

20231006

TINYIOT

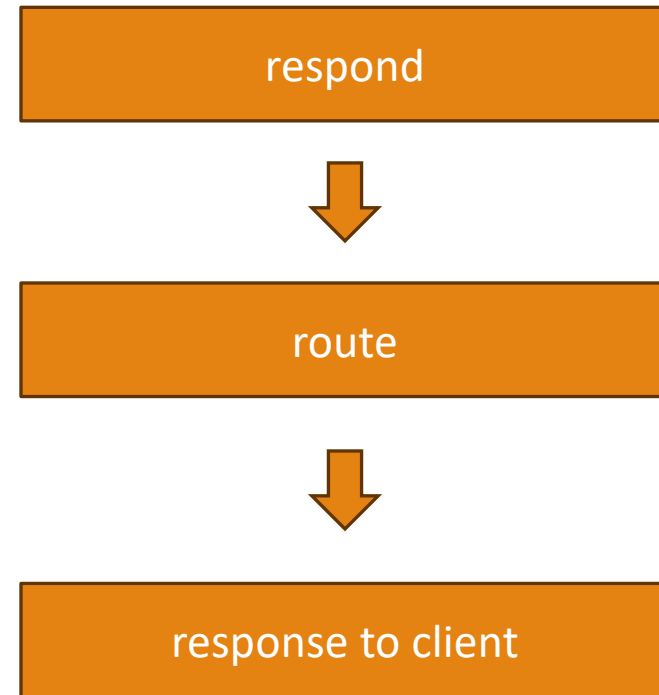
이번 주 진행 상황

1. HTTP.c 수정
2. MN-Register 제작
3. Announcement 공부

httpd.c

기존의 코드는 파이프라인처럼 동작

즉 HTTP request를 파싱하거나,
Response를 파싱하는 코드가 재사용
불가능



httpd.c

Request를 구성하는 HTTPRequest 구조체,

Response를 구성하는 HTTPResponse 구조체 추가

You, 3개월 전 | 1 author (You)

```
typedef struct _request {  
    char *method;  
    char *uri;  
    char *qs;  
    char *prot;  
    char *payload;  
    int payload_size;  
    header_t *headers;  
} HTTPRequest;
```

You, 22시간 전 | 1 author (You)

```
typedef struct _response {  
    int status_code;  
    char *protocol;  
    char *status_msg;  
    header_t *headers;  
    char *payload;  
    int payload_size;  
} HTTPResponse;
```

httpd.c

header 구현 수정

기존의 Array 방식에서 연결리스트로 수정

Header관련 함수 수정

- request_header => search_header
- set_response_header => set_header

```
typedef struct _header{  
    char *name;  
    char *value;  
    struct _header *next;  
} header_t;      You, 3개
```

httpd.c

재사용가능한 구조로 수정

HTTPRequest구조체와 HTTPResponse구조체 사용

parse_http_request와 parse_http_response 로 Request, Response를 파싱해서 각 구조체에 넣어주도록 함수 분리

Request용 함수 추가

send_http_request(char *host, int port, HTTPRequest *req, HTTPResponse *res)

구조체만 짜서 인자로 넘겨주면 Request를 진행해주는 함수 추가.

(부수적으로 Forwarding과 Notify 코드 수정)

MN-Register

Common Services Entities (CSE)

<u>CSE-ID</u>	<u>Type</u>	<u>Name</u>	<u>Resource ID</u>	<u>Release</u>	<u>Reachable</u>	<u>POA</u>	<u>Registrar</u>	<u>Registrees</u>
<i>/id-in</i> <i>/tinyiot</i>	<i>IN</i> MN	<i>cse-in</i> tinyiot	<i>id-in</i> tinyiot	<i>2a, 3, 4, 5</i> 2a	<i>True</i> True	<i>http://127.0.0.1:8000</i> http://192.168.0.252:3000	<i>/id-in</i>	<i>/tinyiot</i>

MN-Register

1 단계 – Registrar CSE 에 CSR 리소스 존재하는지 확인

- 존재 시 2, 3단계 SKIP
- 연결 실패 시 Terminate

```
GET /cse-in/tinyiot HTTP/1.1
Host: 127.0.0.1:8000
X-M2M-Origin: /tinyiot
X-M2M-RI: check-cse-registered
Accept: application/json
Content-Type: application/json
X-M2M-RVI: 2a
```

```
Thu Oct 5 15:34:08 2023 DEBUG [UTIL]: Remote CSE registration check: 404
Thu Oct 5 15:34:08 2023 DEBUG [UTIL]: Remote CSE is not registered
```


MN-Register

2 단계 – Registrar CSE 에 CSR 리소스 create 요청

- 실패 시 Terminate

```
Thu Oct  5 15:34:08 2023 DEBUG [HTTP]: sent
POST /cse-in HTTP/1.1
Host: 127.0.0.1:8000
X-M2M-Origin: /tinyiot
X-M2M-RI: register-cse
Accept: application/json
Content-Type: application/json;ty=16
X-M2M-RVI: 2a
Content-Length: 145
```

```
{"m2m:csr":{"rn":"tinyiot","cst":2,"csi":"tinyiot","srv":["2a"],"rr":true,"poa":["http://172.16.12.250:3000"],"
```

MN-Register

3 단계 – Registrar CSE 의 CSR리소스 생성

```
SQL: INSERT INTO csr (ri,cst,poa,cb,dcse,csi,mei,tri,csz,rr,nl,srv) VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?);  
Add Resource Tree Node [Parent-ID] : tinyiot, [Child-ID] : cse-in
```

MN- Deregister

```
^CThu Oct 5 15:29:21 2023 INFO [MAIN]: Shutting down server...
Thu Oct 5 15:29:21 2023 INFO [MAIN]: De-registrating cse...
Thu Oct 5 15:29:21 2023 DEBUG [HTTP]: header: Host: 127.0.0.1:8000
X-M2M-Origin: /tinyiot
X-M2M-RI: delete-csr
Content-Type: application/json
X-M2M-RVI: 2a
```

CSR 삭제 요청

```
Thu Oct 5 15:29:21 2023 DEBUG [HTTP]: sent
DELETE /~/id-in/cse-in/tinyiot HTTP/1.1
Host: 127.0.0.1:8000
X-M2M-Origin: /tinyiot
X-M2M-RI: delete-csr
Content-Type: application/json
X-M2M-RVI: 2a
```

ACME (IN-CSE)

```
{
  "fr": "/tinyiot",           // from
  "op": 4,                    // operation: DELETE
  "org": "/tinyiot",          // originator
  "rqi": "delete-csr",        // requestIdentifier
  "rvi": "2a",                 // ReleaseVersionIndicator
  "to": "/id-in/cse-in/tinyiot" // to
}
```

```
Thu Oct 5 15:29:21 2023 DEBUG [DB]: Delete [RI] cse-in
```

로컬 CSR 삭제

```
Thu Oct 5 15:29:21 2023 DEBUG [DB]: Closing DB...
Thu Oct 5 15:29:21 2023 DEBUG [DB]: Cleaning ResourceTree...
Thu Oct 5 15:29:21 2023 INFO [MAIN]: Done
```

RR (Request Reachability)

// Todo:

rr: true 인 경우, 주기적으로 retrieve 요청
(ACME의 경우 30초)

```
Thu Oct 5 15:38:08 2023 DEBUG [HTTP]: [GET] ~/tinyiot/TinyIoT
Thu Oct 5 15:38:08 2023 DEBUG [HTTP]: to: ~/tinyiot/TinyIoT
Thu Oct 5 15:38:08 2023 INFO [HTTP]: Request : GET
Thu Oct 5 15:38:08 2023 INFO [HTTP]: Client connected 172.16.12.250

Thu Oct 5 15:38:08 2023 DEBUG [HTTP]: ip: 172.16.12.250
Thu Oct 5 15:38:08 2023 DEBUG [O2M]: Call parse_uri ~/tinyiot/TinyIoT
Thu Oct 5 15:38:08 2023 DEBUG [MAIN]: handle_onem2m_request
Thu Oct 5 15:38:08 2023 INFO [UTIL]: Run time : 0.000017
Thu Oct 5 15:38:08 2023 DEBUG [HTTP]:

HTTP/1.1 200 OK
Connection: Close
Accept: application/json
Content-Type: application/json; charset=utf-8
Access-Control-Allow-Origin: *
Access-Control-Allow-Headers: Accept, Accept-Language, Content-Language, Content-Type
Access-Control-Allow-Methods: GET, PUT, POST, DELETE, OPTIONS
Access-Control-Request-Methods: GET, PUT, POST, DELETE, OPTIONS
Content-Length: 213
X-M2M-RSC: 2000
X-M2M-RVI: 2a
X-M2M-RI: 6816573519915444416

{"m2m:cb":{"rn":"TinyIoT","ri":"tinyiot","ct":"20231005T144157000800","lt":"20231005T144157000800","poa":["http://172.16.12.250:3000"]}}
```

Minor Fix

특정 NIC를 지칭하면 NIC 의 IP address 를 서버 실행 시 받아와 poa를 구성하도록 수정