

SEMI E132-0414

장비클라이언트 인증과 권한 부여에 대한 사양 (SPECIFICATION FOR EQUIPMENT CLIENT AUTHENTICATION AND AUTHORIZATION)

본 표준은 Information & Control Global Technical Committee 에서 기술 검증 되었으며, 본 버전은 global Audits and Reviews Subcommittee 에 의해 2013 년 12 월 23 일 간행 승인 되었다. 2014 년 3 월 이후 www.semiviews.org 와 www.semi.org 에서 열람 가능하다; 2004 년 3 월 초본 간행; 2010 년 8 월 이전 버전 간행.

NOTICE: This translation is a REFERENCE COPY ONLY. If differences should exist between the English version and a translation in any other language, the English version is the official and authoritative version.

본 번역서는 참고용으로써, 영문원본과 한글번역본간에 차이가 발생할 경우, 영문원본을 우선으로 해야함을 알려드립니다. 본 번역서에 대해 문의사항이 있는 경우 krstandards@semi.org 로 문의바랍니다.

1 목적 (Purpose)

1.1 본 표준은 통신을 진행하기 위해 앞서 장비에 클라이언트의 인증을 요구하여 통신에 대한 접근을 제약하는 메소드(method)를 설명하며, 클라이언트 어플리케이션 접근을 컨트롤 하는 유연한 허가 스키마(authorization scheme)를 제공한다. 권한 부여를 통해서 정보를 교환하거나 장비를 컨트롤하고자 하는 주체가 정보 혹은 사용에 대한 필요(need-to-know/need-to-use)를 기반으로 정보의 통신 및 장비 컨트롤이 가능하다.

1.2 본 표준에 명시된 허가 스키마(authorization scheme)는 장비 공급사가 모든 단계별로 접근 컨트롤을 제공할 수 있는 유연성을 허용하여, 이를 통해 제한이 없는 접근 컨트롤(access control)부터, 사전에 정의된 역할에 기반한 접근 컨트롤 (access control)까지, 세분화된 컨트롤을 할 수 있다.

2 범위 (Scope)

2.1 본 표준은 해당 서비스에 대해 인증(authentication)과 권한 부여(authorization)를 필요로 하는 모든 반도체 제조 장비에 적용되며, SEMI E30 통신 및 컨트롤 상태 모델에 의한 통신에는 적용되지 않는다.

2.2 본 표준은 암호화될 세션을 통한 데이터 전송은 요구하지 않으며, 암호화는 인증 프로토콜(authentication protocol)에서 명시할 경우에만 요구된다. 본 표준에서 명시한 인터페이스는 밀폐된 공장 네트워크와 같은 악성 공격이 없는 환경에서 운영 되는 것으로 가정한다.

NOTICE: SEMI Standards and Safety Guidelines do not purport to address all safety issues associated with their use. It is the responsibility of the users of the Documents to establish appropriate safety and health practices, and determine the applicability of regulatory or other limitations prior to use.

SEMI Standards and Safety Guideline 은 그 사용과 관련한 모든 안전상의 유의사항을 언급하지는 않는다. 본 문서에 기재된 사항을 적절한 안전 및 건강 상의 수칙을 설정하고, 사용 전에 제약 사항 혹은 다른 규제사항의 적용 가능성을 판단하는 것은 문서 사용자의 책임이다.

3 제한사항 (Limitations)

3.1 본 문서에서 정의된 개념을 특정 기술 및 프로토콜과 연결하기 위한 관련 표준이 정의되어야 한다.

4 참조 표준 및 문헌 (Referenced Standards and Documents)

4.1 SEMI Standards and Safety Guidelines

SEMI E30 — Generic Model for Communications and Control of SEMI Equipment (GEM)

SEMI E128 — Specification for XML Message Structures

4.2 Other Standards

Unified Modeling Language (UML) Specification, Version 1.4, OMG Specification 01-09-67; http://www.omg.org/technology/documents/modeling_spec_catalog.htm

Uniform Resource Identifiers (URI): Generic Syntax, IETF RFC 2396, August 1998; <http://www.ietf.org/rfc/rfc2396.txt>

NOTICE: Unless otherwise indicated, all documents cited shall be the latest published versions.

별도의 표시가 없으면, 인용된 모든 문서는 최신 발행본이다.

5 용어 (Terminology)

5.1 정의 (Definitions)

5.1.1 *administration* (관리) — 장비 클라이언트를 식별하고, 연관된 자격 증명 및 권한 부여를 설정하는 프로세스이다.

5.1.2 *Asymmetric Key Cryptography* — Public Key Cryptography 참조.

5.1.3 *authenticated communications* (인증된 통신) — 인증된 세션을 설치하고, 권한을 부여하고 관리하기 위해 본 표준을 준수하는 장비와 실행되는 통신으로 인증된 세션을 통한 모든 통신을 포함한다. 인증된 통신 모두가 암호화되지는 않는다는 점에 유의한다.

5.1.4 *authenticated session* (인증된 세션) — 본 문서에서는 이를 *session* 이라고도 하는데, 본 표준에서 정의하는 어플리케이션 단계의 컨셉으로 실제 메시지 교환에서 사용되는 모든 물리적/논리적 연결 혹은 통신 프로토콜 ‘session’ layer 와는 독립적이다. *Authenticated session* 혹은 *session* 은 세션 설정 절차, 상태 모델, 그리고 본 표준에서 정의한 서비스를 따른다.

5.1.5 *authentication* (인증) — 사용자 혹은 프로세스가 제시하는 정체가 올바른지에 대해 판단하는 프로세스이다. 개인 혹은 공개 컴퓨터 네트워크(public computer networks)에서는 종종 암호(passwords)를 사용한 인증(authentication)이 이루어진다. 해당 사용자를 확인 하기 위해서는 암호(password)의 확인이 전제된다. 본 표준은 어플리케이션 프로세스의 인증만을 언급하며, 사용자(human user)의 인증은 지원하지 않거나 언급하지 않는다.

5.1.6 *authorization* (권한 부여) — 특정 사용자나 프로세스에 어떤 동작(action)을 수행하는 권한을 허가하는 프로세스를 의미한다. 권한 부여(authorization)가 이루어지기 전에 인증이 반드시 선행되어야 하는 점에 유의해야 한다.

5.1.7 *client* (클라이언트) — 장비 리소스(resources)에 접근하기 위해 장비와의 통신을 하는 어플리케이션 프로세스이다. 본 표준에서는 ‘client’와 ‘client application’ 용어가 혼용된다.

5.1.8 *communication session* (통신 세션) — 클라이언트와 장비 사이에 발생하는 양방향 메시지 교환의 시리즈를 의미한다. 일반적으로 세션은 클라이언트의 요청으로 시작되고 종료된다.

5.1.9 *cryptographic algorithm* (암호 알고리즘) — 의미 있는 데이터를 의미 없는 데이터로 변환하는 과정으로, 암호화(encryption)라고 불리는 프로세스이다. 만일 이 변환을 되돌릴 수 있다면, 이에 대한 리버스 프로세스(reverse process)를 암호 해독(decryption)이라고 말한다. 일반적으로는 사용된 알고리즘과, keys 라고 말하는, 일부 추가 정보를 알지 못하는 경우 데이터를 올바르게 해독하기는 어렵다.

5.1.10 *encryption* (암호화) — cryptographic algorithm 참조

5.1.11 *message digest* (메시지 다이제스트) — one-way has 참조.

5.1.12 *one-way hash* (한방향 해쉬) — 이는 임의의 길이 데이터를 고정된 길이의 코드(fixed-length code)로 변환하는 한 방향 변환(one-way transformation)이다. 이 변환은 되돌리기 힘들며, 충돌 가능성이 적다. 동일한 메시지는 동일한 해시(hash)를 생산하고, 원본 메시지 상에서의 약간의 변화는 다른 해시(hash)를 만들어, 메시지의 완전성(message integrity)을 제공한다. 본 문서에서는 One-way Hash, Hash, 그리고 Digest 가 혼용된다.

5.1.13 *Persistent Session* (퍼시스턴트 세션) — 장비 셧 다운(shutdowns) 혹은 전원 종료(power offs)와 무관하게 설정된 *Authenticated Session* 이다. 장비 셧 다운(shutdown) 및 스타트 업(startup) 중 퍼시스턴트 세션(persistent session)의 동작(behavior)은 본 문서에서 설명된 세션 상태 모델(ssession stat model)에 정의된 바를 따른다.

5.1.14 *Public Key Cryptography (PKC)* — 이는 Asymmetric Key Cryptography 라고도 하며, 데이터의 암호화(encryption) 및 암호 해독(decryption)에 두 개의 키(key)를 사용하는 암호화 알고리즘(cryptographic algorithm) 이다. 하나의 (key)는 공개 키(public key)로 지정되어 자유롭게 공개될 수 있으나, 비공개 키(private key)로 지정된 다른 key 는 절대 공개되지 않는다. 하나의 키로 암호화된 데이터는 대응하는 다른 key 로만 해독될 수 있다. key 교환과 메시지 신호에서 광범위하게 사용되는 방식이다.

5.1.15 *Secret Key Cryptography (SKC)* (암호 키 암호화) — 이는 Symmetric Key Cryptography (대칭 키 암호화)라고도 하며, 데이터에 대한 암호화(encryption)와 암호 해독(decryption)에 한 개의 키(key)만을 사용하는 암호화 알고리즘(cryptographic algorithm)이다. SKC 는 PKC 보다 빠르다는 이점이 있어 벌크 암호화(bulk encryption)에 광범위하게 사용된다.

5.1.16 *Session* (세션) — 본 문서에서 별도의 언급이 없다면 ‘session’은 *Authenticated Session* 을 의미하며, 다른 곳에서 정의된 session 들과는 무관하다.

5.1.17 *Symmetric Key Cryptography* (대칭 키 암호화) — Secret Key Cryptography 참조.

6 규약 (Conventions)

6.1 *Unified Modeling Language (UML) Notation* — 본 표준에서 사용된 모든 클래스와 시퀀스 다이어그램은 Unified Modeling Language 표기법을 따른다.

6.2 *Attribute Tables* (속성 테이블) — 본 표준에서 정의된 클래스의 속성을 정의하고 목록화 하기 위해 사용되는 테이블의 예는 다음과 같다.

Table 1 속성 테이블 포맷 (Attribute Table Format)

<i>Attribute Name</i>	<i>Definition</i>	<i>Form</i>
		아래 목록 참조

6.2.1 *Form* (형태) — 속성의 데이터 유형을 정의한다. 이 열에서 사용되는 데이터 유형을 설명하는 데 사용되는 용어들은 SEMI International Standards Compilation of Terms, 혹은 표준의 일부로 포함되어 있다. compilation of terms 를 참조하여 SEMI type 이름에 대한 의미를 확인 가능 하다.

6.2.2 *Attribute Table* (속성 테이블) — 본 표준에서 정의된 클래스 간의 연관을 설명하고 목록화 하기 위해 사용되는 테이블의 예시는 다음과 같다.

Table 2 연관 테이블 형식 (Association Table Format)

<i>Association Role Name</i>	<i>Definition</i>	<i>Comments</i>

6.2.2.1 *Association Role Name* (연관 역할 이름) — 명시된 연관의 역할 이름이다.

6.2.2.2 *Definition* (정의) — 연관의 목적 혹은 기능을 설명한다.

6.2.2.3 *Comments* (코멘트) — 연관에 대한 추가적인 설명이다.

6.3 *Operation Definition Tables* (작업 정의 테이블) — 본 표준에서 정의된 인터페이스 작업(operation)을 설명하고 목록화 하기 위해 사용되는 테이블의 예시는 다음과 같다.

Table 3 동작 정의 (Operation Definition)

<i>Operation</i>	<i>Description</i>	<i>Type</i>	<i>Requestor/Sender</i>	<i>Responder/Receiver</i>
		아래 목록 참조		

6.3.1 *Operation* (작업) — 작업(operation)의 이름이다.

6.3.2 *Type* (유형) — 작업(operation)의 메시징 시맨틱(messaging semantics)을 말한다. 본 표준에서는 Request-Response(RR)과 Fire-and-Forget(FF) 시맨틱(semantics)만이 사용된다.

6.3.3 *Requestor/Sender* (요청자/전송자) — RR 시맨틱은 요청을 하는 개체(entity)를 식별하며 FF 시맨틱은 메시지를 전송하는 개체(entity)를 식별한다. ‘Client’, ‘Admin’, 그리고 ‘Equipment’ 중 하나일 수 있으며, 클라이언트는 본 표준을 준수하는 장비와 통신하는 모든 어플리케이션이 해당될 수 있다.

6.3.4 *Responder/Receiver* (응답자/수신자) — RR 시맨틱은 요청에 응답하는 개체(entity)를 식별하며, FF 시맨틱은 메시지를 수신하는 개체(entity)를 식별한다. ‘Client’, ‘Admin’, 혹은 ‘Equipment’ 중 하나일 수 있다.

6.4 *Operation Argument Definition Table* (작업 인수 정의 테이블) — 본 표준에서 정의된 인터페이스 작업(operation)에 대한 인수를 설명하고 목록화 하는데 사용되는 테이블은 다음과 같다.

Table 4 동작 인수 정의 (Operation Argument Definitions)

Argument	Description	Kind	Form
		아래 목록 참조	아래 목록 참조

6.4.1 *Argument* (인수) — 인수의 이름을 명시한다.

6.4.2 *Kind* (종류) — 작업(operation)에 대해 ‘in’, ‘out’, ‘error’ 인수 중 어떤 인수인지 명시한다. ‘error’ 인수는 항상 ‘out’ 인수로서의 기능을 하지만, 작업(operation)이 성공적으로 완료되지 않았음을 의미한다. 특정 작업(operation)에 대해 발생 가능한 모든 오류(error)가 목록화되지는 않지만 대표적인 몇몇 항목들이 여기에 나열된다.

6.4.3 *Form* (형태) — 인수의 데이터 유형을 정의한다. 이 열에서 사용되는 데이터 타입을 설명하는 데 사용되는 용어들은 SEMI International Standards Compilation of Terms, 혹은 표준의 일부로 포함되어 있다. compilation of terms 를 참조하여 SEMI type 이름에 대한 의미를 확인 가능하다. .

7 배경 (Background)

7.1 SEMI E30 에서 정의하는 일반적인 반도체 장비 통신에서는, 단일 클라이언트 어플리케이션만이 통신을 위해 장비에 접근 가능하다. 이는 물리적 전송 수단(transfer medium)이 장비에 하나의 시리얼 케이블 연결(serial cable connection)일 때 유효하다. 이러한 환경에서, 어플리케이션이 실행되는 머신(machine)으로의 시리얼 연결만을 제공함으로써 하나의 클라이언트 어플리케이션에 대한 장비와의 인터페이스만이 허용된다.

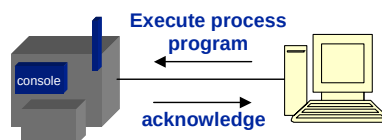


Figure 1
시리얼 케이블을 통한 장비 통신 (Equipment Communication via Serial Cable)

7.2 장비 통신이 시리얼 케이블(serial cable medium)에서 Ethernet 으로 이동함에 따라, 클라이언트 노드(node)에서 장비로의 경로가 제공 된다는 조건 하에서 네트워크 상의 모든 클라이언트가 장비에 요청을 전송할 수 있다. 장비와의 통신을 안전하게 할 특정 조치 없이는, 권한이 부여된 클라이언트만이 장비와의 통신이 가능하도록 하는 것이 불가능하다.

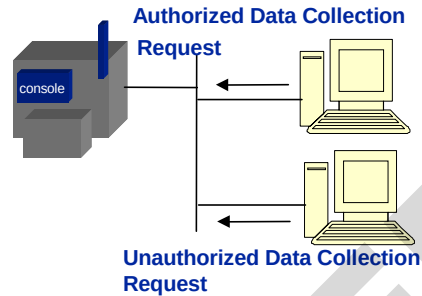


Figure 2
이더넷을 통한 장비 통신 (Equipment Communication via Ethernet)

7.3 장비가 만일 네트워크로부터 단일 연결 요청만을 받는다고 하더라도, 연결을 시도하는 많은 클라이언트 사이의 레이스(race) 조건에는 취약하다. 이는 연결을 최초로 시도하는 클라이언트가 장비와 통신할 권한이 없는 경우에도 접근 권한을 부여 받은 ‘first-come, first-served’ 규정 때문으로, 이러한 환경은 장비 리소스(resource)를 충분히 보호할 수 없다.

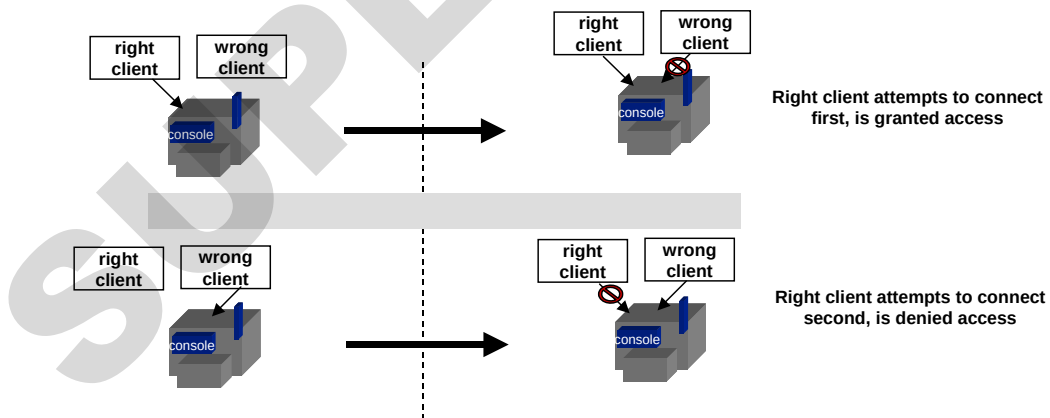


Figure 3
First-come First-served 통신 (First-Come First-Served Communication)

7.4 인증된 클라이언트만이 장비 리소스(resource)를 활용하도록 하는 보다 확실한 방법은 클라이언트가 우선적으로 자신을 증명하도록 하는 방식이다. 만일 장비가 클라이언트의 ID (identity)를 알고 있다면, 장비는 클라이언트가 추가 요청을 처리할 수 있는 권한을 가지고 있는지를 판단 가능하다.

8 개요 (Overview)

8.1 본 표준은 제조 장비의 클라이언트가 연속적인 통신에 앞서 장비에 대한 인증을 실시하는 메커니즘을 정의한다. 이를 위해, 인증 절차와 장비가 클라이언트를 식별할 수 있는 수단이 사전에 정의되어야 한다. 본 표준은 중앙 집중화된 관리 모델을 지원하기 위한 메커니즘과 권한 부여에 대한 유연한 스키마(scheme)를 정의한다.

8.2 *Central Administration Model* (중앙 관리 모델) — 중앙 보안 관리 모델(central security administration model)은 아래 Figure 4와 같은 방식으로 개념화된다.

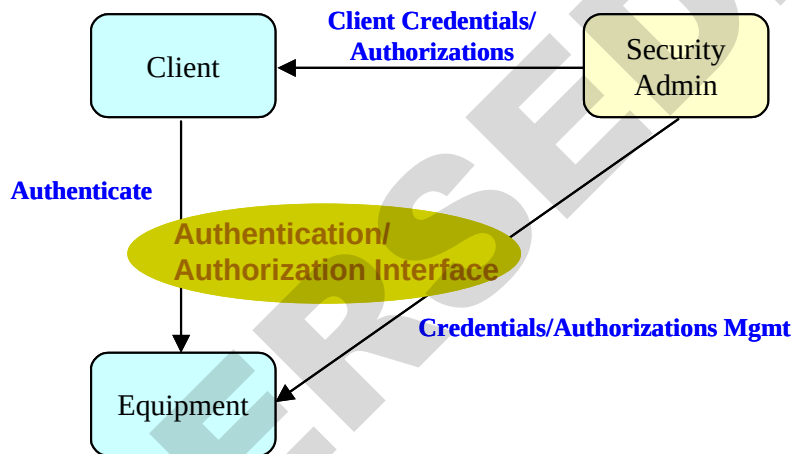


Figure 4

보안 관리 모델 (Security Administration Model)

8.2.1 *Security Admin* (보안 관리자) — 보안 관리자(security admin)는 클라이언트의 자격을 발행하며, 클라이언트의 권한 및 가능한 경우 장비에 클라이언트의 권한을 설정하고, 임의의 역할과 그들의 권한을 정의한다. 클라이언트 자격은 클라이언트에 대한 식별 정보이다. 보안 관리자는 본 표준에서 정의한 관리(administration) 인터페이스를 사용하여 장비 권한과 클라이언트 자격을 관리한다. 자격을 확인하기 위해 보안 관리자와 클라이언트 사이를 통신하는 인터페이스는 본 문서에서 정의되지 않으며, 클라이언트 어플리케이션의 내부 구성과 같이 단순한 형태일 것이다.

8.2.2 *Client* (클라이언트) — 클라이언트는 자신에게 할당되어 발행된 모든 자격을 가지고, 인증된 세션이 설정되도록 장비에게 자신을 인증한다.

8.2.3 *Equipment* (장비) — 관리 모델에서 장비의 역할은 장비에 대한 보안 관리 개체를 설정하고, 정의된 권한을 가지며, 세션을 설정하도록 클라이언트를 인증하고, 클라이언트 권한 레벨을 결정하며, 설정된 세션에 대해 이를 시행하는 것이다.

8.2.4 *Interfaces* (인터페이스) — 인증, 관리, 세션 설정 인터페이스는 클라이언트 ID(identities)와 권한을 설정하는 컨셉과 함께 다음 세션에서 정의되나, 실제 사용된 기술(technology)과 프로토콜(protocols)은 다른 관련 표준에서 자세히 정의된다.

8.3 접속 시도/응답 인증 (*Challenge/Response Authentication*)

8.3.1 인증 프로세스는 클라이언트가 장비와의 통신 세션을 최초로 시도할 때 시작된다. 이 요청을 수신하면, 장비는 클라이언트에게 ID(identity) 증명을 시도한다. 장비는 생성된 세션 데이터를 개인(private) 혹은 암호 키(예. password)를 사용하여 암호화함으로써 요청한 클라이언트만이 알 수 있도록 한다. 장비는 클라이언트에게 할당된 공개 키(public key) 혹은 공유된 암호 키(secret key)를 사용하여 데이터를 암호 해독하는 클라이언트의 요청을 검증할 수 있는데, 만일 암호 해독 텍스트가 일반 텍스트와 일치한다면 클라이언트가 확인된 것이다.

8.3.2 분리된 외부 접속(outgoing connection)에서의 통신 기술의 경우, 장비가 이벤트를 전송하기 위해서는 장비는 반드시 수신 주소가 유효한지 검증해야 한다. 이는 클라이언트 종단점(endpoint)에 핑 메시지(ping message)를 전송하여 확인 가능하다(§ 14.1.1 참조). 본 표준은 현재 클라이언트가 설정된 세션을 다른 종단점으로 이동 시킬수 있는 메커니즘을 정의하지 않는다.

8.4 권한 부여 (*Authorization*)

8.4.1 장비상의 서비스를 사용하고자 하는 클라이언트 어플리케이션은 반드시 해당 권한을 가져야 한다. 정의된 클라이언트에게 사전에 접근 컨트롤(access control) 권한을 부여한 후 클라이언트의 요청을 확인하여 권한 부여(Authorization)가 구현된다. 일반적인 동작 흐름(operational flow)은 다음과 같다:

- *Authentication and Credentials Exchange* (인증 및 자격 교환) — 클라이언트 어플리케이션이 장비에 해당 ID(identity)를 인증하고 인증된 세션을 설정한다.
- *Session Established* (세션 설정) — 장비가 선택적으로 클라이언트 어플리케이션의 권한을 캐시(cache)하고 그 권한을 새롭게 생성된 세션과 연관시킨다.
- *Service Request* (서비스 요청) — 클라이언트 어플리케이션이 서비스 요청을 한다.
- *Privilege Validation* (권한 검증) — 장비가 클라이언트 어플리케이션의 요청을 그에 할당된 권한에 대해 검증한다.
- *Session Closed* (세션 종료) — 클라이언트 어플리케이션이 세션을 종료하고, 모든 세션 상태 정보가 초기화되거나 삭제된다.

8.4.2 클라이언트가 각각의 서비스 요청에 대해 자신을 인증하지 않기 때문에, 권한 검증은 해당 클라이언트에 대한 인증된 세션에 대해 수행된다. 이는 세션의 구현이 상태 기반(stateful) 혹은 접속 지향형(connection-oriented) 통신 프로토콜을 사용해야만 한다면, 인증이 모든 프로토콜 단계의 재 연결 중 수행 되어야 한다는 의미는 아니다. 만약 구현 프로토콜이 접속되지 않았다면, 세션은 세션 생성에 필요한 인증 프로세스 중 협의된 세션 식별자의 상호 확인으로 식별될 수 있다.

NOTE 1: 본 표준은 최종사용자가 장비 서비스 혹은 정보에 컨트롤 할 수 있는 메커니즘은 정의하지 않는다. 본 표준에 정의된 인증 및 허가 스키마(scheme)는 어플리케이션 단계에만 적용된다. 최종 사용자의 장비 기능 및 정보에 대한 접근 관리는 본 표준에서 정의된 인증 통신 세션을 활용하는 어플리케이션의 권한이다.

9 인증 (Authentication)

9.1 *Authentication Message Flow* (인증 메시지 흐름) — 일반적인 세션 설정 중의 인증 메시지 흐름(authentication message flow)은 Figure 5 에서 설명된다. 화살표는 핸드셰이크(handshake) 도중에 교환되는 주요 메시지를 나타내며, 옆의 설명들은 각 메시지에서 교환되는 주요 데이터를 명시한다. 실제 사용된 메시징/전송 프로토콜과 암호화 알고리즘에 따라, 몇몇 추가 메시지 혹은 데이터 매개 변수(parameter)가 추가될 수 있지만, 개념은 변하지 않는다. 알고리즘과 프로토콜의 세부 사항들은 연관된 별도의 기술 표준에서 설명된다. 용어 서버(server)와 장비(equipment)는 차이 없이 사용된다.

9.1.1 *Session Request* (세션 요청) — 본 메시지는 인증된 세션 설정 시작을 장비에게 요청한다. 이 메시지를 수신하면, 장비는 Figure 5 에 보여진 것과 같이 인증 핸드셰이크를 시작한다. 만일 요청이 수락되지 않으면, 장비는 특정 프로토콜 경고 혹은 오류 메시지로 응답하고 접속을 종료한다. 관련 내용은 장비 접속 시도 부분을 참조한다.

9.1.2 *Equipment Challenge* (장비 접속 시도) — 세션 요청이 수락된 이후, 장비는 접속 시도에 응답한다. 장비는 인증에 사용될 알고리즘을 선택 하고(적용 가능한 경우), 해당 식별 정보를 제공하고 클라이언트가 자신을 인증할 수 있도록 접속 시도한다. 이는 클라이언트가 자신을 증명하기 위한 간단한 요청이며, 일반 매개 변수(parameter) 대신에 오류가 반환될 수 있음에 유의해야 한다. 개념을 지원하기 위해 사용되는 구현 프로토콜의 요구로 추가적인 오류가 필요할 수도 있다.

9.1.3 *Client Authenticate* (클라이언트 인증) — 클라이언트가 서버의 접속시도에 응답하여, 선택된 알고리즘을 확인하고(적용 가능한 경우), 장비를 식별한 후 식별 클라이언트의 ID(identity)에 대한 증거를 제시한다. 이는 일반적으로 클라이언트만이 알고 서버가 검증 가능한 키(key)로 암호화된 임의의 데이터 해시(hash)로 증명한다. 인증 프로토콜에 있어 고유한 암호는 클라이언트가 생성하여 응답으로 장비에 전송될 수 있다. 이 암호의 교환에 사용되는 실제 알고리즘은 관련 표준에서 명시된 특정 인증 프로토콜에 의해 정의된다.

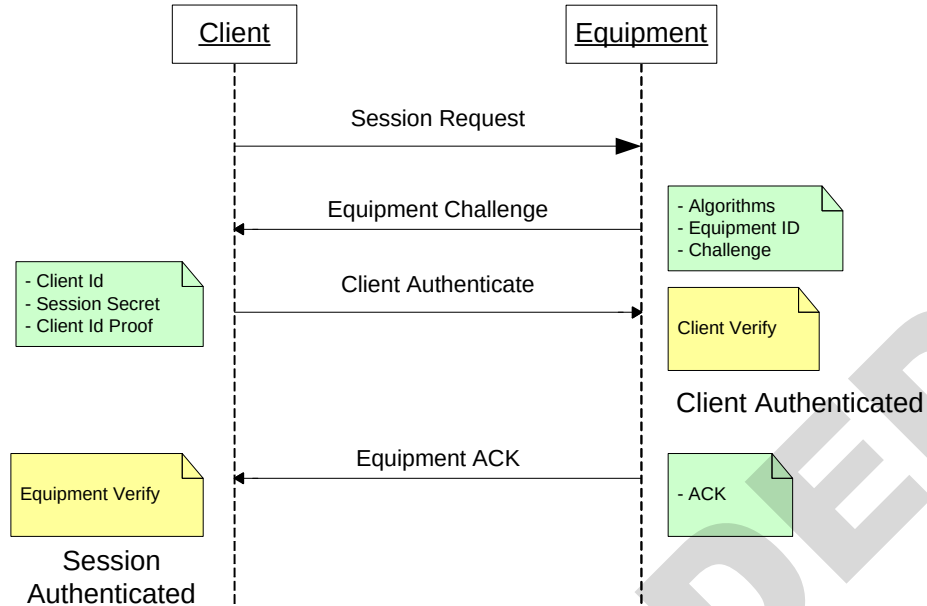


Figure 5
인증 메시지 흐름 (Authentication Message Flow)

Table 5 장비 접속 시도 매개 변수 정의 (Equipment Challenge Parameter Definitions)

Parameter	Description	Kind	Form
Algorithms	키(key) 교환, 한 방향 해시 기능과 데이터 암호화에 사용될 암호화 알고리즘을 식별한다.	OUT	사용된 알고리즘을 식별하기 위해 필드가 정해진, Text.
Equipment Id	선택된 프로토콜에서 정의된 동일한 포맷의 장비의 ID(identity)이며, 암호 기반 스키마(scheme)에서와 같이 이름 식별자와 같이 간단할 수 있다.	OUT	Text
Challenge	클라이언트가 자신을 인증하도록 하는 요청이며, 클라이언트가 서명할 랜덤 데이터(random data)를 포함하며 반복 재생(reply)이 금지된다.	OUT	Text
error	예상치 못했거나 적절하지 못한(Illegal) 메시지이다.	error	사용된 인증 프로토콜에 의해 정의됨.

Table 6 클라이언트 인증 매개 변수 정의 (Client Authenticate Parameter Definitions)

Parameter	Description	Kind	Form
Client Id	선택된 프로토콜에 의해 정의된 형태의 클라이언트의 ID(identity)이며, 암호 기반의 스키마(scheme)에서의 이름 식별자와 같이 간단할 수 있다.	IN	Text

Session Secret	인증 프로세스에서 고유하게, 랜덤으로(randomly) 생성된 데이터이다. 장비의 공개 키(public key) 혹은 공유된 암호 키(shared secret key)를 통해 암호화되는데, 사용된 실제 프로토콜과 키 교환(key exchange)을 위해 선택된 알고리즘에 따라 달라진다.	IN	암호화된 Binary
Client Id Proof	해당 클라이언트만이 알 수 있는 암호를 사용하여 임의의 데이터를 암호화한 것으로, ID(identity)에 대한 클라이언트의 증명 가능성이 가능하며, 서명으로써 다뤄진다. 랜덤 데이터(random)는 장비의 접속 시도로부터 혹은 그로부터 파생된 데이터일 수 있다.	IN	암호화된 Binary
error	예상치 못했거나 규칙을 벗어난 (illegal) 메시지이다.	error	사용된 인증 프로토콜에 의해 정의됨

9.1.4 *Equipment ACK* (장비 ACK) — 장비가 클라이언트의 응답을 검증하고 본 최종 인증 프로토콜 메시지 내에서 교환된 세션 암호를 확인한다. 응답(acknowledgement)은 파생된 암호 혹은 암호의 해시(hash)에 해당하는 매개 변수(parameter)를 포함한다. 파생된 세션 암호에 대한 파생 알고리즘(derivation algorithm)은 장비와 클라이언트 모두에게 알려지며, 프로토콜 표준의 일부이다. 이를 통해 클라이언트가 해시(hash)의 검증으로 교환을 확인할 수 있다. 이후 세션을 통한 통신은 파생된 암호 키(key)를 사용한다. 본 표준은 데이터가 설정된 세션을 통해 암호화할 필요가 없으며, 파생된 암호가 세션을 식별하고 인증된 클라이언트의 권한 부여를 위해서만 사용된다는 점에 유의해야 한다. 선택된 프로토콜 혹은 사용된 구현 기술의 필요에 의해 추가적인 오류가 정의될 수 있다.

Table 7 장비 ACK 인수 정의 (Equipment ACK Argument Definitions)

<i>Parameter</i>	<i>Description</i>	<i>Kind</i>	<i>Form</i>
ACK	교환된 세션 암호로부터 파생된 데이터의 해시(hash)이며, 이를 통해 장비가 암호를 올바르게 해독했는지 확인한다. 핸드셰이크(handshake) 시작 시 동의된 해시 알고리즘(hash algorithms)이다.	OUT	암호화된 Text.
error	예상되지 않았거나 적절하지 않은 메시지(illegal message).	error	사용된 특정 인증 프로토콜에 의해 정의.
error	적절하지 않은 매개 변수(illegal message).	error	사용된 특정 인증 프로토콜에 의해 정의.
error	인증 실패.	error	사용된 특정 인증 프로토콜에 의해 정의.

9.2 세션 식별자 (Session Identifier)

9.2.1 EstablishSession 요청(¶ 12.2.5 참조)으로 반환된 세션 id는 이후의 통신에서 세션을 식별하고 권한을 부여할 때 사용될 수 있다. 식별자는 메시지 전송에서 특수 헤더(special head)에 첨부되고, 통신 메시지의 형식 매개 변수(formal parameter)로 포함되지 않아 각각의 인터페이스 표준에서 정의한 원본 구조(original structure)를 유지하게 된다.

10 권한 부여 (Authorization)

10.1 권한 부여 개요 (Authorization Overview)

10.1.1 장비에서 권한이 필요한 서비스를 사용하고자 하는 모든 클라이언트는 권한을 부여 받아야 한다. 권한 부여는 정의된 클라이언트에게 우선 접근 컨트롤 권한을 부여하여 구현되고, 그 이후에 클라이언트의 요청에 대한 검증을 통해 이루어진다. 사용 가능한 모든 권한은 서비스 단계의 접근 컨트롤 권한을 포함하고, 이는 특정 서비스나 기능을 실행시키는 권한과 같은 것들이며, 또한 장비 제조사가 정의한 더 세부적인 단계의 권한을 포함할 수도 있다. 인증된 세션에서 가능한 모든 서비스에 대해 접근이 허용되는 디폴트(default) 권한은 이 표준에서 정의되며, 모든 구현에서 지원된다. 그러므로 장비 사용자는 장비에 대한 보안 요건을 인증에만 한정해서 줄일 수 있는 선택권을 가지고 있으며, 이는 정의된 모든 클라이언트에 대해 모든 접근 권한을 지정함으로써 가능하다.

10.1.2 Figure 6은 본 표준에서 정의된 Role, Principal, Privilege 사이에서의 관계에 대한 추상적인 개요를 보여준다. Principal은 오로지 하나의 Role에만 지정되거나, 하나 혹은 그 이상의 Privilege에 해당된다. Role은 하나 혹은 그 이상의 Privilege에 대해 고유하게 식별할 수 있는 집합으로 정의되며, 다른 Role으로부터 구성될 수 없다. 0개 혹은 그 이상의 Principal이 최대한 한 개의 Role에 지정될 수 있다. 이러한 관계는 Access Control List 엔트리(entry)의 관리를 통해 시행될 수 있으며, ¶10.2에서 설명된다.

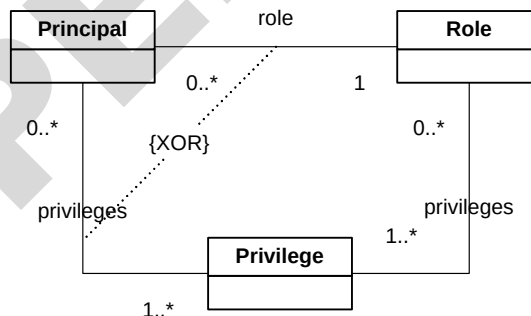


Figure 6

ACL의 추상 모델 (Abstract Model of ACL)

10.2 접근 컨트롤 목록 (Access Control List)

10.2.1 접근 컨트롤 권한(access control privilege)은 Access Control List(ACL)을 사용하여 정의된 클라이언트(Principal)에 지정된다. ACL은 ACL 엔트리(entry)의 집합(collection)이며, 각 엔트리(entry)는 특정 주체(subject)에 대해 가능한 권한의 하위 집합(subset)을 허용한다. 주체(subject)는 특정 principal, 고유한 이름을 가진 role, 혹은 장비에 대해 성공적으로 인증된 모든 principal들을 대표할 수 있다.

Principal 은 명시된 권한 목록이나 role 으로의 지정을 통해 간접적으로 권한을 부여 받을 수 있다. 장비는 본 표준에서 정의한 ACL 을 비 휘발성 메모리에 저장해야 한다.

10.2.2 Role 은 Principal 에 대한 권한 지정 관리를 단순화하기 위해 사용된다. 각각의 고유한 Role 은 ACL 엔트리(entry)를 가지고 있으며, 해당 role 에 적용된 권한을 명시한다. 이러한 권한은 이들 권한을 가져야 하는 각각의 principal 에 대해 RoleAssignment ACL 을 통해 권한이 부여 될 수 있다. Principal 은 RoleAssignment 혹은 이들에게 적용된 명시적 권한의 목록(PrivilegeAssignment)을 가지고 있어야 한다; 두 항목 모두를 가질 수는 없다.

10.2.3 장비는 Role 혹은 권한 배정 통한 직접적인 혹은 인증된 모든 principal 에 대한 ACLEntry 의 존재를 통한 간접적인 ACLEntry 가 없는 클라이언트의 세션 요청은 거절한다.

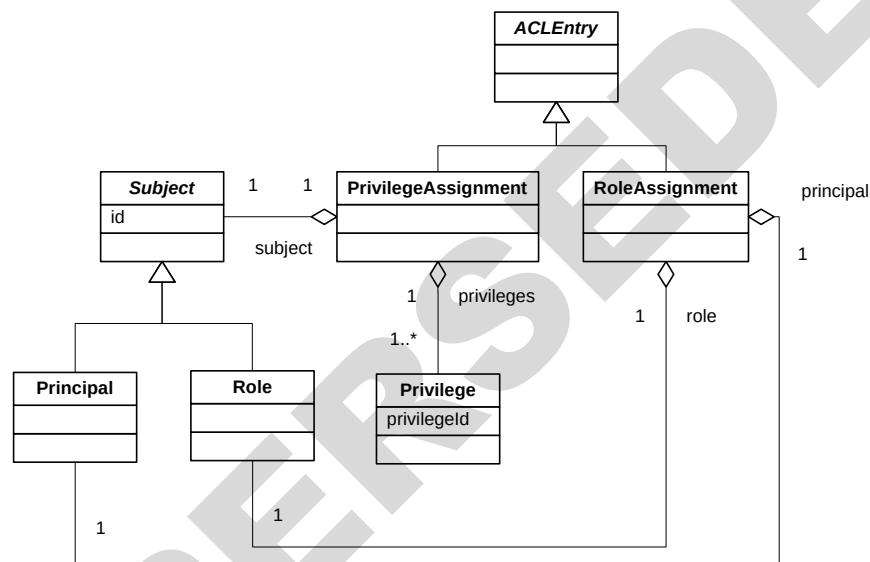


Figure 7
ACL 엔트리 설명 (ACL Entry Description)

10.2.4 ACLEntry

10.2.4.1 Figure 7 은 장비 상의 신규 ACLEntry 생성 요청에 사용되는 클래스를 보여준다 (§11.3.2.3 참조). ACLEntry 는 두 종류의 ACL 엔트리(entries), PrivilegeAssignment, RoleAssignment 에 대한 추상 기본 클래스이다. 각 주체(subject)에 대해 하나의 ACLEntry 만이 존재 가능하며, 동일 주체(subject)에 대해 두 번째 엔트리(entry) 추가를 시도하는 경우 오류가 발생한다 (§11.3.2 참조). 본 클래스는 속성과 연관이 존재하지 않는다.

10.2.5 PrivilegeAssignment

10.2.5.1 이 클래스는 주어진 주체(Principal 혹은 Role)와 장비 상에서 그에 배정된 권한 사이의 연관을 정의한다. Principal 에 대한 PrivilegeAssignment 는 principal 에 제한하여 권한이 부여되는 모든 권한을 설정한다. 주어진 Principal 은 PrivilegeAssignment ACLEntry 나 RoleAssignment ACLEntry 중 하나를 가져야 한다 (§10.2.6 참조); 각각에 대해 하나 혹은 그 이상의 항목을 가질 수는 없다. Role 에 대한