

Problem 3: Endpoint Adaptation

Risks

- Protocol conversion bridges are vulnerable to attack, and protocol conversion may require “unpacking” data in flight, making it available to interception

Opportunity

- A system wishing to talk to a WoT Thing can access the metadata for a thing and set up an end-to-end encrypted channel directly to that thing, bypassing multiple translation steps

Problem 4: Semantic Discovery

Risks

- Semantic search is relatively expensive
 - Pathological semantic queries can be created that can consume an unreasonable amount of resources
- If semantic discovery services are “open”, then they will be subject to denial of service attacks

Opportunity

- Semantic discovery is a powerful capability we would like to make available to users

Further Issue: Metadata Protection

- TODO: where do we discuss protection of the metadata: authorized access, authentication, etc?

Problem 5: Distributed Security

Opportunity

- A Thing Description can provide information that can be useful for enabling distributed security mechanisms
- How can we make validated and authenticated Thing Descriptions available in a distributed fashion?
- What distributed security mechanisms should we support and what information do they need?

Examples:

- Payments/deposits for services: Interledger address
- Web of Trust: References
- Access Control: Management Thing API
- Discovery: Distributed discovery services and Thing Directories
- Caching and proxies: encrypted state

Example

- Garage Door opener...
- TODO: Show each of the issues in context

Summary

Main W3C WoT deliverable and differentiator:

- Universal metadata format (“Thing Description”) for IoT services (“Things”) and associated common Thing abstraction

Use of Web Standards for IoT has specific issues:

- Local links, HTTPS, and certificates

Use of semantic metadata has specific risks and opportunities:

- Vulnerability analysis
- Endpoint adaptation vs. link-by-link translation
- Preventing denial-of-service attacks on semantic discovery services

Follow-up Actions

- Plugfest – Prague, March 24-25, 2018
- Interest Group
- Working Group
 - W3C Web of Things Security and Privacy Considerations
- Collaborations
- Security Validation



Web of Things: Interest Group Members

