Implementation report

draft-reddy-dots-signal-channel-10

March 2017

Presenter : Flemming Andreasen
Proof of concept

• Used californium framework ([https://eclipse.org/californium/](https://eclipse.org/californium/)) to exchange DOTS messages using CoAP over DTLS.
• Californium implements DTLS 1.2
Proof of concept

• Californium framework provides APIs to
  ➢ Use congestion control
  ➢ Configuration of message transmission parameters and heartbeat timeout
  ➢ Mark messages as Confirmable or Non-confirmable
Proof of concept

Proof of concept

Constrained Application Protocol, Non-Confirmable, PUT, MID:2719

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>1</td>
</tr>
<tr>
<td>Type</td>
<td>Non-Confirmable (1)</td>
</tr>
<tr>
<td>Token Length</td>
<td>8</td>
</tr>
<tr>
<td>Code</td>
<td>PUT (3)</td>
</tr>
<tr>
<td>Message ID</td>
<td>2719</td>
</tr>
<tr>
<td>Token</td>
<td>4ee7d4f8f01595e6</td>
</tr>
<tr>
<td>Opt #1: Uri-Host</td>
<td><a href="http://www.example.com">www.example.com</a></td>
</tr>
<tr>
<td>Opt #2: Uri-Path</td>
<td>.well-known</td>
</tr>
<tr>
<td>Opt #3: Uri-Path</td>
<td>v1</td>
</tr>
<tr>
<td>Opt #4: Uri-Path</td>
<td>DOTS-signal</td>
</tr>
<tr>
<td>Opt #5: Uri-Path</td>
<td>signal</td>
</tr>
<tr>
<td>Opt #6: Content-Format</td>
<td>application/cbor</td>
</tr>
</tbody>
</table>

End of options marker: 255

[Response In: 339]

Payload: Payload Content-Format: application/cbor, Length: 27

Payload Desc: application/cbor

Concise Binary Object Representation

Map: (1 entries)

...0 0001 = Unsigned Integer: 1

Map: (2 entries)

...0 0010 = Unsigned Integer: 2

Unsigned Integer: 123344

...0 1011 = Unsigned Integer: 11

Array: (2 elements)

Text String: Server1

Text String: Server2
Proof of concept (http://cbor.me/)

```
{  
  1: {  
    2:99,  
    11:[
      "Server1",
      "Server2"
    ]  
  }  
}
```

```
null  
```

```
24 Bytes
```

```
a1
  01  # map(1)
  a2  # map(2)
    
    02  # unsigned(2)
    0b  # unsigned(99)
    82  # unsigned(11)
    67  # array(2)
      67  # text(7)
        53657276657231  # "Server1"
      67  # text(7)
        53657276657232  # "Server2"
```