

# IETF 106 TEEP Hackathon Report

Akira Tsukamoto, Nov. 19, 2019

# What we planned

- Open Trust Protocol:
  - Evaluate OTrPv1 vs TEEP (aka OTrPv2) proposal
  - Test implementations of OTrP-over-HTTP
    - draft-ietf-teep-otrp-over-http-02
- Brought prototypes of TAM and TEEP device
  - TAM with node-js by Isobe-san
  - TAM with SGX by Dave Thaler
  - TEEP device on OP-TEE by Akira Tsukamoto
  - TEEP device on SGX by Dave Thaler

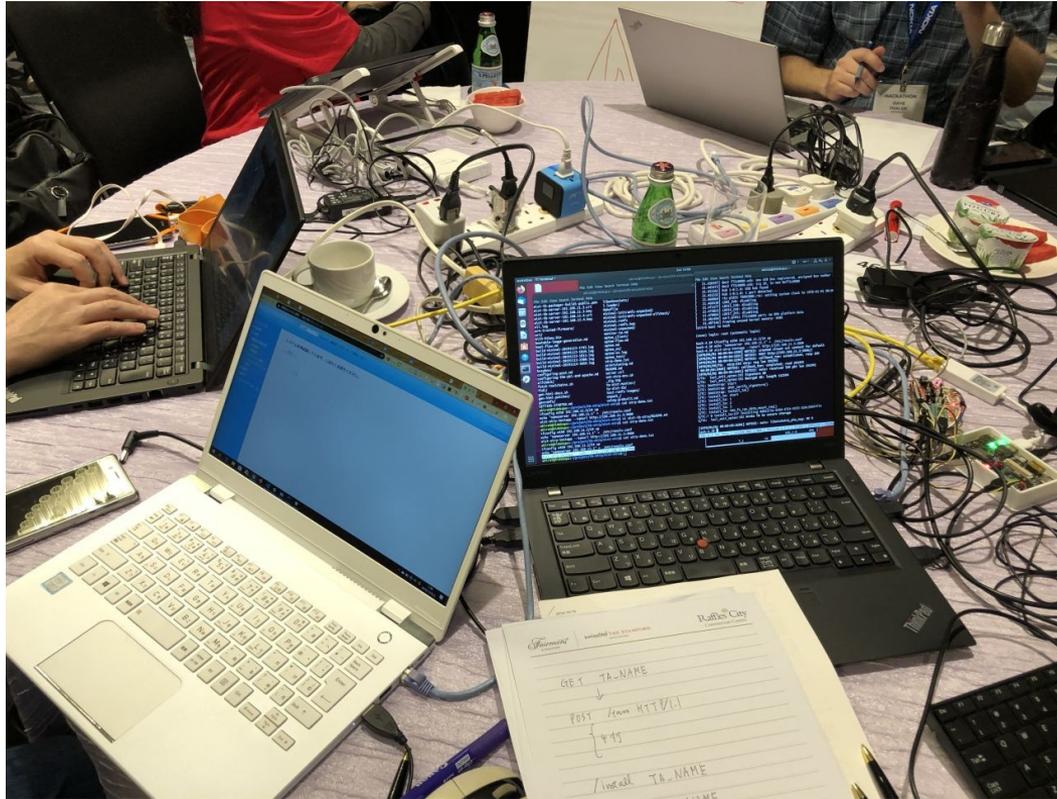
# Great TEEm <sup>ㄴ</sup>



# What got done

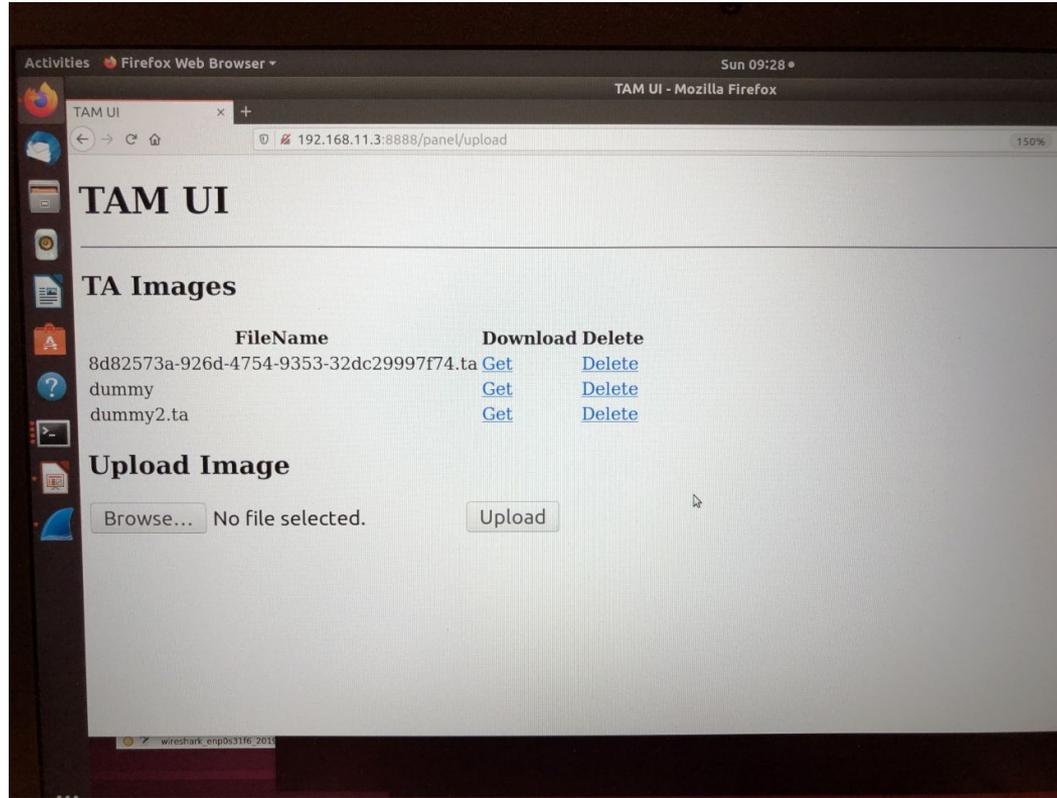
- First time to interop OTrP/TEEP protocol implementations built from specs.
  - See pictures on following pages.

# On the Table



IETF 106 Hackathon - TEEP

# TAM`s UI for uploading TA





# Hacking, Debugging!

Wireshark interface showing network traffic analysis. The main pane displays a list of captured packets, with packet 7 selected. The packet details pane shows an HTTP POST request to /api/tam. The packet bytes pane shows the raw data of the request.

No.	Time	Source	Destination	Protocol	Length	Info
2.	720.6..	192.168.11.2	192.168.11.3	HTTP	3..	POST /api/tam HTTP/1.1 Continuation
2.	720.6..	192.168.11.2	192.168.11.2	HTTP	3..	HTTP/1.1 204 No Content
3.	1100..	192.168.11.3	192.168.11.3	HTTP	2..	POST /api/tam HTTP/1.1
3.	1100..	192.168.11.3	192.168.11.2	HTTP	3..	HTTP/1.1 204 No Content
3.	1316..	192.168.11.2	192.168.11.3	HTTP	2..	POST /api/tam HTTP/1.1
3.	1316..	192.168.11.3	192.168.11.2	HTTP	3..	HTTP/1.1 204 No Content
7.	4204..	192.168.11.2	192.168.11.3	HTTP	2..	POST /api/tam HTTP/1.1
7.	4204..	192.168.11.3	192.168.11.2	HTTP	3..	HTTP/1.1 204 No Content

Request Method: POST  
Request URI: /api/tam  
Content-Type: application/json

```
26.240886] rtc-pl031 f8003000.rtc: setting system clock to 1970-01-01 00:00:26 UTC (2)
26.249254] LD02_2V8: disabling
26.252409] LD013_1V8: disabling
26.255658] LD014_2V8: disabling
26.258895] LD017_2V5: disabling
26.262355] uart-pl011 f/113000.uart: no DMA platform data
26.268005] Freeing unused kernel memory: 448K
Initrd boot to bash

(none) login: root (automatic login)

bash-4.3# ifconfig eth0 192.168.11.2/24 up
bash-4.3# echo "nameserver 192.168.11.1" > /etc/resolv.conf
bash-4.3# aist-otrp-testapp --tamurl http://192.168.11.3:8888
[1970/01/01 00:02:43:9124] NOTICE: created client ssl context for default
[1970/01/01 00:02:43:9125] NOTICE:
http://192.168.11.3:8888/api/tam HTTP/1.1
Host: example.com
Accept: application/otrp-json
Content-Type: application/otrp-json
Content-Length: 0

[1970/01/01 00:02:43:9126] NOTICE: lws_client_connect_via_info: start 1
[1970/01/01 00:02:43:9126] NOTICE: lws_client_connect_via_info: start 2
[1970/01/01 00:02:43:9126] NOTICE: lws_client_connect_via_info: start 3
[1970/01/01 00:02:43:9129] NOTICE: lws_client_connect_via_info: start 4
[1970/01/01 00:02:43:9167] NOTICE: callback_tam: established, resp 204

CTRL-A Z for help 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0
```

```
(none) login: root (automatic login)

bash-4.3#
bash-4.3#
bash-4.3# ifconfig eth0 192.168.11.2/24 up
bash-4.3# echo "nameserver 192.168.11.1" > /etc/resolv.conf
bash-4.3# aist-otrp-testapp --tamurl http://192.168.11.3:8888
[1970/01/01 00:00:37:5601] NOTICE: lws_client_connect_via_info: start 1
[1970/01/01 00:00:37:5602] NOTICE: lws_client_connect_via_info: start 2
[1970/01/01 00:00:37:5603] NOTICE: lws_client_connect_via_info: start 3
[1970/01/01 00:00:37:5604] NOTICE: lws_client_connect_via_info: start 4
[1970/01/01 00:00:37:5605] NOTICE: callback_tam: established, resp 204

CTRL-A Z for help 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0

connection: = close
http/1.1 = 204 No Content
cache-control: = no-store
date: = Sun, 17 Nov 2019 02:27:38 GMT
etag: = W/"a-f/dejJ7Cxb14KKHroepR2JHs04"

main: libalstotrp_tam_msg: -5
bash-4.3#
bash-4.3#
bash-4.3#
bash-4.3#
bash-4.3#
CTRL-A Z for help 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0
thread_mutex_unlock(&ctx->lock); /* ctx->lock ----- %/
472L, 12184C written 255,1-8 52%
```

# TEEP Device installing TA

```
I/TC: bootstrap: install_ta()
on I/TC: install_ta: start
: unI/TC: install_ta: 1
I/TC: install_ta: 2
I/TC: install_ta: 3
I/TC: install_ta: tee_fs_rpc_data_mount_req()
E/TC:? 0 install_ta:117 Installing 8d82573a-926d-4754-9353-32dc29997f74
E/TA: lws_l_emit_optee:101 Wrote TA to secure storage

[1970/01/01 00:00:41:9851] NOTICE: main: libaistotrp_pta_msg: OK 0
bash-4.3# aist-otrp-test
aist-otrp-test-ta-client aist-otrp-testapp
bash-4.3# aist-otrp-test-ta-client
AIST ta-aist-test client
I/TA: TA_InvokeCommandEntryPoint: AIST OTRP Test TA: Hello IETF TEEP!

aist_otrp_test_ta_client: done
bash-4.3#
```

CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | t

100,1-8 45%

# What we learned

- Filed issues
  - draft-ietf-teep-otrp-over-http-03
    - #5: demuxing TEEP vs OTrP
  - draft-tschofenig-teep-protocol-00.txt
    - Would like to have JSON example
- A lot of implementation action items
  - Prerequisite required for OTrP/TEEP
    - HTTP, JSON, CBOR stack must be completely working
  - Understand TEE concepts, such as SGX, Arm TrustZone, knowledge of implementation details (e.g. OP-TEE)

# What went well

- Constructing stand alone wired network on Hackathon table for TAMs and TEEP devices but having uplink
  - This will prevent harming IETF network when sending broken packets. ^^
  - My TEEP device needs to talk to ntp, since does not have RTC.
- Cross checking different TAMs and different TEEP device OTrP messages.
  - Dave`s TAM even sends back what was wrong in the message in the http response. e.g. Content-length missing etc.
- Able to come up for the future plan.

# Future consideration

- How to make it easier to implementation TEEP system?
- What to do for reference implementation?
  - At the hackathon, I started of OTrP debugging and end up debugging http header and json parser.
- IDE Development environment for TA on TEE?
- Many selections for hardware and software stack for TEEP
  - Which hardware?
  - Which software stack to use on TEEP device?
    - JSON stack
    - HTTP stack
    - Crypto stack for TLS and JWE, JWS
    - CBOR parser

# Hardware recommendation

- Reference TAM machine
  - Recommending IBM PC compatible machine?
  - Any other hardware requirement?
- Reference TEEP device (IoT device, Edge device and etc)
  - Recommended device for each Intel, ARM, RISC-V.
    - ARM, OP-TEE usable device
      - Raspberry Pi 3B (Cortex A53) or later?
    - Intel, SGX usable device
      - Laptop PC? (not all SGX usable)
    - RISC-V, PMP extension usable device
      - HiFive Unleashed? (the device only exist at the moment)

# Software stack recommendation

- TAM
  - HTTP stack: Apache
  - JSON stack: Node.js
  - Crypto: openssl
  - CBOR: ?
- TEEP device (limited hardware performance)
  - rootfs: buildroot, Yocto/OE, openwrt?
  - HTTP stack: libwebsocket?
  - JSON stack: libwebsocket?
  - Crypto(TLS,JWE,JWS): openssl, LibreSSL, mbedTLS, wolfSSL, s2n?
  - CBOR: ? ?

# Nice to have? Or out of scope?

- TEEP: Testbed on Internet
  - TAM: Everybody connecting from their own TEEP devices
- IDE Development environment for TA on TEE
  - OpenEnclave
- Hosting github for TEEP reference implementation?
- TAM: security hardware
  - SGX: Any other? OpenTitan?
- TEEP: security hardware
  - Any other? Azure Sphere IoT?

# My notes from hackathon

- Fix header for HTTP compliant
  - I broke the HTTP header when revising OTrP messages.
- Add JSON parsing for every packet received
- Cleanup and dependency fix of Makefile
  - It does not detect some dependency when I change some of the code.
- microUSB cable for flashing bootloader
  - Suffered a lot of having bad connection, have to change both the 3D printed case and cable.
- Add dumping the all content of http packet every time
  - To reduce the time using wireshark.
- Buy reliable self-powered USB-hub.
  - One of the hub did not recognize the gpio board.

# Wrap Up

## Team members:

Akira Tsukamoto

Kuniyasu Suzuki

Kohei Isobe

Dave Thaler

Hannes Tschofenig

Nancy Cam-Winget

<https://trac.tools.ietf.org/wg/teep/>

This presentation of hackathon is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).