

SEMI E132-0310^{E2}

装置クライアントの認証(Authentication)および権限付与(Authorization)のための仕様

免責事項: この SEMI スタンダードは、投票により作成された英語版が正式なものであり、日本語版は日本の利用者各位の便宜のために作成したものです。万が一英語と日本語とに差異がある場合には英語版記載内容が優先されます。

本スタンダードは、global Information & Control Committee で技術的に承認されている。現版は 2009 年 12 月 16 日、global Audits and Reviews Subcommittee にて発行が承認された。2010 年 2 月に www.semi.org で入手可能となる。初版は 2004 年 3 月発行、前版は 2008 年 11 月に発行された。

注意: 本スタンダードは、2010 年 8 月、関連するスキーマファイル中の誤りを訂正するため、編集上の修正がなされた。変更箇所は、SEMI E132.1 XML WSDL files である。

1 目的

1.1 本仕様では、クライアントが通信を開始する前に装置に対してクライアントを認証するように要求することによって、装置との通信に対するアクセスを制限する方法について記述し、クライアントのアプリケーションアクセスを管理する柔軟性のある権限付与スキームを提供する。権限付与により、情報を交換したい当事者や装置を制御したい当事者が、知る必要がある情報および使用する必要がある情報に基づいて実行することが可能となる。

1.2 本スタンダードで指定する権限付与スキームにより、装置ベンダーは、アクセス管理制限無しから、定義済みの役割ベースのアクセス管理、非常にきめの細かい管理まで任意のレベルの細かさでアクセスを管理する柔軟性を提供できる。

2 範囲

2.1 これは、半導体製造装置のサービスに対して認証と権限付与を必要とするすべての半導体製造装置に適用されるスタンダードである。SEMI E30 通信モデルおよび制御状態モデルによって管理される通信には適用されない。

2.2 本スタンダードでは、確立されたセッションを通じ送信されるデータを暗号化する必要はなく、暗号化は認証プロトコルの指定によって必要な場合に限られる。本スタンダードによって指定されるインタフェースは、クローズドな工場内のネットワーク等、悪意のある攻撃がない環境で動作することが前提となる。

注意: このスタンダードはその使用に関連したすべての安全問題を取り扱うことを意図していない。このスタンダードの使用者は、その責任において、適切な安全および健康上の作業方法を確立し、また使用前に規制上その他の制限への適用性を判断するものである。

3 制限

3.1 特定の技術およびプロトコルを本文書内で定義される概念にマッピングする付属のスタンダードを定義する必要がある。

4 参照スタンダードと文書

4.1 SEMI スタンダード

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

SEMI E128 — Specification for XML Message Structures

4.2 その他のスタンダード

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>

注意: 別途指示のない場合, 引用されたすべての文書は最新の出版バージョンである。

5 用語

5.1 定義

5.1.1 管理 (*administration*) — 装置クライアントの識別と関連する資格を設定し, 権限付与を設定するプロセス。

5.1.2 非対称キー暗号化 (*Asymmetric Key Cryptography*) — 公開キー暗号化 (*Public Key Cryptography*) を参照。

5.1.3 認証された通信 (*authenticated communications*) — 認証されたセッションを確立, 管理するための権限を付与する本仕様に準拠した装置による通信。それには, 認証されたセッションのすべての通信が含まれる。認証された通信は必ずしも暗号化される必要はないことに注意すること。

5.1.4 認証されたセッション (*authenticated session*) — 本スタンダードでは, セッションとしても言及される。本スタンダードで定義されるアプリケーションレベルの概念であり, いかなる物理的, 論理的接続, また実際のメッセージ交換で使用される通信プロトコル「セッション」レイヤーとも無関係である。認証されたセッション, あるいはセッションは, 本文書が示すとおり, セッション確立手順, 状態モデル, 定義されたサービスに従う。

5.1.5 認証 (*authentication*) — ユーザやプロセスが実際に宣言されている者あるいは物であるかを決定するプロセス。プライベートおよびパブリックなコンピュータネットワークでは, しばしばパスワードを使用して認証を実行する。パスワードを知っていることが, そのユーザが認証可能であることを保証するものと仮定される。本スタンダードは, アプリケーションプロセスの認証のみを対象とするものであって, 人間のユーザの認証についてはサポートあるいは対処しない。

5.1.6 権限付与 (*authorization*) — ユーザやプロセスに対して特定のアクションを実行する権限を付与するプロセス。ユーザやプロセスは, 権限付与の前に, 認証されていなければならないことに注意する。

5.1.7 クライアント (*client*) — 装置と通信して装置のリソースのアクセスするアプリケーションプロセス。本仕様では, 「クライアント」と「クライアントアプリケーション」という用語を区別せずに使用する。

5.1.8 通信セッション (*communication session*) — クライアントと装置間で行われる一連の双方向のメッセージ交換。セッションは, 通常, クライアントの要求により開始, 終了する。

5.1.9 暗号化アルゴリズム (*cryptographic algorithm*) — 意味のあるデータからなんらかの意味のないデータへの変換であり, 暗号化と呼ばれるプロセス。変換が可逆であれば, 逆のプロセスは, 復号化と呼ぶ。通常は, 使用されたアルゴリズムおよびキーと呼ぶ何らかの追加情報がなければ, そのデータを計算によって正しく復号することは困難である。

5.1.10 暗号化 (*encryption*) — 暗号化アルゴリズム (*cryptographic algorithm*) を参照。

5.1.11 メッセージダイジェスト (*message digest*) — 一方向ハッシュ (*one-way hash*) を参照。

5.1.12 一方向ハッシュ (*one-way hash*) — 任意の長さのデータを固定長コードへ一方向変換すること。この変換を計算により逆算することは困難であり, コリジョンは起こりにくい。同じメッセージは, 常に同じハッシュとなり, 元のメッセージをわずかに変更すると, 異なったハッシュとなるのでメッセージは一貫性を有する。本文書では, 「一方向ハッシュ」と「ハッシュアンドダイジェスト」という用語を区別せずに使用する。

5.1.13 永続されたセッション (*Persistent Session*) — 装置の停止, あるいは電源オフを乗り切るように確立された認証されたセッション。装置停止, 起動時の永続セッションの挙動は, 本文書に示されるセッション状態モデルによって定義される。

5.1.14 公開キー暗号化 (*Public Key Cryptography (PKC)*) — 非対称暗号化 (*Asymmetric Key Cryptography*) としても知られ, データの暗号化と復号化に 2 つのキーを採用する暗号化アルゴリズムである。ひとつのキーは公開キーとして指定され, 自由に公開されるが, もうひとつのキーは秘密キーに指定され, 決して公開されない。ひとつ

のキーで暗号化されたデータは、もうひとつのキーによってのみ復号化できる。PKC は、キー交換やメッセージサインに広く使用されている。

5.1.15 秘密キー暗号化 (Secret Key Cryptography (SKC)) — 対称キー暗号化 (Symmetric Key Cryptography) としても知られ、暗号化と複合化の両方にひとつのキーを採用する暗号化アルゴリズムである。SKC は、PKC より速いので、大きなデータの暗号化に広く使用されている。

5.1.16 セッション (Session) — 本文書では、「セッション」という言葉は、別途指示がない限り認証されたセッションを意味する。別途定義されるであろうセッションとは無関係である。

5.1.17 対称キー暗号化 (Symmetric Key Cryptography) — 秘密キー暗号化 (Secret Key Cryptography) を参照。

6 表記法

6.1 統一モデル化言語 (Unified Modeling Language (UML)) による表記 — 本スタンダードのすべてのクラスおよびシーケンスダイアグラムでは、統一モデル化言語 (UML) による表記を利用する。

6.2 属性表 (Attribute Tables) — 次の表は、本仕様で定義されるクラスの属性を記載し、説明するために使用される表の例である。

表 1 属性表のフォーマット

属性名	定義	形式
		次のリストを参照

6.2.1 フォーム (Form) — 属性のデータタイプを定義する。この欄で使用されるデータタイプを説明するために使用される用語は、SEMI Compilation of Terms で定義されているか、仕様の一部として含まれている。SEMI の型名の意味の定義については、SEMI Compilation of Terms を参照すること。

6.2.2 関連付け表 (Association Tables) — 次の表は、本仕様で定義されるクラスの関連付けを記載し、説明するために使用される表の例である。

表 2 関連付け表のフォーマット

関連の役割名	定義	コメント

6.2.2.1 関連の役割名 (Association Role Name) — 指定されている関連付けの役割名。

6.2.2.2 定義 (Definition) — 関連付けの機能や目的を記述する。

6.2.2.3 コメント (Comments) — 関連付けに関する追加のコメントや注がある場合に記載。

6.3 操作定義表 (Operation Definition Tables) — 次の表は、本仕様で定義されるインタフェースの操作を記載し、説明するために使用される表の例である。

表 3 操作の定義

操作	定義	タイプ	要求元／送信元	応答元／受信元
		次のリストを参照		

6.3.1 操作 (Operation) — 操作の名前を指定する。

6.3.2 型 (Type) — 操作のメッセージの意味を指定する。本仕様では、要求-応答 (RR) および送り切り (FF) の意味だけが使用される。

6.3.3 要求元／送信元 (Requestor/Sender) — RR の意味については、要求を行うエンティティを識別し、FF の意味については、メッセージを送信するエンティティを識別する。‘Client’、‘Admin’、‘Equipment’のどれかである。クライアントは、本仕様に準拠する装置と通信する任意のアプリケーションでよい。

6.3.4 応答元／受信元 (Responder/Receiver) — RR の意味については、要求に応答するエンティティを識別し、FF の意味については、メッセージを受信するエンティティを識別する。‘Client’、‘Admin’、‘Equipment’のどれかである。

6.4 操作引数定義表 (Operation Argument Definition Table) — 次の表は、本仕様で定義されるインタフェースの操作の引数を記載し、説明するために使用される表の例である。

表 4 操作引数の定義

引数	定義	種類	フォーム
		次のリストを参照	次のリストを参照

6.4.1 引数 (Argument) — 引数名を指定する。

6.4.2 種類 (Kind) — 引数が操作の‘in’、‘out’、‘error’の引数であるかを指定する。‘error’の引数は常に‘out’の引数として機能するが、操作が正常に完了しなかったことを示す。特定の操作に起こり得るすべてのエラーが記載されるわけではなく、代表的なものが選択される。

6.4.3 フォーム (Form) — 引数のデータタイプを定義する。このコラムでデータタイプを説明するために使用される用語は、SEMI Compilation of Terms で定義されているか、仕様の一部として含まれている。SEMI のタイプ名の意味の定義については、Compilation of Terms を参照すること。

7 背景

7.1 SEMI E30 で定義されるような従来の半導体製造装置の通信では、ひとつのクライアントアプリケーションだけが装置にアクセスして通信するものと仮定されている。この仮定は、物理伝送媒体が装置へのシリアルケーブル接続である場合には有効であった。そのような環境では、アプリケーションが実行される装置へのシリアル接続を提供することにより、ひとつのクライアントアプリケーションだけが装置へインタフェースできるようにすることが可能である。

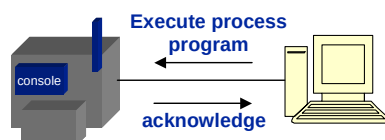


図 1
シリアルケーブルによる装置通信

7.2 装置通信がシリアルケーブルの媒体からイーサネットに移行すると、ネットワーク上の任意のクライアントは、そのクライアントノードから装置へのパスがあれば、装置へ要求を送信することが可能になる。装置との通信を安全に行える特定の手段をとらないと、権限を与えられたクライアントだけが装置と通信することはできなくなる。

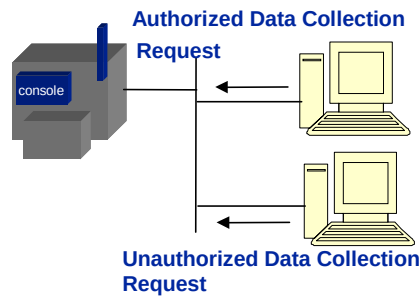


図 2

イーサネットによる装置通信

7.3 たとえ装置がネットワークからの単一の接続の要求だけを受け入れるとしても、接続しようとする多数のクライアントの競合状態に対しては依然として脆弱である。これにより、結果的に装置と最初に接続しようとしたクライアントに、たとえそのクライアントがその特定の装置と通信できる権限を与えられていなくても、アクセスを付与されるという先着順 (first-come, first-served) というポリシーとなる。そのような環境は、装置リソースを適切に保護するには不十分である。

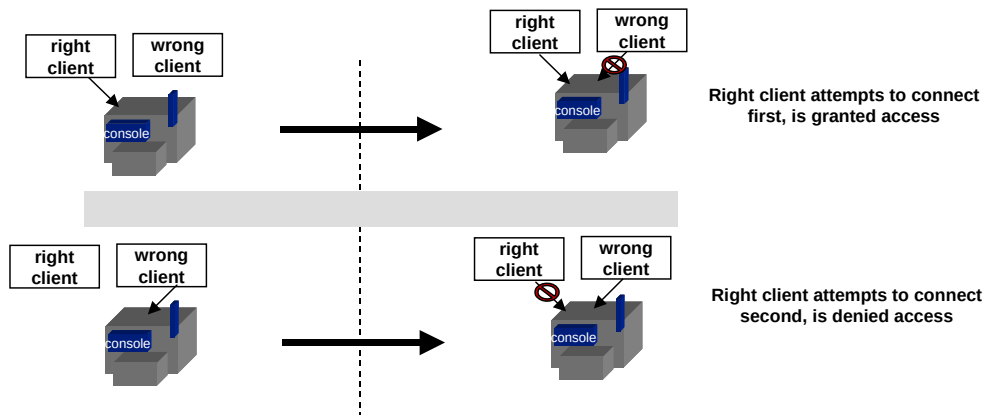


図 3

先着順 (first-come, first-served) 通信

7.4 権限を与えられたクライアントだけに装置のリソースの利用が許可されるようにする、より安全な手法は、先ずクライアントに対して装置に対する身元を証明するように要求することである。装置がクライアントの身元を知っている場合は、装置は、そのクライアントは次の要求に進むために権限を付与されるものと想定することができる。

8 概要

8.1 本スタンダードでは、製造装置のクライアントが通信を開始する前に装置に対して認証するためのメカニズムを定義する。このことを容易にするために、認証手続き自体およびその前の装置によってクライアントの身元を確定する手段の両方を定義しなければならない。本スタンダードでは、権限付与のための柔軟性のあるスキームおよび集中管理モデルをサポートするためのメカニズムについても定義する。

8.2 集中管理モデル— 集中セキュリティ管理モデルの概念を次の図 4 に示す。

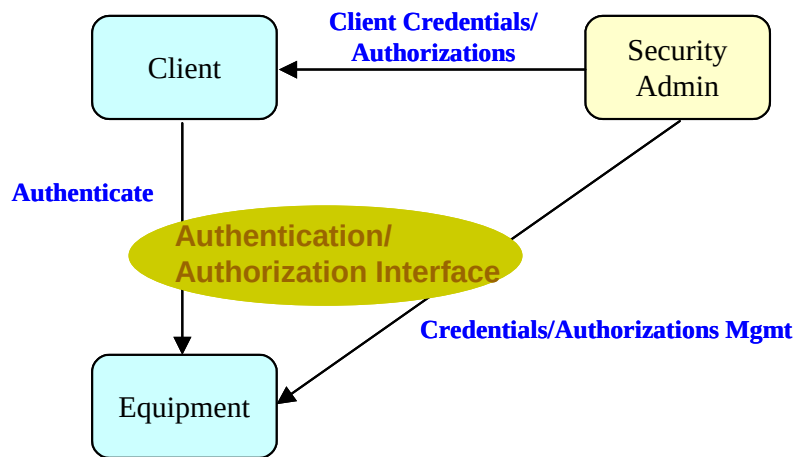


図 4
セキュリティ管理モデル

8.2.1 セキュリティ管理 (Security Admin) — セキュリティ管理は、クライアントの資格認定を発行し、クライアントの権限付与および可能ならば装置に関するクライアントの資格認定を確立し、カスタムの役割とそれらの権限付与を定義する。クライアントの資格認定は、そのクライアントの情報を識別している。セキュリティ管理は、本仕様で定義される管理インタフェースを使用して装置の権限付与とクライアントの資格認定を管理する。資格認定を通信するセキュリティ管理とクライアントの間のインタフェースは、本文書では定義せず、クライアントアプリケーション内の設定と同様にシンプルである可能性がある。

8.2.2 クライアント — クライアントは割り当てられた資格認定を保持し、装置に対して認証し、認証されたセッションを確立する。

8.2.3 装置 — 管理モデルにおける装置の役割は、装置のセキュリティ管理エンティティを確立し、定義された権限付与を保持し、クライアントを認証してセッションを確立し、クライアントの権限付与レベルを決定し、それらの確立されたセッションに強制することである。

8.2.4 インタフェース — 認証、管理、セッション確立のインタフェースについては、クライアントの身元と権限付与を確立する概念と共に、後述する項で定義する。しかし、実際に使用される技術とプロトコルの詳細は、別の付属スタンダードで定義されると思われる。

8.3 チャレンジ/レスポンスによる認証

8.3.1 クライアントが装置との通信セッションを最初に確立しようとした時に、認証プロセスが開始される。装置はこの要求を受信すると、クライアントに対して装置への身元を証明するようにチャレンジする。クライアントは、要求したクライアントだけが知っている公開キーまたは秘密キー（例えば、パスワード）を使用してセッションが生成する何らかのデータを暗号化しなければならない。装置は、データを復号するために割り当てられた、対応する公開キーまたは共有する秘密キーを使用してクライアントの要求を検証し、復号されたデータがプレーンテキストと一致するとクライアントは確認される。

8.3.2 装置がイベントを送信するために別の送信コネクションを確立しなければならない通信技術については、装置は、受信アドレスが有効であることも検証しなければならない。これは、クライアントのエンドポイントに対して、ping メッセージを送信することによって実行する（¶14.1.1 参照）。本仕様では、現在は、確立されたセッションをクライアントが異なるエンドポイントに送信できるメカニズムについては定義していない。

8.4 権限付与 (Authorization)

8.4.1 装置上の特権サービスを呼び出したいクライアントは、それを実行できるように権限を与えられなければならない。権限付与は、先ず定義されているクライアントに対してアクセス管理特権を付与することによって実装し、その後でクライアントの要求に対してチェックされる。一般的な動作フローは次の通りである。

- **認証と資格認定の交換** — クライアントアプリケーションが装置に対してその身元を認証し、認証されたセッションを確立する。
- **セッション確立** — 装置はオプションでクライアントアプリケーションの特権をキャッシュし、これらの特権を新たに作成されたセッションに関連付ける。
- **サービス要求** — クライアントアプリケーションは、サービス要求を実行する。
- **特権検証** — 装置は、クライアントアプリケーションに割り当てられている特権に対してクライアントアプリケーションの要求を有効にする。
- **セッションクローズド** — クライアントはセッションをクローズし、すべてのセッション状態情報は、リセットあるいはクリアされる。

8.4.2 クライアントはそれ自体を各サービス要求に対して認証するのではないので、特権の検証は、そのクライアントに作成された認証済みセッションに対して実行される。このことは必ずしも、セッションの実装に状態あるいはコネクション指向の通信プロトコルが必要であるということ、あるいは認証はプロトコルレベルのすべての接続中に実行しなければならないということではない。実装プロトコルがコネクションレスである場合、相互に合意したなんらかのセッション識別子によって識別することができ、セッション確立に必要な認証プロセス中に交渉される。

注 1: 本仕様では、装置サービスまたは情報へのエンドユーザのアクセスを管理するためのメカニズムについては定義しない。本仕様で定義される認証および権限付与のスキームは、アプリケーションレベルだけに適用される。装置の機能や情報に対するエンドユーザアクセスの管理は、完全に、本仕様で定義される認証された通信セッションを利用するアプリケーションの責任である。

9 認証 (Authentication)

9.1 **認証メッセージフロー** — 通常のセッション確立時の認証メッセージフローを図 5 に示す。矢印はハンドシェイク中に交換される主要なメッセージを示し、その説明文は各メッセージ内で交換される重要なデータを明記している。実際のメッセージ交換/送信プロトコルおよび使用する暗号化アルゴリズムに依存して、いくつかの追加メッセージやデータパラメータを追加できるが、考え方は同じである。アルゴリズムおよびプロトコル固有の内容については、別の技術マッピング仕様で詳述される。サーバと装置という用語は区別せずに使用する。

9.1.1 **セッション要求** — このメッセージにより、装置に認証セッション確立を起動するように要求する。装置はこのメッセージを受信すると、図 5 に示すように認証ハンドシェイクを開始する。要求が受け入れられない場合は、装置はプロトコル固有の警告あるいはエラーメッセージにより応答し、接続を終了する。装置チャレンジを参照。

9.1.2 **装置チャレンジ** — セッション要求が許容されると、装置はチャレンジによって応答する。装置は認証(該当する場合)に使用するアルゴリズムを選択し、それ自体の識別を提供して、クライアントに対して認証するようにチャレンジする。このチャレンジは、クライアントに身元の証明を要求するだけの簡易なものにできる。正常なパラメータの代わりにエラーが戻ることがあるということに注意する。この概念をサポートするために使用される実装プロトコルの必要性に応じて、追加のエラーが必要になることもある。

9.1.3 **クライアント認証** — クライアントはサーバチャレンジに応答し、選択されたアルゴリズム(該当する場合)と装置の身元を確認し、クライアント自体の身元証明を提示する。この証明は、通常は、クライアントだけが知っていてサーバが検証できるキーによって暗号化された何らかのランダムデータのハッシュである。クライアントによってセッションに固有の秘密情報も生成され、応答時に装置に送信される。この秘密情報を使って、ハンドシェイクの検証、その後の対称暗号化を使用したメッセージ交換、認証プロトコルに可能な識別子となる秘密セッションキーが派生する。この秘密情報の交換に使用される実際のアルゴリズムは、付属のスタンダードで定義される。

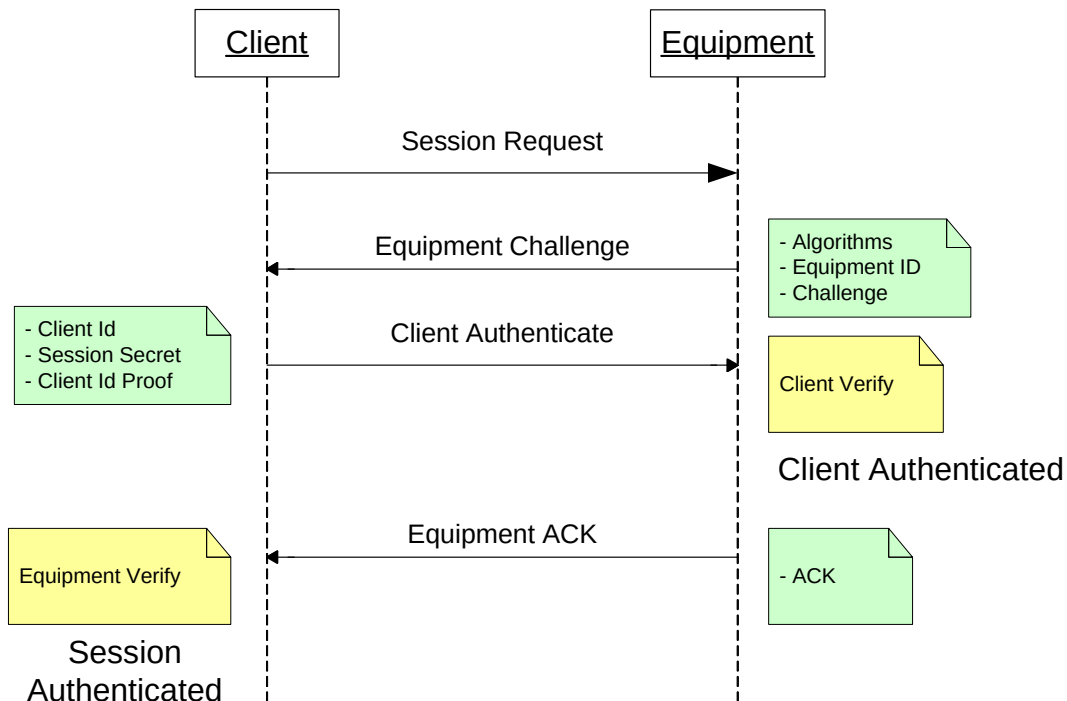


図 5
認証メッセージフロー

表 5 装置チャレンジパラメータの定義

パラメータ	定義	種類	フォーム
Algorithms	キー交換，一方方向ハッシュの機能，データ暗号化に使用する暗号化アルゴリズムを識別する。	OUT	テキスト，使用されるアルゴリズムを識別するための区切られたフィールド付き。
Equipment Id	装置の身元。選択されたプロトコルによって定義される正確なフォーマットであり，パスワードベースのスキーム等での名前識別子のように簡易にできる。	OUT	テキスト。
Challenge	クライアントに対するクライアント自体の認証要求。クライアントがサインするためのランダムなデータを含み，再生を防ぐ。	OUT	テキスト。
error	不正または予期しないメッセージ。	error	使用される特定の認証プロトコルによって定義される。

表 6 クライアント認証パラメータの定義

パラメータ	定義	種類	フォーム
Client Id	クライアントの身元。選択されたプロトコルによって定義される正確なフォーマットであり，パスワードベースのスキーム等での名前識別子のように簡易にできる。	IN	テキスト。
Session Secret	認証プロセスに固有なランダムに生成されるデータ。これは，キーの交換に使用される実際のプロトコルおよび選択されたアルゴリズムに依存し，装置の公開キーまたは確立された共有秘密キーを使用して暗号化される。	IN	Binary, encrypted.
Client Id Proof	このクライアントだけが知っている秘密情報を使った何らかのランダムデータの暗号，クライアントの身元の申し立てを提供する。これは，署名として機能する。ランダムデータは，装置のチャレンジからのデータまたはそれから派生したデータ。	IN	Binary, encrypted.

error	不正または予期しないメッセージ。	error	使用される特定の認証プロトコルによって定義される。
-------	------------------	-------	---------------------------

9.1.4 装置 ACK — 装置はクライアントの応答を検証し、この最終の認証プロトコルメッセージ内で交換されたセッションの秘密情報を確認する。この確認応答には、秘密情報のハッシュまたはその派生された秘密情報であるパラメータが含まれる。派生されたセッション秘密情報の派生アルゴリズムは、装置とクライアントの両方が知っていて、プロトコル仕様の一部である。これによりクライアントは、ハッシュを検証することにより情報交換を検証できる。セッションのその後の通信では、派生した秘密キーを利用する。このスタンダードでは、確立されたセッション全体にわたりデータを暗号化する必要はなく、派生された秘密情報はセッションの識別および認証されたクライアントの権限付与のみに使用することに注意する。選択されたプロトコルおよび実装に使用される技術の必要性に応じて、追加のエラーを定義できる。

表 7 装置 ACK の引数の定義

パラメータ	定義	種類	フォーム
ACK	交換されたセッション秘密情報から得られたデータのハッシュであり、これが秘密情報の装置の正しい暗号化を証明する。ハッシュアルゴリズムは、ハンドシェイク開始時に合意したアルゴリズム。	OUT	暗号化されたテキスト
error	不正または予期しないメッセージを受信した。	error	使用される特定の認証プロトコルによって定義される。
error	不正なパラメータ。	error	使用される特定の認証プロトコルによって定義される。
error	認証失敗。	error	使用される特定の認証プロトコルによって定義される。

9.2 セッション識別子

9.2.1 EstablishSession 要求 (¶12.2.5 参照) から返されたセッション id は、その後の通信の為にセッションを識別し、権限付与を強制できるセッションを識別するために使用される。この識別子は、メッセージ送信の特殊なヘッダに埋め込まれ、通信メッセージ内の正式なパラメータではないので、対応するインタフェース仕様によって定義される元の構造を維持できる。

10 権限付与 (Authorization)

10.1 権限付与の概要

10.1.1 装置上の特権サービス呼び出したいクライアントは、それを実行できるように権限を与えられなければならない。権限付与は、まず定義されているクライアントに対してアクセス管理特権を付与することによって実装し、その後でクライアントの要求に対してチェックされる。使用可能な特権の集合全体には、特定のサービスや機能呼び出す特権等、標準化されたサービスレベルのアクセス管理が含まれるが、サプライヤ定義の細かい特権も含むことができる。認証されたセッションから利用可能なすべてのサービスにアクセスできるデフォルトの特権は、本仕様で定義され、すべての実装がサポートしなければならない。従って、装置のユーザは、装置のセキュリティ強制を認証だけに減じるオプションがあり、それは定義されたすべてのクライアントにこの全アクセス特権を割り当てることによっておこなう。

10.1.2 図 6 は、本スタンダードで定義されるロール (Role)、プリンシパル (Principal)、特権 (Privilege) の概念の関係を示した抽象的な概要である。プリンシパルはひとつのロールに付与されるか、ひとつ、もしくは複数の特権に付与される。ロールは一意に認識できるひとつ、あるいは複数の特権のセットとして定義され、他のロールからは構成されない。ほとんどのロールには、ゼロもしくは複数のプリンシパルが付与される。これらの関係は¶10.2 に説明されるアクセス管理リストエントリを通して実行される。

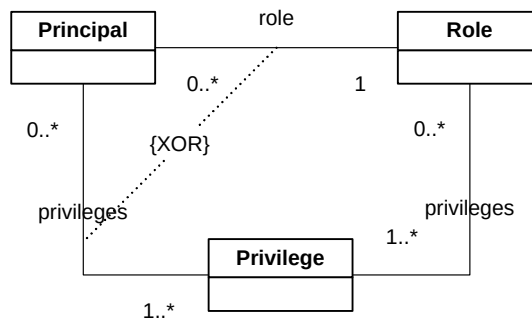


図 6
ACL のアブストラクトモデル

10.2 アクセス制御リスト

10.2.1 アクセス管理の特権は、アクセス制御リスト (Access Control List (ACL)) を使って、定義されたクライアント (プリンシパル) に割り当てられる。ACL は、ACL エントリのコレクションであり、各エントリは特定の対象に対して使用可能な特権のサブセットを付与する。対象は、特定のプリンシパル、ユニークな名前を持つ役割、または装置への認証が成功したすべてのプリンシパルを代表できる。プリンシパルは、特権の明示的リストによって付与される、あるいは役割への割当てによって間接的に付与される特権を持つことができる。装置は、本仕様で定義される ACL を不揮発性メモリ内に保持すること。

10.2.2 役割を使用して、プリンシパルに対する特権の割当て管理を簡略化できる。それぞれのユニークな役割には、その役割に割り当てられた特権を指定する ACL エントリがある。そして、これらの特権は、それを持つべき各プリンシパルについて、RoleAssignment ACL エントリにより、プリンシパルに付与できる。プリンシパルは、RoleAssignment またはそれに割り当てられた特権の明示的なリスト (PrivilegeAssignment) のどちらかを有していないと、両方持つことはできない。

10.2.3 役割や特権の割当てによる明示的なものであっても、認証されたすべてのプリンシパルに ACLEntry が存在することによる暗黙的なものであっても、装置は、ACLEntry がないクライアントからのセッション要求は拒否すること。

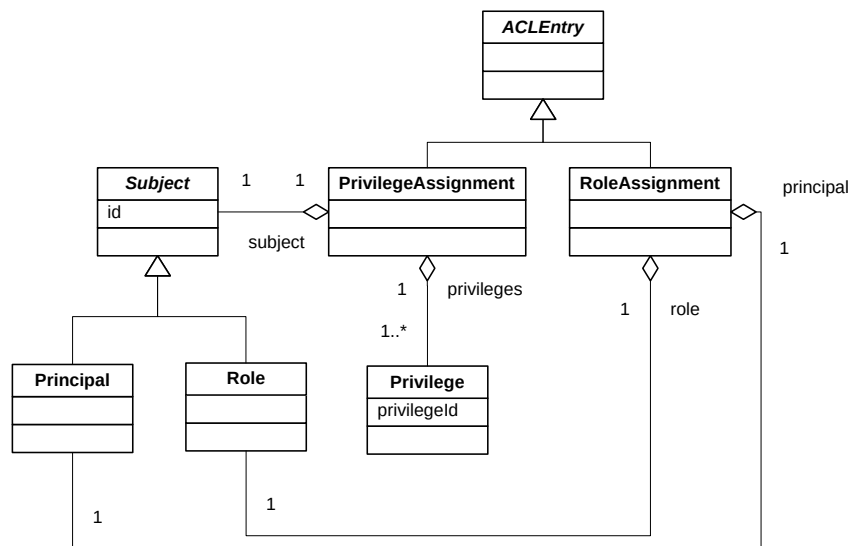


図 7
ACL エントリの説明