AutoCrypt

PnC vPKI Process

제어기 Provisioning Certificate 발급 절차

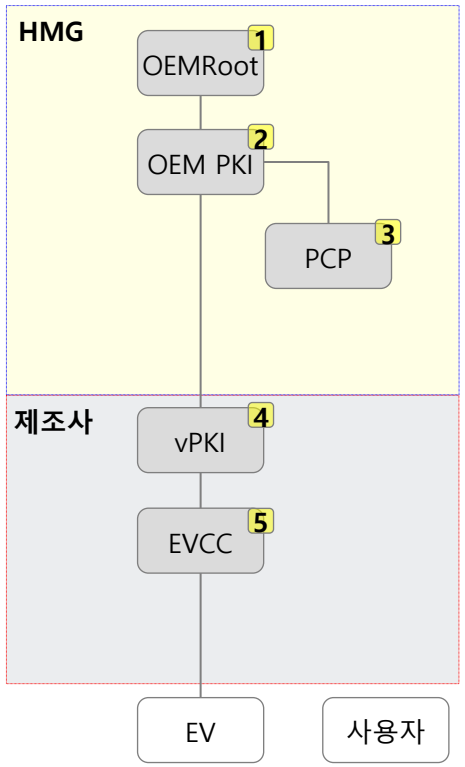
AUTOCRYPT

TU-Automotive AWARDS

Best Automotive Cybersecurity Product/Service **2019**

The Automotive Tech Company of the Year Finalist **2020**

시스템 구성도



구 분	No	이름	세 부 내 용	비고
OEM	1	OEM Root	OEM PKI의 Root 인증 기관	S/W System
	2	OEM PKI	OEM Provisioning Certificate를 발급하는 인증 기관	S/W System
	3	OEM PCP	OEM Provisioning Certificate를 저장하는 Pool	S/W System
	4	vPKI 서버	OEM Provisioning Certificate 발급 요청을 하는 시스템	S/W System
	5	EVCC 제어기	V2GRoot와 EVCC 인증서를 탑재한 제어기	H/W

ISO 15118-2 Provisioning cert PCID 생성 규칙

항목	값	비고
RANDOM UUID 방식		
UUID	c681b613-d440-494b-8ace-8e1c8fb9111e	Version 4 or 5
BASE(32) conversion of UUID	88J8V39P42B77SPMNG5L9VL6AY	
Selected 12 char from above	PMNG5L9VL6AY	
MAC Address 기반 방식		
NOT operation of MAC Address	04E77E000148 ->FB1881FFFEB7	
WMI	~~KMH~~	~~Hyundai Korea~~
	KM8	ISO 15118-2
ID Type	P	
Supplier ID	지정된 ID값	2~Z
PCID/wo check digit	KM8P2FB1881FFFEB7	
Value string of above	20228252151118811515151514117	
Checksum of value string	573221402	
Modulo-11 of checksum	6	
PCID	**KM8P2FB1881FFFEB76**	**18 chars**

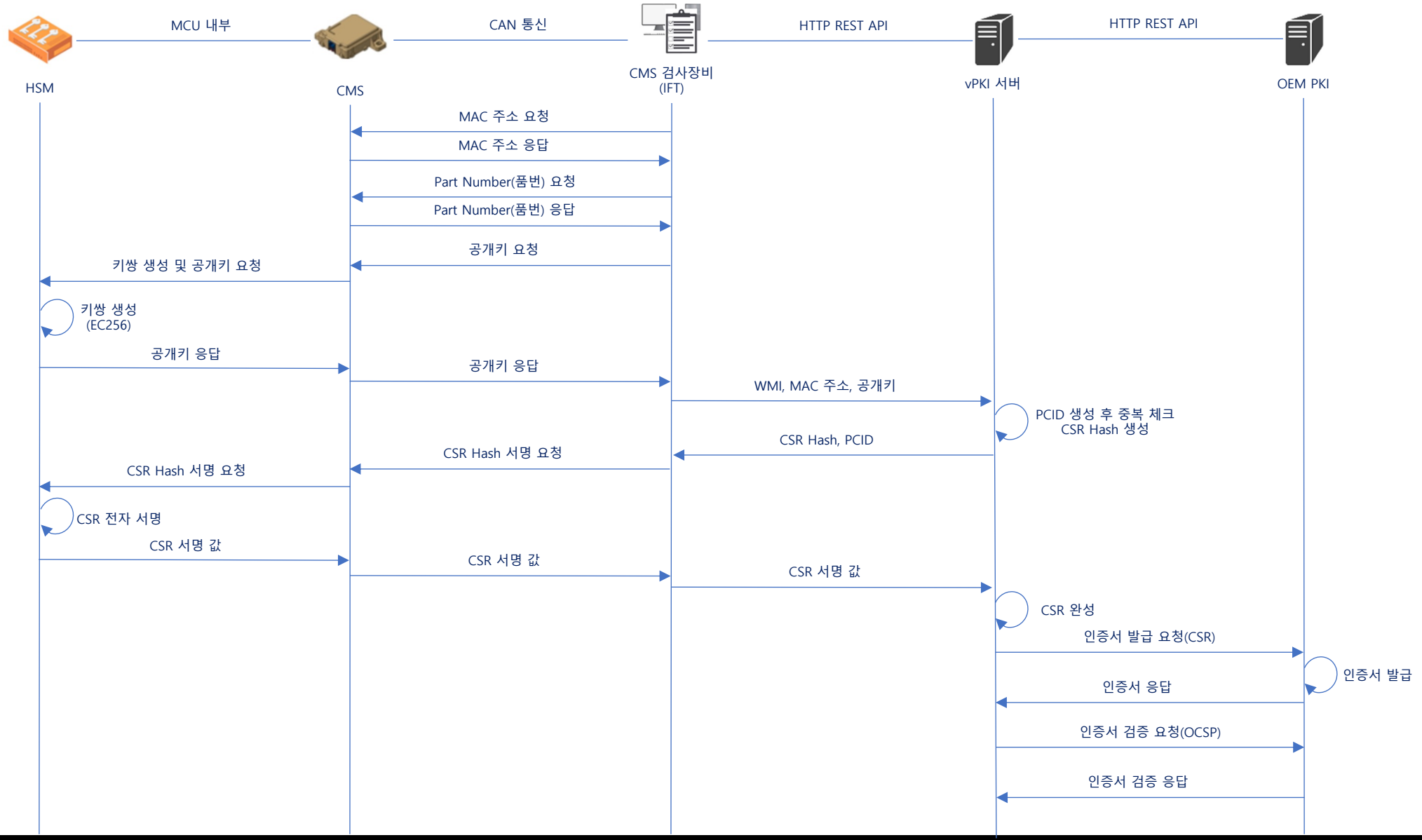
ISO 15118-20 Provisioning cert PCID 생성 규칙

항목	값	비고
RANDOM UUID 방식		
UUID	c681b613-d440-494b-8ace-8e1c8fb9111e	Version 4 or 5
BASE(32) conversion of UUID	88J8V39P42B77SPMNG5L9VL6AY	
Selected 12 char from above	PMNG5L9VL6AY	
MAC Address 기반 방식		
NOT operation of MAC Address	04E77E000148 ->FB1881FFFEB7	
WMI	~~KMH~~	~~Hyundai Korea~~
	KM7	ISO 15118-20
ID Type	P	
Supplier ID	지정된 ID값	2~Z
PCID/wo check digit	KM7P2FB1881FFFEB7	
Value string of above	20227252151118811515151411 7	
Checksum of value string	573221386	
Modulo-11 of checksum	1	
PCID	**KM8P2FB1881FFFEB71**	**18 chars**

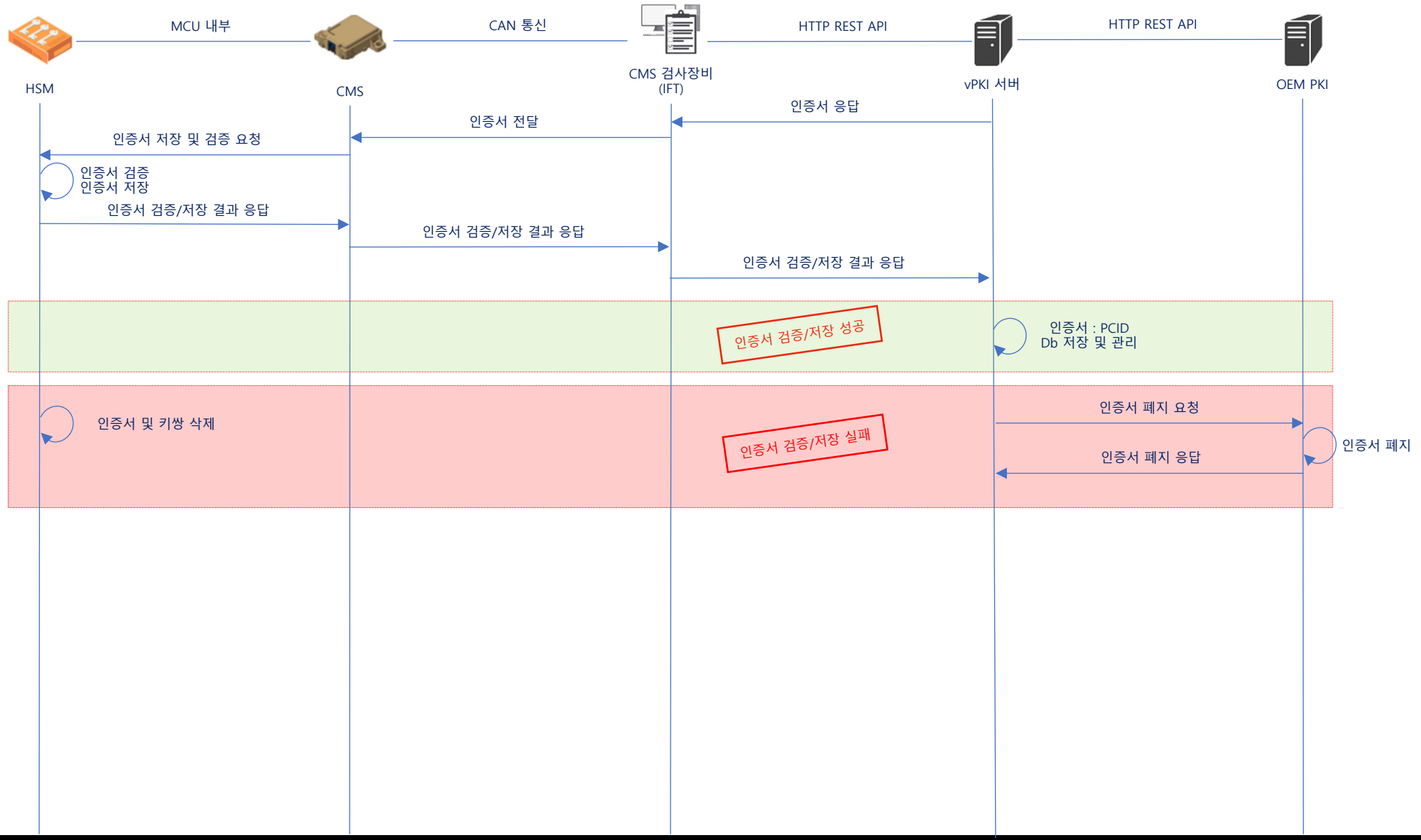
ISO 15118-20 vehicle cert PCID 생성 규칙

항목	값	비고
RANDOM UUID 방식		
UUID	c681b613-d440-494b-8ace-8e1c8fb9111e	Version 4 or 5
BASE(32) conversion of UUID	88J8V39P42B77SPMNG5L9VL6AY	
Selected 14 char from above	39P42B77SPMNG5	
MAC Address 기반 방식		
NOT operation of MAC Address	04E77E000148 ->FB1881FFFEB7	
WMI	~~KMH~~	~~Hyundai Korea~~
	~~KM8~~	
	KM7	ISO 15118-20
ID Type	V	
Supplier ID	지정된 ID값	2~Z
EVCCID/wo check digit	KM7V2**00**FB1881FFFEB7	2자리는 **00**으로 고정
Value string of above	2.02273E+26	
Checksum of value string	573221162	
Modulo-11 of checksum	8	
EVCCID	**KM7V200FB1881FFFEB78**	**20 chars**

OEM Provisioning Certificate 발급 절차



OEM Provisioning Certificate 발급 절차



ISO 15118-2 Provisioning cert API

CSR 생성 요청

URL	/api/v1/tbscsr
method	POST

Request

```
body : {
    iftid: string,                  // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.
    cnInfo : {
        macaddr : string,           // 인증서 발급 대상 장비의 MAC address.
        wmi : string,               // PCID에 들어갈 WMI 값.("KM8")
        idType : string,            // "P"
        supplierId : string.        // 지정된 ID값 2~Z
    },
    certInfo : {
        dc : string,                // 인증서 dc값("OEM")
        tierCode: String,           // 협력사 코드(P002)
        unitCode : String           // 제어기 코드(C001)
    },
    publickey : string,             // 인증서 발급 대상 장비에서 생성된 키쌍 중 공개키 값. Secp256r1으로 키 생성 후 unpressess 결과값을 전달.
}
```

Response – Success

```
{
    "status": string        // success
    "message": string,      // "",
    "data": {
        "hashedtbscsr": string  // 중계서버에서 생성한 CSR 원문을 SHA256 Hash 후 base64 인코딩한 값.
        "pcid": string //pcid: WMI와 Mac address 기반으로 생성한 PCID 값
        }
}
```

Response – Failure

```
{
    "status": string,  // error
    "message": string,  // Exception Message
    "data": {}
}
```

인증서 발급 요청

URL	/api/v1/certificate
method	POST

Request

```
{
    "iftid": string                  // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값. ( "ift001"
    "csrsignature":  string      // CSR을 SHA256withECDSA 알고리즘으로 서명한 데이터를 base64 인코딩한 값
    "unitCode" : string           // "제어기 코드", (C001)
    "tierCode" :  string          // "협력사 코드", (P002)
    "vehicleCode" :  string     // "차종 코드",
    "localCode" :  string        // "지역 코드",
    "brandCode" : string        //"브랜드 코드"
}
```

Response – Success

```
{
   "status": string, // success
   "message":  string, // ""
   "data": {
          "leafcertificate": string, // 발급받은 인증서를 base64 인코딩한 값
           "subcacertificate": string,  // 인증서를 발급한 SubCA 인증서를 base64 인코딩한 값
        }
}
```

Response – Failure

```
{
    "status": string  // error
    "message": string  // Exception Message
    "data": {}
}
```

인증서 검증 결과 전송

URL	/api/v1/verifyresult
method	POST

Request

```
{
    "iftid": string //  IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.
   "pcid": string  // WMI와 Mac address 기반으로 생성한 PCID 값
    "result": string // success
}
```

Response – Success

```
{
   "status": string // "success",
   "message": string // "",
   "data": {
          "iftid": string   // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.
          "pcid": string    // WMI와 Mac address 기반으로 생성한 PCID 값
          "ift_result": string  // IFT 장비로부터 받은 결과 값
          "server_result": string // 서버 처리 결과 값 ("nothingtodo" || "revoked")
        }
}
```

Response – Failure

```
{
    "status": "error",
    "message": "Exception Message",
    "data": ""
}
```

ISO 15118-20 Provisioning cert API

Provisioning cert CSR 생성 요청

URL	/api/v2/prov/tbscsr
method	POST

Request

```
body : {
    iftid: string,                  // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.
    cnInfo : {
        macaddr : string,           // 인증서 발급 대상 장비의 MAC address.
        wmi : string,               // PCID에 들어갈 WMI 값.
        idType : string             // "P"
        supplierId : string.        // 지정된 ID값 2~Z
    },
    certInfo : {
        dc : string,                // 인증서 dc값("OEM20")
        tierCode: String,           // 협력사 코드(P002)
        unitCode : String           // 제어기 코드(C001)
    },
    publickey : string,             // 인증서 발급 대상 장비에서 생성된 키쌍 중 공개키 값. Secp521r1으로 키 생성 후 unpressess 결과값을 전달.
    certType : string;              //"prov_cert"
}
```

Response – Success

```
{
    "status": "success",
    "message": "",
    "data": {
        "hashedtbscsr": "SHA512 hashed base64 encoded TBS CSR",     // 중계서버에서 생성한 CSR 원문을 SHA512 Hash 후 base64 인코딩한 값.
        "pcid": string                                              // WMI와 Mac address 기반으로 생성한 PCID 값
        }
}
```

Response – Failure

```
{
    "status": "error",
    "message": "Exception Message",
    "data": {}
```

인증서 발급 요청

URL	/api/v2/prov/certificate
method	POST

Request

```
{
    "iftid": string                  // "ift001" IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값,
    "csrsignature": string           // hashedCsrStr 값을 개인키로 서명 후 der encoding한 결과 값
    "unitCode" : string          // "제어기 코드", (C001)
    "tierCode" :  string         // "협력사 코드", (P002)
    "vehicleCode" :  string    // "차종 코드",
    "localCode" :  string       // "지역 코드",
    "brandCode" : string       //"브랜드 코드"
}
```

Response – Success

```
{
    "status": "success",
    "message": "",
    "data": {
        "leafcertificate": string ,        // provisioning cert  vehicle cert  발급받은 인증서를 base64 인코딩한 값
        "subcacertificate": string       // sub ca2 cert 인증서를 발급한 SubCA 인증서를 base64 인코딩한 값
      }
}
```

Response – Failure

```
{
    "status": "error",
    "message": "Exception Message",
    "data": {}
}
```

인증서 검증 결과 전송

URL	/api/v2/prov/verifyresult
method	POST

Request

```
{
    "iftid": “ift001”   // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.,
   “pcid”: “WMIABCDEFGHI9”. // WMI와 Mac address 기반으로 생성한 PCID 값
   “result”: “success” // 발급받은 인증서를 검증/설치한 결과. success 혹은 실패 이유(협의필요)
}
```

Response – Success

```
{
   "status": "success",
   "message": "",
   "data": {
      "iftid":" ift001 " . // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.,
      "pcid":" WMIABCDEFGHI9 "  // WMI와 Mac address 기반으로 생성한 PCID 값,
      "ift_result":"success "  // IFT 장비로부터 받은 결과 값,
           "server_result":"nothingtodo" || “revoked”. //서버 처리 결과 값
       }
}
```

Response – Failure

```
{
    "status”: “error”,
    "message”: “Exception Message”,
    "data": {}
}
```

ISO 15118-20 vehicle cert API

vehicle cert CSR 생성 요청

URL	/api/v2/vehicle/tbscsr
method	POST

Request

```
body : {
    iftid: string,                  // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.
    cnInfo : {
        macaddr : string,           // 인증서 발급 대상 장비의 MAC address.
        wmi : string,               // PCID에 들어갈 WMI 값.
        idType : string             // "V"
        supplierId : string.        // 지정된 ID값 2~Z
    },
    certInfo : {
        dc : string,                // 인증서 dc값("EV")
        tierCode: String,           // 협력사 코드(P002)
        unitCode : String           // 제어기 코드(C001)
    },
    publickey : string,             // 인증서 발급 대상 장비에서 생성된 키쌍 중 공개키 값. Secp521r1으로 키 생성 후 unpressess 결과값을 전달.
    certType : string;              // "vehicle_cert"
}
```

Response – Success

```
{
    "status": "success",
    "message": "",
    "data": {
        "hashedtbscsr": "SHA512 hashed base64 encoded TBS CSR",     // 중계서버에서 생성한 CSR 원문을 SHA512 Hash 후 base64 인코딩한 값.
        "pcid": string                                              // WMI와 Mac address 기반으로 생성한 PCID 값
        }
}
```

Response – Failure

```
{
    "status": "error",
    "message": "Exception Message",
    "data": {}
```

Vehicle 인증서 발급 요청

URL	/api/v2/vehicle/certificate
method	POST

Request

```
{
    "iftid": string                    // "ift001" IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값,
    "csrsignature": string             // hashedCsrStr 값을 개인키로 서명 후 der encoding한 결과 값,
    "unitCode" : string          // "제어기 코드", (C001)
    "tierCode" :  string         // "협력사 코드", (P002)
    "vehicleCode" :  string    // "차종 코드",
    "localCode" :  string       // "지역 코드",
    "brandCode" : string       //"브랜드 코드"
}
```

Response – Success

```
{
    "status": "success",
    "message": "",
    "data": {
        "leafcertificate": string ,        // vehicle cert  발급받은 인증서를 base64 인코딩한 값
        "subcacertificate": string       // sub ca2 cert 인증서를 발급한 SubCA 인증서를 base64 인코딩한 값
     }
}
```

Response – Failure

```
{
      "status": "error",
      "message": "Exception Message",
      "data": {}
}
```

Vehicle 인증서 검증 결과 전송

URL	/api/v2/vehicle/verifyresult
method	POST

Request

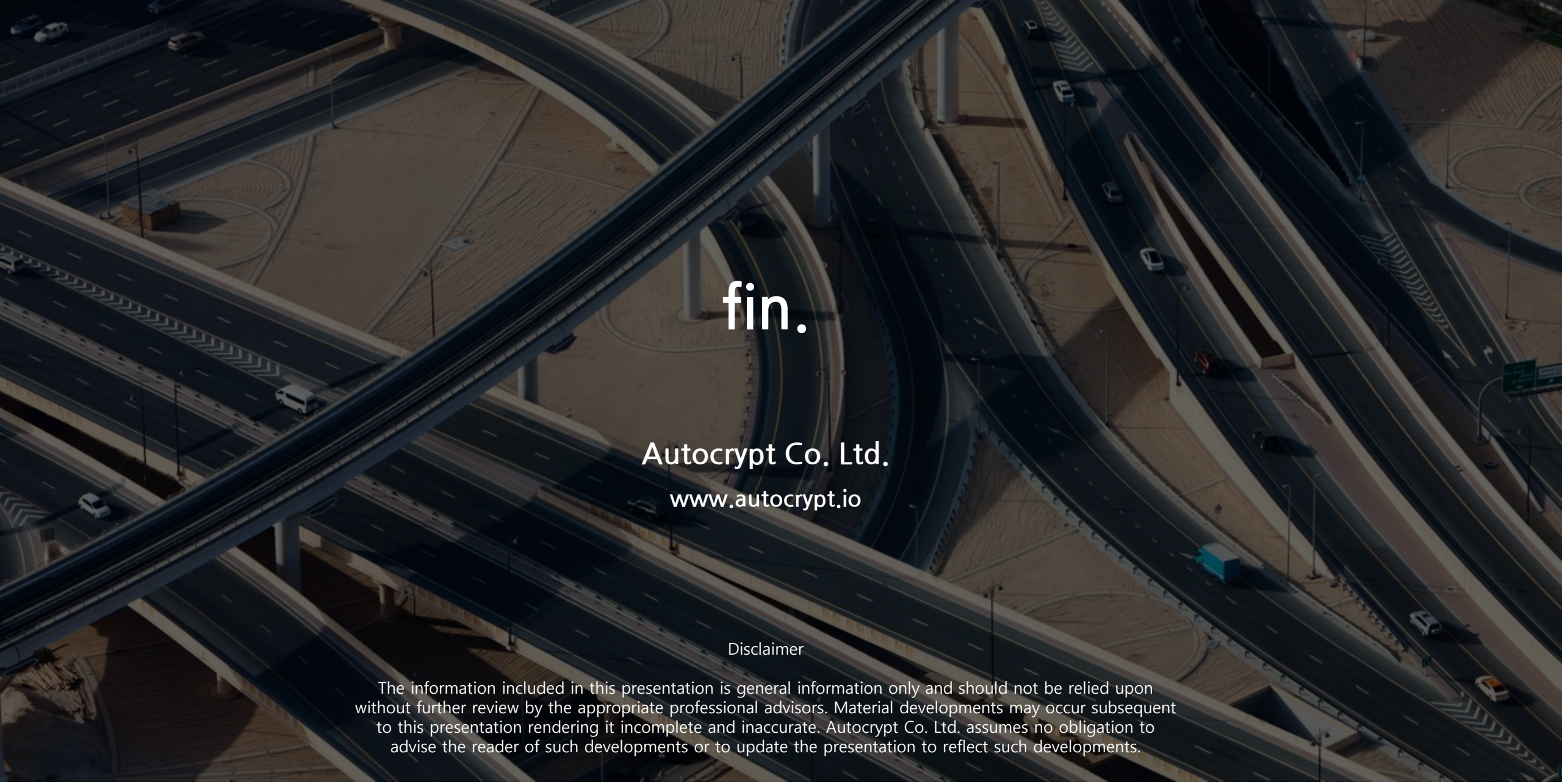
```
{
    "iftid": "ift001" // IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.,
   "pcid": "WMIABCDEFGHI9" // WMI와 Mac address 기반으로 생성한 PCID 값
   "result": "success" // 발급받은 인증서를 검증/설치한 결과. success 혹은 실패 이유(협의필요)
}
```

Response – Success

```
{
   "status": "success",
   "message": "",
   "data": {
      "iftid":" ift001 "  //IFT 장비의 ID, 중계서버에서 각 IFT장비를 구분할 수 있는 값.,
      "pcid":" WMIABCDEFGHI9 "  // WMI와 Mac address 기반으로 생성한 PCID 값,
      "ift_result":"success " . // IFT 장비로부터 받은 결과 값,
           "server_result":"nothingtodo" || "revoked" // 서버 처리 결과 값
       }
}
```

Response – Failure

```
{
    "status": "error",
    "message": "Exception Message",
    "data": {}
}
```



fin.

Autocrypt Co. Ltd.

www.autocrypt.io

AUTOCRYPT