IETF Protocols and IoT

https://jaime.win/lecture

Jaime Jiménez jaime.jimenez@ericsson.com

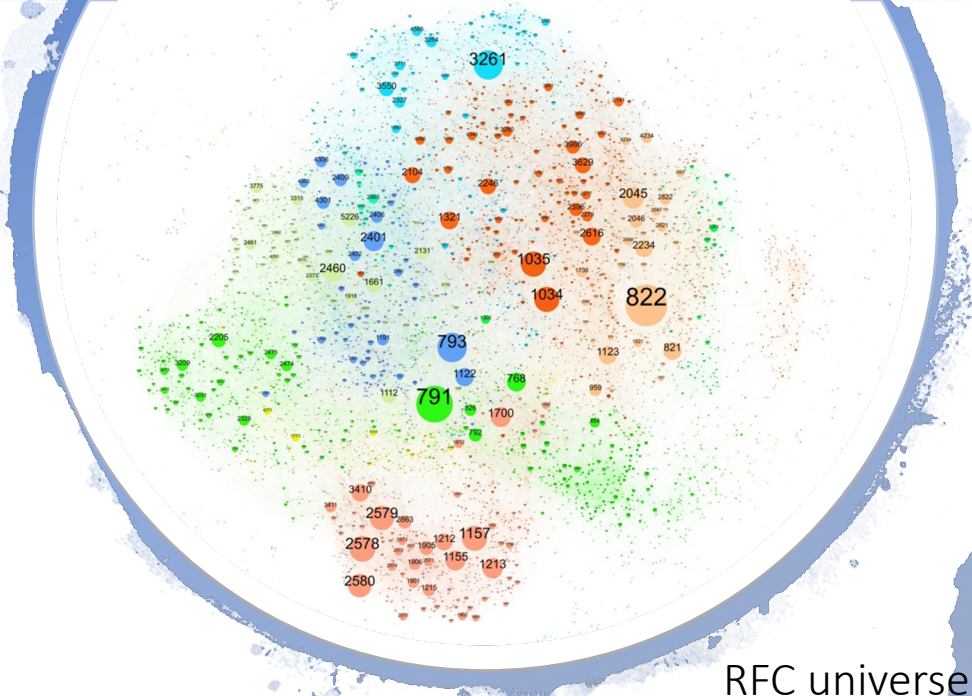
@jaim

10th April 2019

Table of Contents

Internet Engineering Task Force (IETF)

The Internet



RFC universe

courtesy of [Peter Krantz]

Internet Map

Evolved from US ARPANET back in 1986.

1998	50 million users	25 million servers
2018	3.4 billion users	20 billion devices

3 billion unique IPv4 addresses today.
Billions of IPv6 devices tomorrow.

The IETF mission

- IETF's role: Specify the underlying, fundamental Internet technologies
- "Permissionless innovation" - others can build on top - unlike "App Stores" or Telco services.
- RFC3935: "The goal of the IETF is to make the Internet work better."
- Some well known achievements …
  - RFC791 The Internet Protocol.
  - RFC792 The Internet Control Message Protocol.
  - RFC821 The Simple Mail Transfer Protocol.
  - RFC768 User Datagram Protocol.
  - RFC959 The File Transfer Protocol.
  - RFC793 The Transmission Control Protocol.
  - RFC854 Telnet Specification.
  - RFC1119 Network Time Protocol.
  - RFC1157 A Simple Network Management Protocol.
  - RFC1035 Domain names - implementation and specification.
  - RFC1945 Hypertext Transfer Protocol.
  - RFC2131 Dynamic Host Configuration Protocol.
  - RFC3261 The Session Initiation Protocol.
  - RFC6455 The WebSocket Protocol.
  - RFC5321 Simple Mail Transfer Protocol.
  - RFC7540 Hypertext Transfer Protocol Version 2.
  - RFC6749 The OAuth 2.0 Authorization Framework.
  - RFC4271 The Border Gateway Protocol.
  - RFC4287 The Atom Syndication Format.
  - RFC4251 The Secure Shell (SSH) Protocol Architecture.
  - RFC8200 Internet Protocol, Version 6 (IPv6) Sepcification.

”Rough consensus and running code"

The IETF Organization

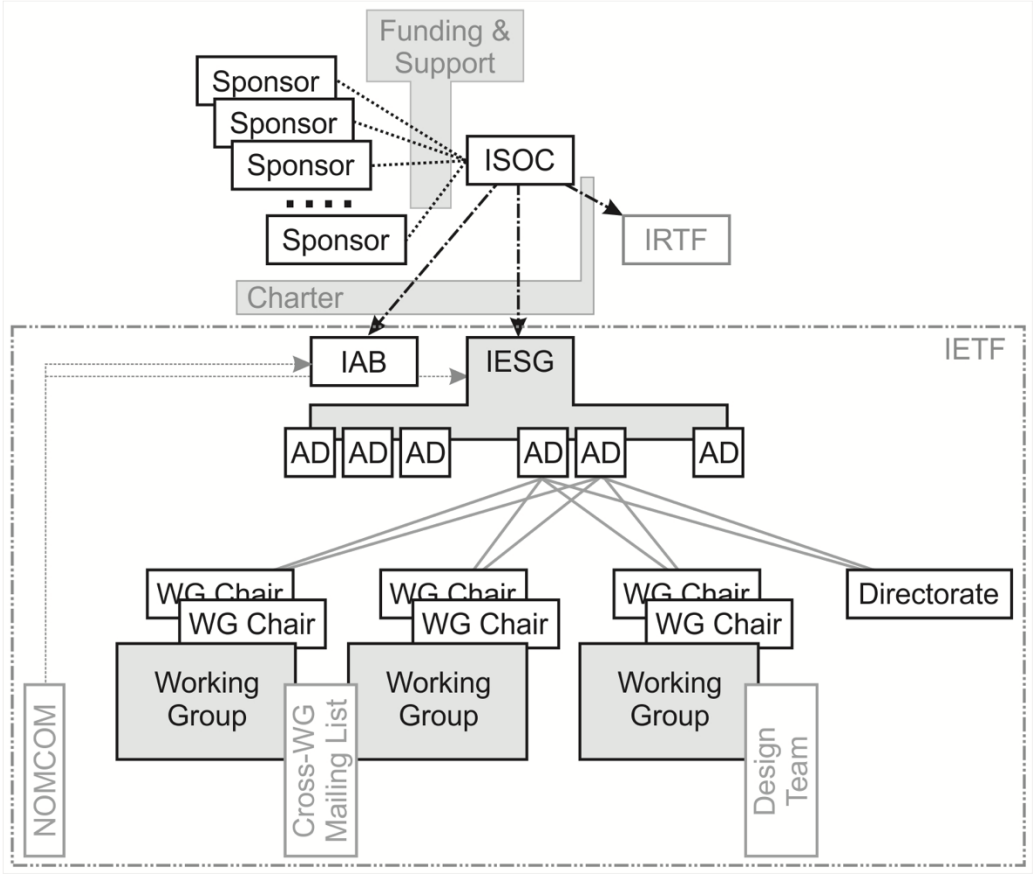
Technical work of the IETF is done in its Working Groups, which are organized by topic into several areas (e.g., routing, transport, security, etc.).

IETF working groups are grouped into areas, and managed by Area Directors.

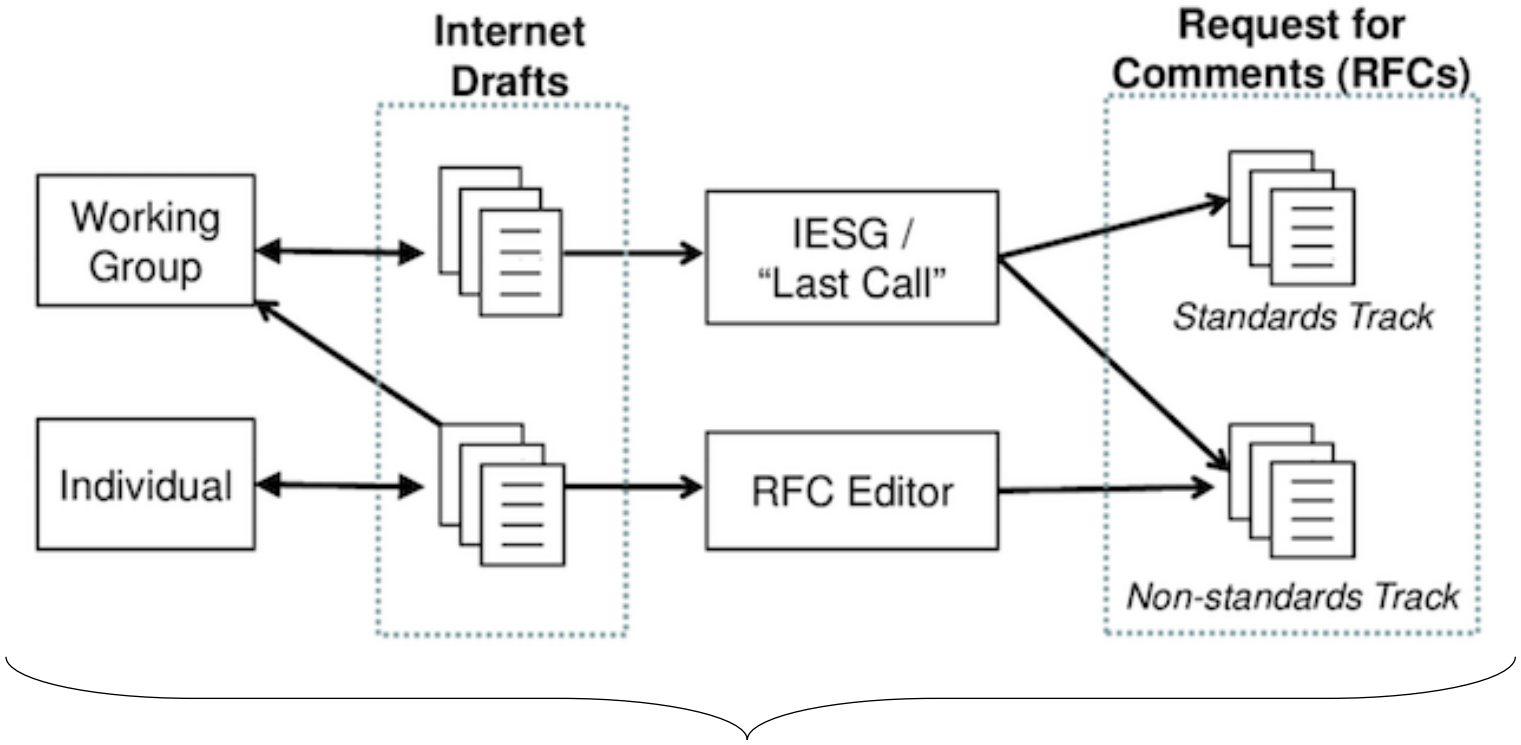
The ADs are members of the Internet Engineering Steering Group (IESG)

The Internet Architecture Board, (IAB) is responsible for defining the overall architecture of the Internet, providing guidance and broad direction to the IETF.

The IAB also serves as the technology advisory group to the Internet Society (ISOC).



A draft's lifetime



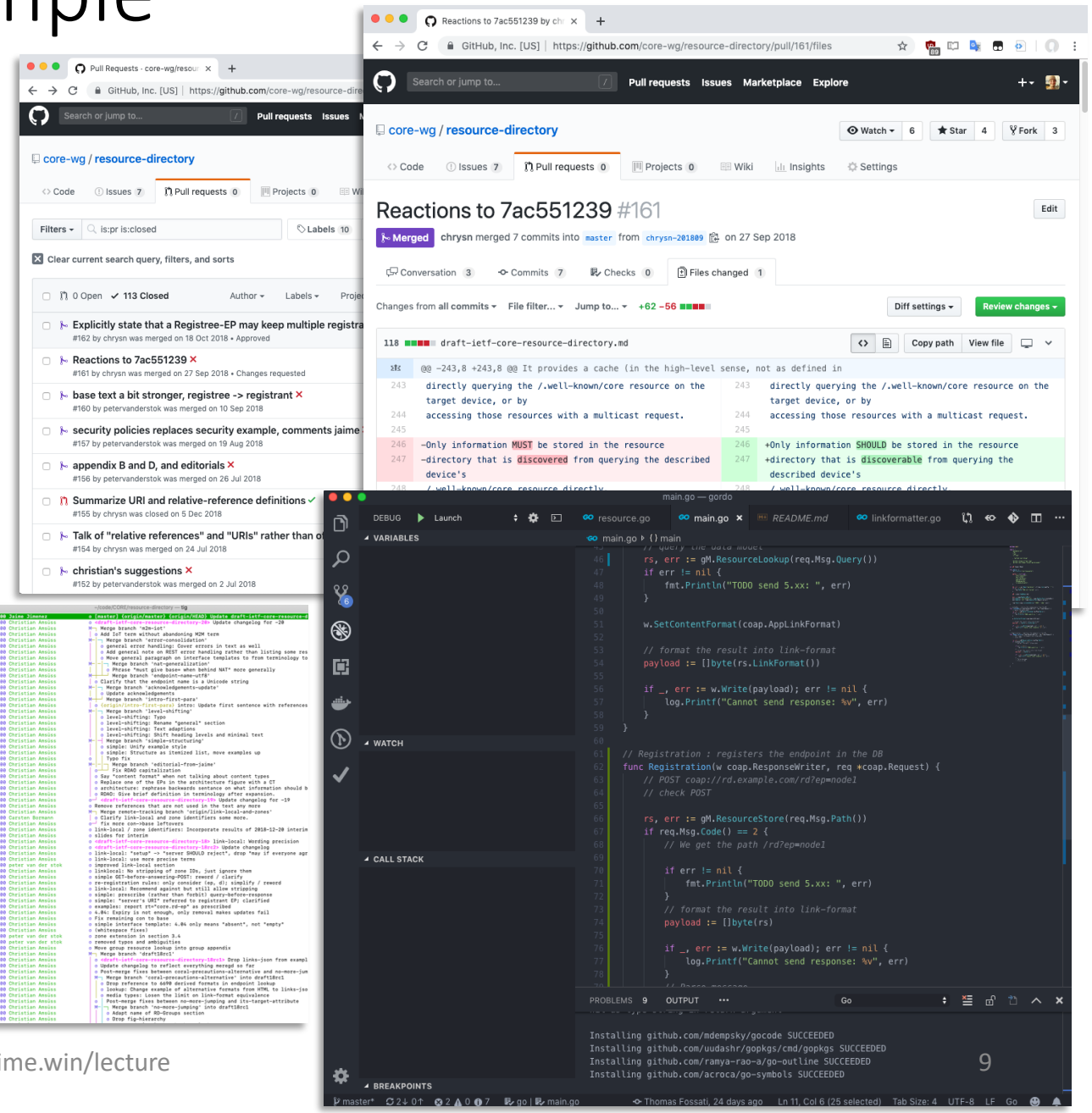
N Years
"Consensus is time consuming"

Ways of working: RD Example

Working on the draft
https://github.com/core-wg/resource-directory

Working on the code
https://github.com/thomas-fossati/gordo

Other examples
https://github.com/Ell-i/coap-rd
https://github.com/nning/core-rd



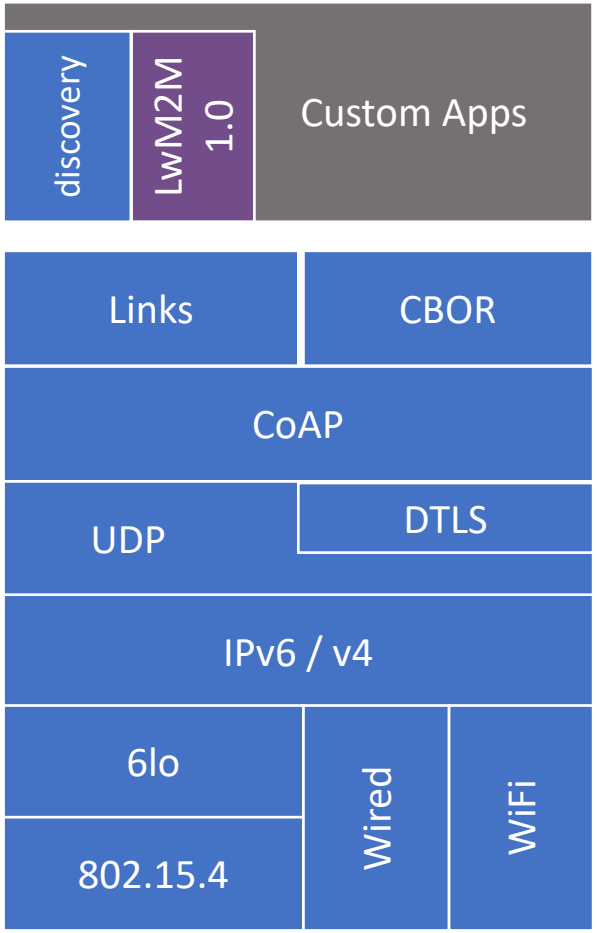
Constrained Application Protocol (CoAP)

IETF: dozen+ years of IoT standards

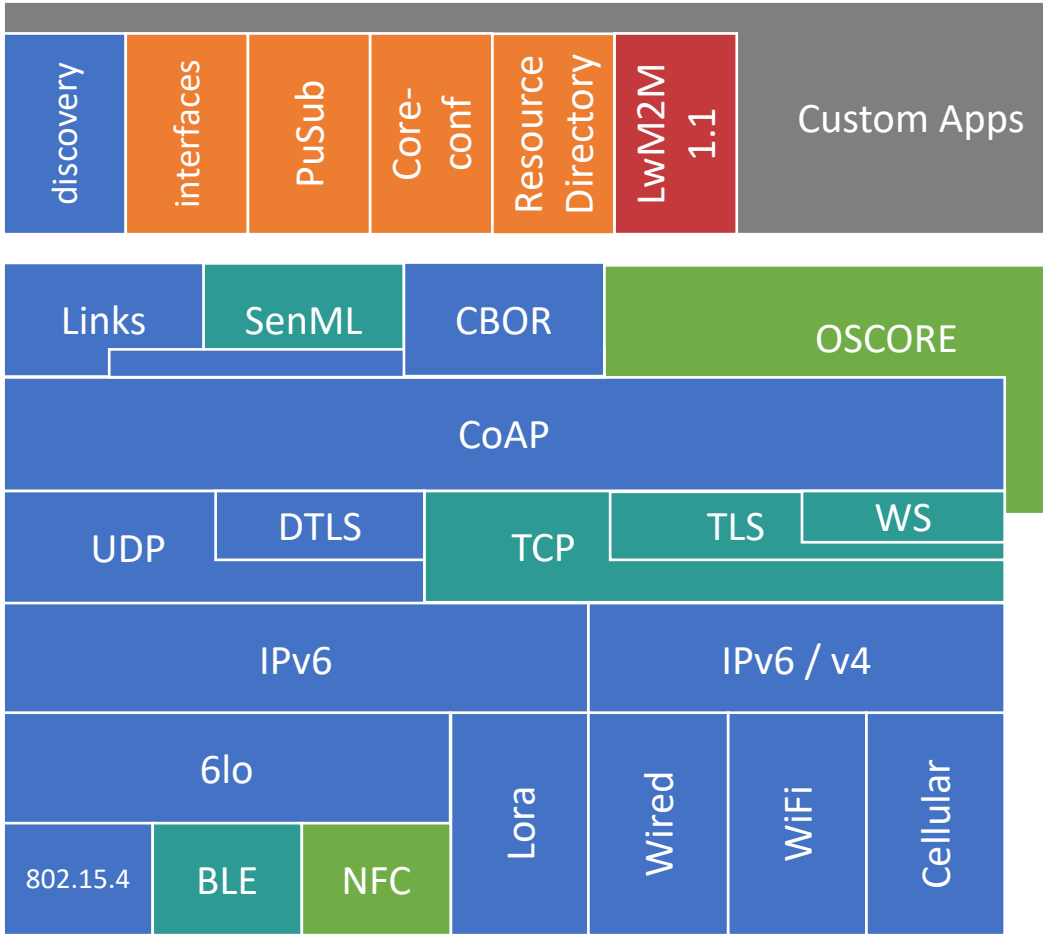
RFC 2689	RFC 3485	RFC 3544	RFC 3819	RFC 3940	RFC 3941	RFC 4629
RFC 4919	RFC 4944	RFC 5049	RFC 5401	RFC 5740	RFC 5856	RFC 5857
RFC 5858	RFC 6282	RFC 6469	RFC 6568	RFC 6606	RFC 6775	RFC 6690
RFC 7049	RFC 7228	RFC 7252	RFC 7388	RFC 7390	RFC 7400	RFC 7641
RFC 7668	RFC 7744	RFC 7925	RFC 7959	RFC 8075	RFC 8132	RFC 8152
RFC 8307	RFC 8323	RFC 8376	RFC 8392	RFC 8424	RFC 8516	…and more

Connectivity WGs Application WGs Security WGs

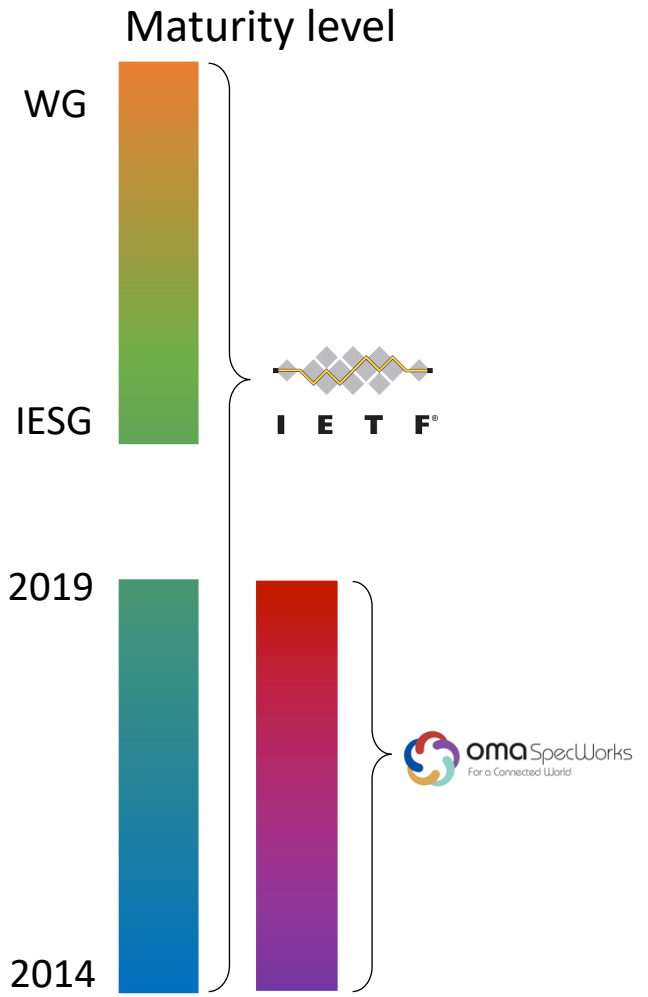
Standards Device Stack



2014

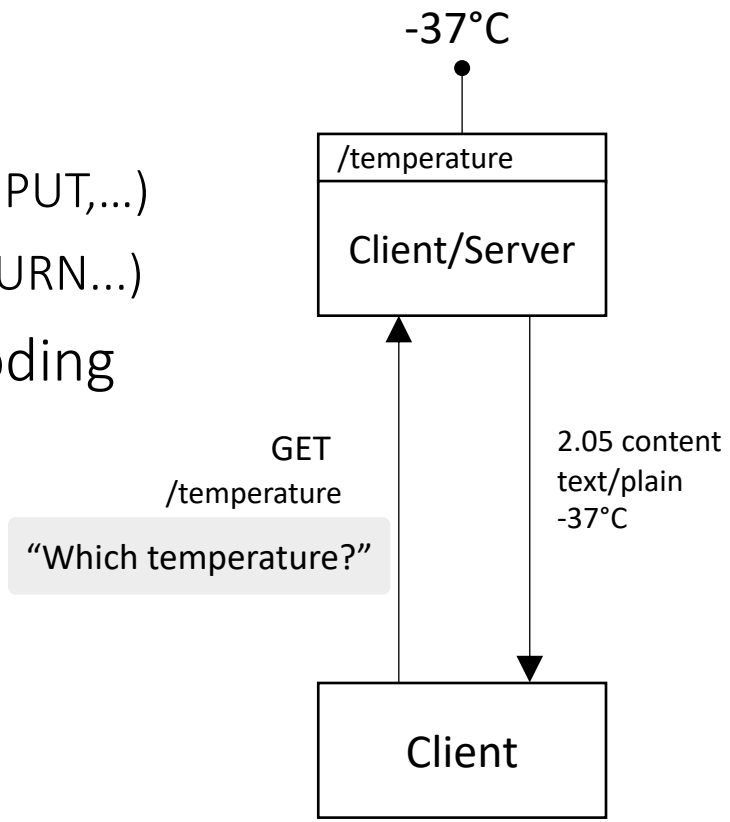


2019



The **Co**nstrained **A**pplication **P**rotocol (**CoAP**)

- CoAP (RFC7252) implements HTTP's **REST** model
  - Simple devices: 100 to 250 KiB code and 10 to 50 KiB RAM
  - Each device can be client and server exposing resources
  - CoAP defines methods to access those resources (GET, POST, PUT,...)
  - Same key concepts borrowed from HTTP (Media types, URL, URN...)
- Has a compact 4-byte header, with simple options encoding
- Simple protocol, datagram (UDP, DTLS)
  - Reliability through header message type "*CON/NON*"
  - With TCP/TLS (RFC8323) support for NAT-ed environments
- The Resource Directory provides a directory service

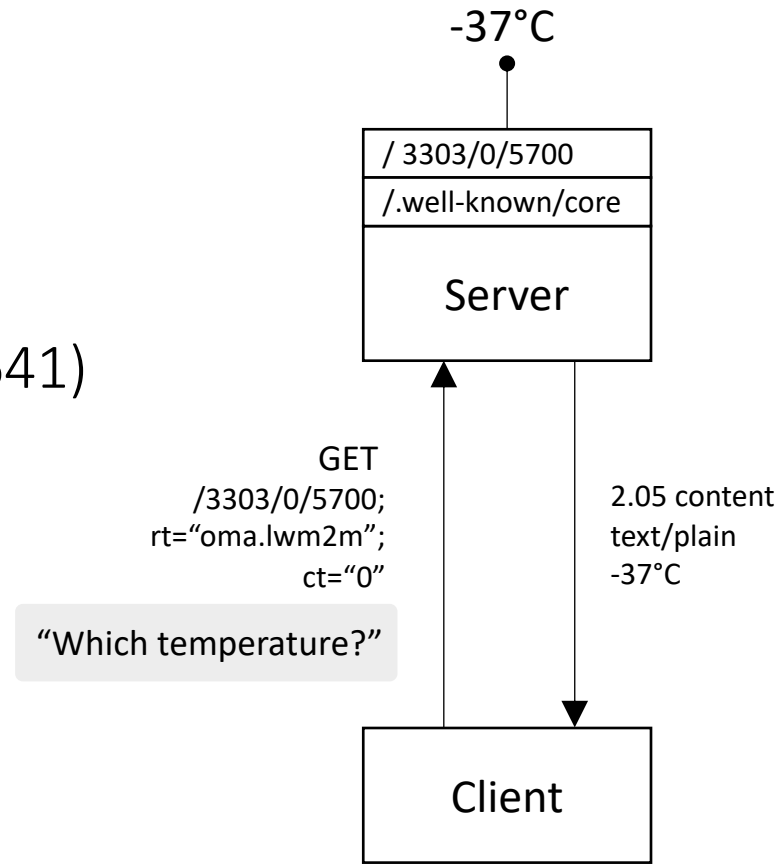


The Constrained Application Protocol (CoAP)

- CoRELink (RFC6690) provides a link format
  - Reuses Web Linking RFC5988 for IoT.
  - Enables query parameters for discovery (lt, gt…)
  - Enables attribute and relation types (rt, if, sz).

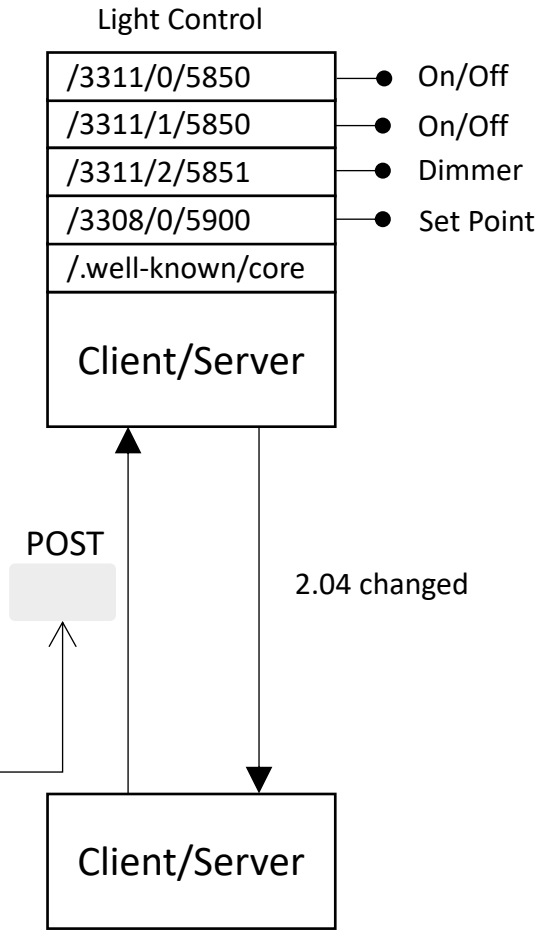
    `<3303/0/5700>;rt="oma:lwm2m:temp";ct="0"`
- Notifications available through *observe* option (RFC7641)
  - Can observe and add query parameters to the observation

    `<3303/0/5700?lt=0>`
- The *"/.well-known/core"* URI provides discovery
- Multiple serialization formats used with CoAP
  - SenML (RFC8428): Minimalistic JSON
  - CBOR (RFC7049): Binary serialization
- Multiple implementations available at coap.technology



Example Actuation

- When using IPSO actuation is handled with executable (E) resources as well as readable and writeable ones (RW).
  - Parameters needed for actuation are passed along on resources.
  - Type and range of values are known to client based on the schema.
  - Actuation can use executable and writeable (EW) resources.



A server Write-Composite to switch off 2 light sources, dim a 3rd to 20% and set the thermostat to 18 degrees will have a JSON payload as shown in table below. Lights are all controlled by instances of IPSO Light Control Object (Object ID 3311), while thermostat is controlled by an instance of IPSO object Set Point (Object ID 3308).

```
[{"n":"/3311/0/5850", "vb":false},
{"n":"/3311/1/5850", "vb":false},
{"n":"/3311/2/5851", "v":20},
{"n":"/3308/0/5900", "v":18}]
```

Serialization Formats

SenML - JSON

```
[{"bn":"/3/0/","n":"0","vs":"Open
Mobile Alliance"},
{"n":"1","vs":"Lightweight M2M
Client"},
{"n":"2","vs":"345000123"},
{"n":"3","vs":"1.0"},
{"n":"6/0","v":1},
{"n":"6/1","v":5},
{"n":"7/0","v":3800},
{"n":"7/1","v":5000},
{"n":"8/0","v":125},
{"n":"8/1","v":900},
{"n":"9","v":100},
{"n":"10","v":15},
{"n":"11/0","v":0},
{"n":"13","v":1367491215},
{"n":"14","vs":"+02:00"},
{"n":"16","vs":"U"}]
```

SenML-CBOR

```
90 a3 21 65 2f 33 2f 30 2f 00 61 30
03 74 4f 70 65 6e 20 4d 6f 62 69 6c
65 20 41 6c 6c 69 61 6e 63 65 a2 00
61 31 03 76 4c 69 67 68 74 77 65 69
67 68 74 20 4d 32 4d 20 43 6c 69 65
6e 74 a2 00 61 32 03 69 33 34 35 30
30 30 31 32 33 a2 00 61 33 03 63 31
2e 30 a2 00 63 36 2f 30 02 01 a2 00
63 36 2f 31 02 05 a2 00 63 37 2f 30
02 19 0e d8 a2 00 63 37 2f 31 02 19
13 88 a2 00 63 38 2f 30 02 18 7d a2
00 63 38 2f 31 02 19 03 84 a2 00 61
39 02 18 64 a2 00 62 31 30 02 0f a2
00 64 31 31 2f 30 02 00 a2 00 62 31
33 02 1a 51 82 42 8f a2 00 62 31 34
03 66 2b 30 32 3a 30 30 a2 00 62 31
36 03 61 55
```

SenML-CBOR diagnostic

```
[{-2: "/3/0/", 0: "0", 3: "Open
Mobile Alliance"}, {0: "1", 3:
"Lightweight M2M Client"},
{0: "2", 3: "345000123"},
{0: "3", 3: "1.0"},
{0: "6/0", 2: 1},
{0: "6/1", 2: 5},
{0: "7/0", 2: 3800},
{0: "7/1", 2: 5000},
{0: "8/0", 2: 125},
{0: "8/1", 2: 900},
{0: "9", 2: 100},
{0: "10", 2: 15},
{0: "11/0", 2: 0},
{0: "13", 2: 1367491215},
{0: "14", 3: "+02:00"},
{0: "16", 3: "U"}]
```

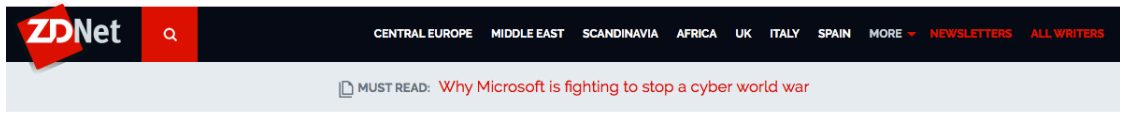
Distributed Denial of Service (DDOS) and CoAP

Record-breaking attacks

[1]

Devices infected by Mirai continuously scan the internet for the IP address of Internet of things (IoT) devices. Mirai includes a table of IP Address ranges that it will not infect, including private networks and addresses allocated to the United States Postal Service and Department of Defense.[14]

Mirai then identifies vulnerable IoT devices using a table of more than 60 common factory default usernames and passwords, and logs into them to infect them with the Mirai malware.[6][15][16] Infected devices will continue to function normally, except for occasional sluggishness,[15] and an increased use of bandwidth. A device remains

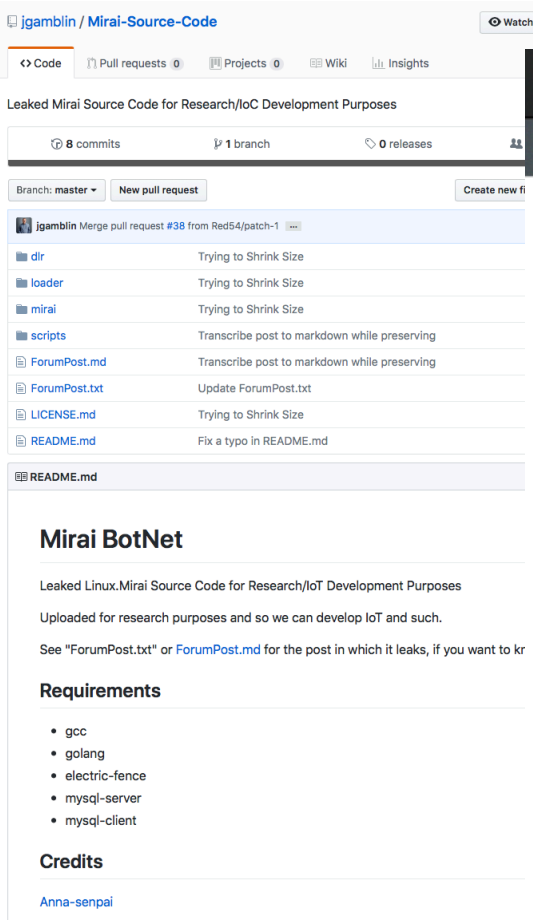


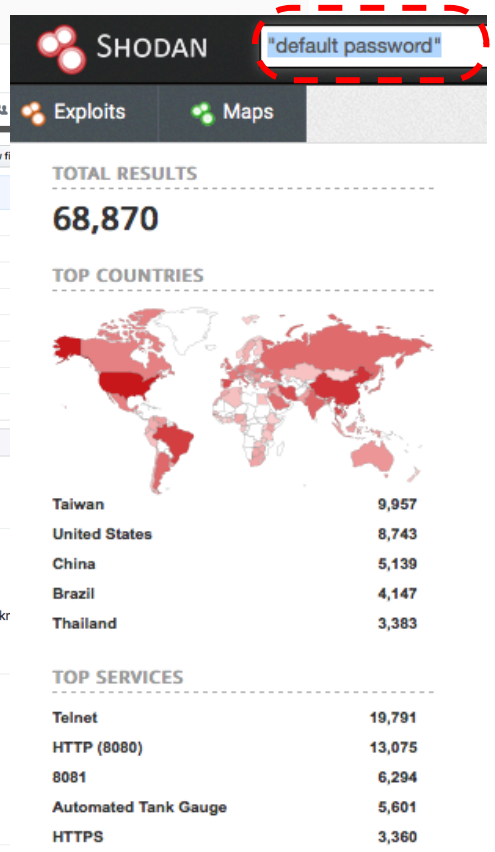
[2]

The CoAP protocol is the next big thing for DDoS attacks

CoAP DDoS attacks have already been detected in the wild, some clocking at 320Gbps.

By Catalin Cimpanu for Zero Day | December 5, 2018 -- 04:13 GMT (04:13 GMT) | Topic: Security

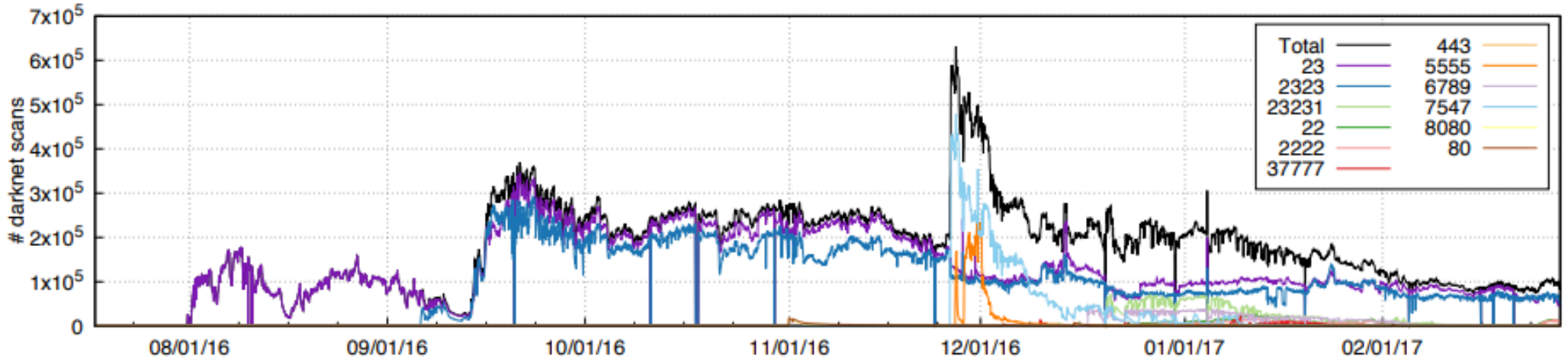




[1] https://en.wikipedia.org/wiki/Mirai_(malware)
[2] https://www.zdnet.com/article/the-coap-protocol-is-the-next-big-thing-for-ddos-attacks/

DDOS

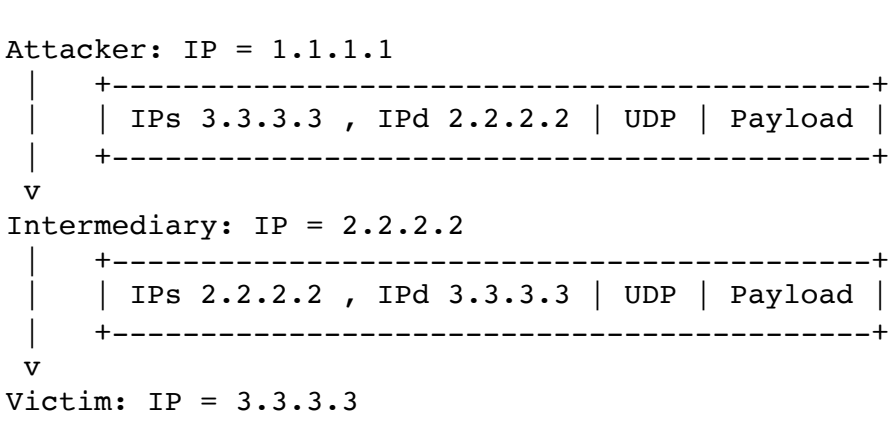
- The purpose of any denial-of-service attack is to limit the availability of a service.
- Distributed denial-of-service (DDoS) attacks came to public notice when they were used to bring down Panix, a New York ISP, for several days in 1996.
- Over time they improved:
  - 2007 hit 6 out of the 13 ROOT DNS servers back.
  - 2016 Mirai's botnet, used to perform very large (600Gps) DDOS attacks using an average of 300.000 infected devices (i.e., DVRs, IP cameras, routers, and printers) and as many as 600000.
- DDOS is used to blackmail, DDOS can be hired as cheap as 15€ a month.



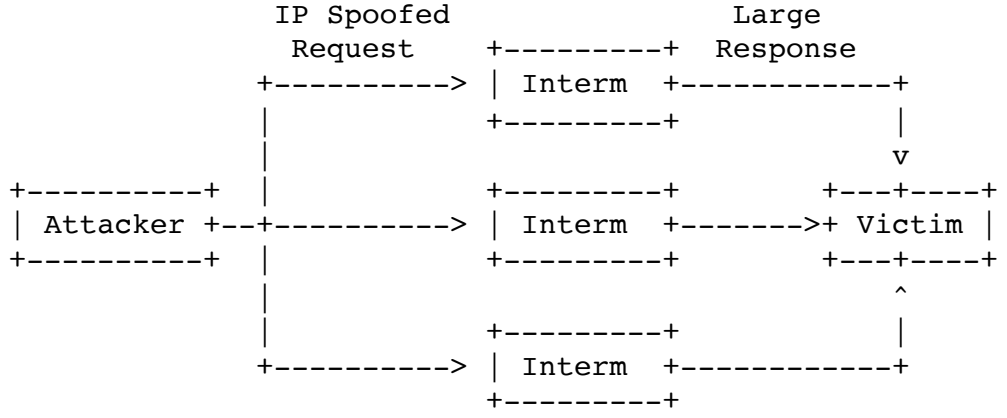
UDP is part of the (hard to solve) problem

- UDP is a simple, connection-less transport protocol.
  - No handshaking
  - No dedicated end-to-end connection
  - Has multicast
- Useful in the IoT context *and* in DDOS
- Two types of attacks

1. IP Address Spoofing



2. Packet Amplification



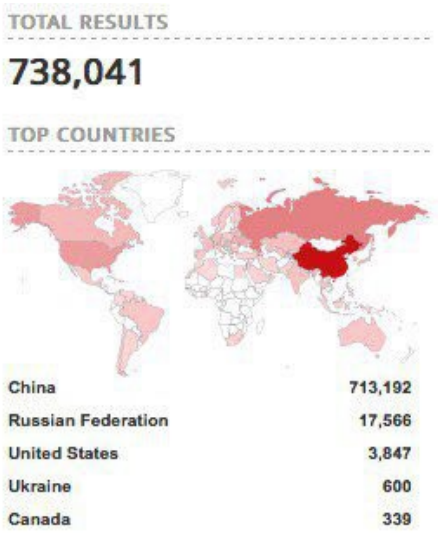
Amplification = number-of-devices * (1 MBps * Amplification Factor)

Enter CoAP

- RFC7252 Security Section already warned about these attacks.
- As CoAP becomes more mainstream, there will be more attacks.
- CoAP has already been used already reaching *55Gbps on average*.
- Number of visible CoAP endpoints increased rapidly
  from 6500, to 220000, to 738041 and down to 436854 after patching.
- CoAP has a substantial Amplification Factor, and a huge number of potential endpoints.

https://www.shodan.io/search?query=port%3A5683

Protocol (port)	Request size	Avg/Max amplification	Numbers
DNS (53)	37 b	28/54	13.986.243
NetBIOS (137)	50 b	3/229	601.869
SIP (5060)	128 b	3/19	15.161.110
SNPM (161)	40 b	34/553	2.184.406
NTP (123)	12 b	22/198	9.483.324
CoAP (5683)	**21 b**	**16/97**	**436.854**
LDAP (389)	52 b	45/55	494.276
Memcached (11211)	15 b	73/51000 (!!)	39.785



Some solutions

- Avoid NoSec mode, using instead any of the other secure modes.
- Avoid exposing CoAP endpoints to the wider Internet if they don't need to.
- Using a randomized token value.
- When using CON messages we can detect unexpected ACK and RST from the deceived endpoint.
- Verify every now and then that observations are still valid by sending a CON from the observed CoAP resource.
- Limiting the response rate can also palliate the problem.
- Routers and middleboxes have high bandwidth, which is a scarce resource in the case of constrained networks. That will be a mitigating factor, making CoAP nodes less attractive for this attack.
- Large amplification factors should not be provided if the request is not authenticated.
- CoAP servers should not accept multicast requests that can not be authenticated in some way
- Ensure all default passwords are changed to strong passwords.
- Update IoT devices with security patches as soon as patches become available.
- Network operators could prevent the leakage of packets with forged source addresses by following the guidelines of RFC2827.

https://jaime.win/lecture

http://jaime.win/slides/upm2019.pdf