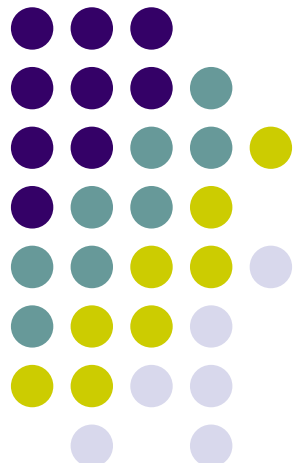
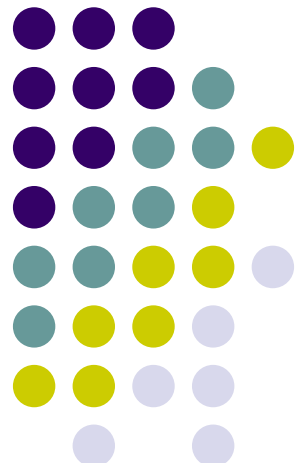
Securely Available Credentials - Credential Server Framework

渡邊研究室

００ｊ１２４　前羽　理克



1.Introduction



1. Introduction

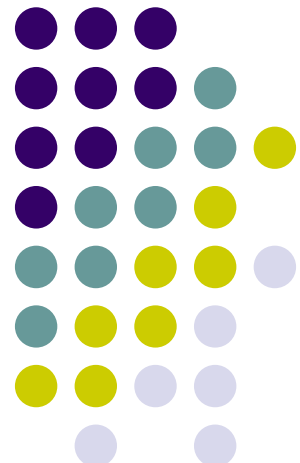
ネットワークの拡大及び多様化に伴い、
デジタル認証の需要が増してきている

エンドユーザが自分のクレデンシャル（一般的には鍵ペアや証明書の集まり）をデバイス等に関わらず、セキュアにどこからでも登録・取得・更新できる のが望ましい

＜このドキュメントでは、以下のことを提案する＞

- セキュアにどこからでも登録・取得・更新が可能な、標準フレームワーク
- 安全な認証交換プロトコルの発展のためのハイレベルなアウトライン

2. Functional Overview



2.1 Definitions

Client authentication information:

クライアント認証のために、クライアントからサーバに提供される情報

Credentials (証明書) :

暗号化目的とその関連データが、インターネット上での安全なコミュニケーションのサポートをするために使用される

Password token:

サーバを証明するためにクライアントによって使用されるパスワードから価値を引き出すもの

Secured credentials:

暗号化された安全な、ひとつまたは複数のcredentialsのセットをさす。また、暗号化レイヤより上の層を使用することによって守られている

Strong password protocol:

Serverにclientを安全に認証させるためのプロトコル

- パスワード等の秘密を記憶する
- 他の秘密情報を運ぶ

2.2 Credentials

Credential は 秘密性の高い情報を含んでいる

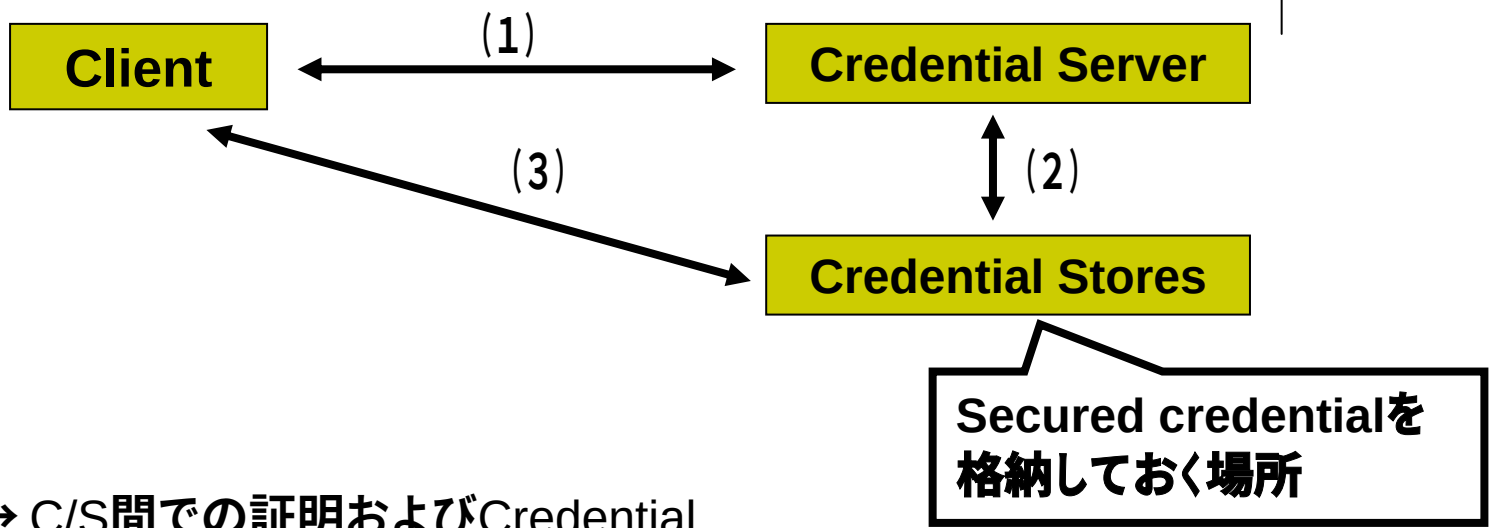
End user device は、mobile 環境下に置かれている

- network transmission **間で、クリアに送信してはならない**
- end user device **に格納する場合クリアな状態で行うべきではない**

Secure Credential を定義

- **プロトコル上では、ネットワークデバイスで使用できる不透明なデータオブジェクトを提供する**
- Network transmission **間では、**Second encryption layer **によって保護されている**

2.3 Network Architecture

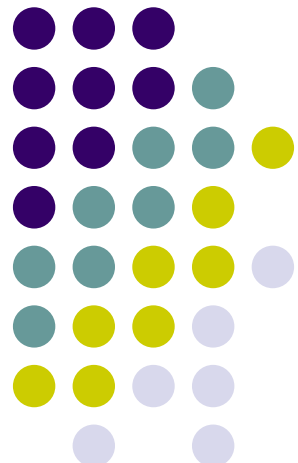


Protocol 1 → C/S**間での証明および**Credential Server**から**user credential **の** Up/Download **に使用される**

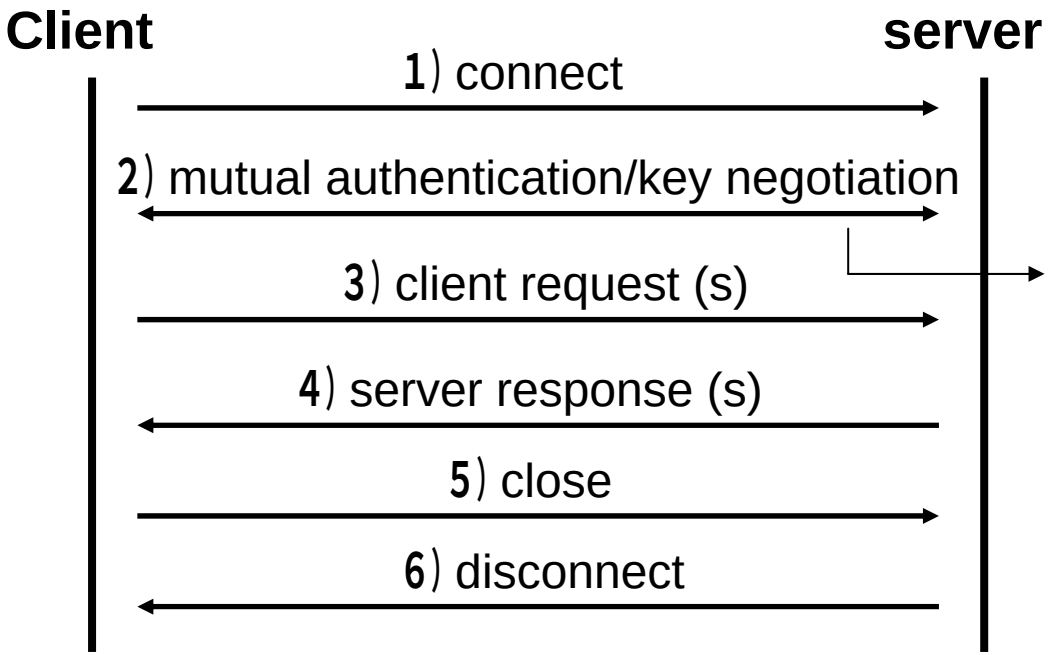
Protocol 2 →user credential **の保管/取り出しのために、** Credential Server **によって使用される**

Protocol 3 →Credential Stores **から** user credential **の保管/取り出しのために、**Client **によって使用される**

3.Protocol Framework



3.1 Secured credential exchange protocol

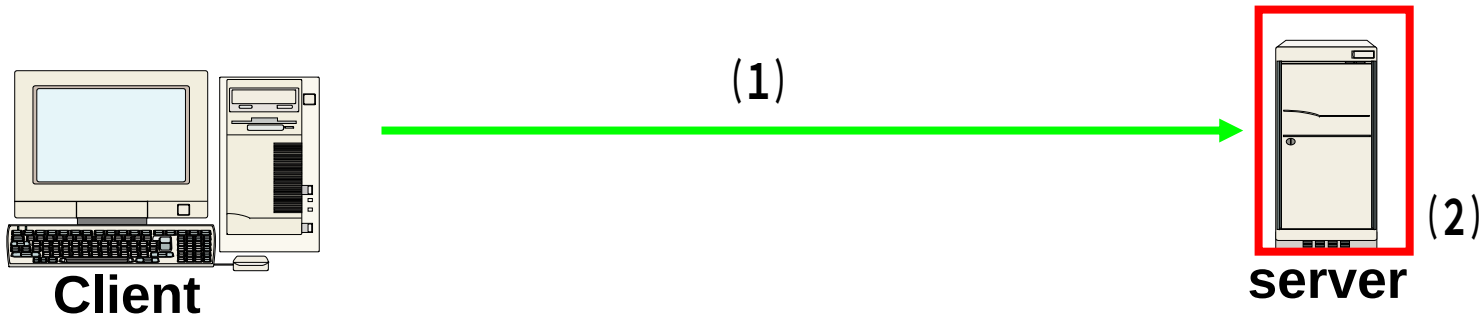


C/S 間で互いを証明する強力なパスワードプロトコルを使用し、セッションレベルの暗号化鍵の交換を行う

結果、C/S間で強力な秘密の共有をおこなうことになる

交信処理は、request-response 処理の連続で構成されている

3.2 Credential Upload



(1) アップロードするcredential とそれに関連するデータフィールドを含んでいる“Put”メッセージをクライアントより送信する

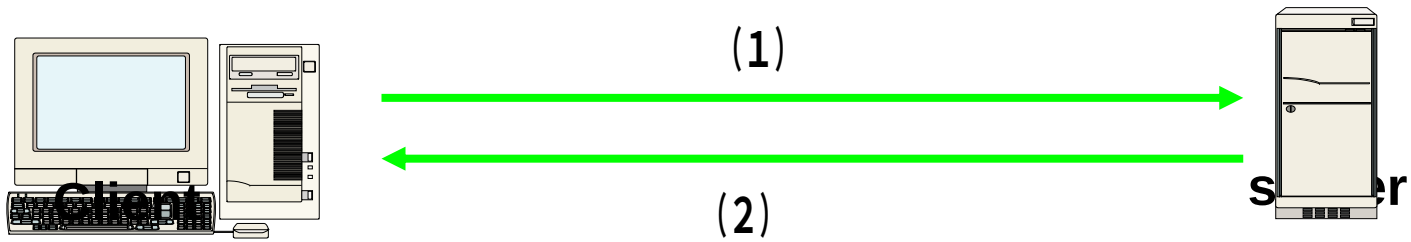
(2) 送られてきた、credential 及びデータフィールドを”credential store” に格納する

Put命令には任意で”credentialーID”を添えることができる

↓

“download” “remove”命令の安全性の向上につながる

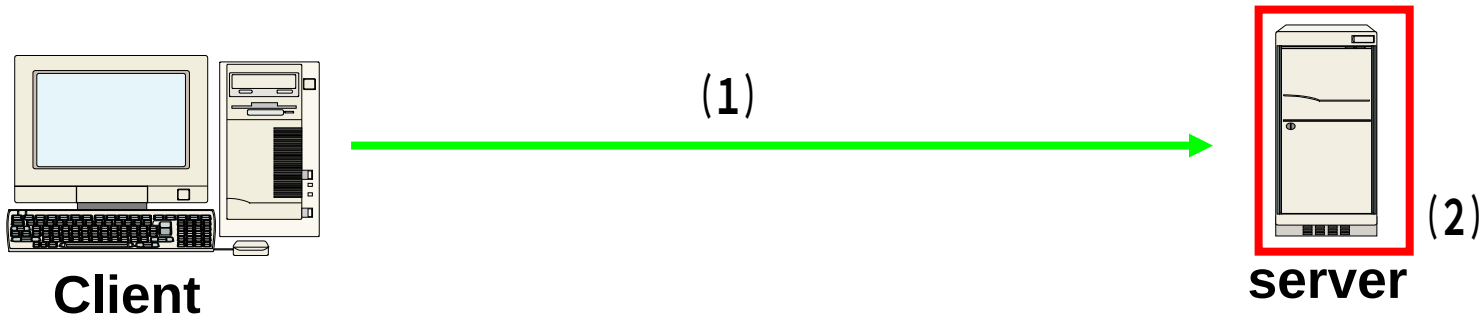
3.3 Credential Download



(1) Credential をダウンロードするための“Get”メッセージを送信する

(2) Credential & ID を含んだ情報を返す

3.4 Credential Removal



(1) **ユーザ からサーバへ、消去したい**Credential **の** credential name **を 含んだ** Delete **メッセージを送信する**

(2) **メッセージを受け取ったサーバは、送られてきた**”credential name” **の** “credential”**を削除する**

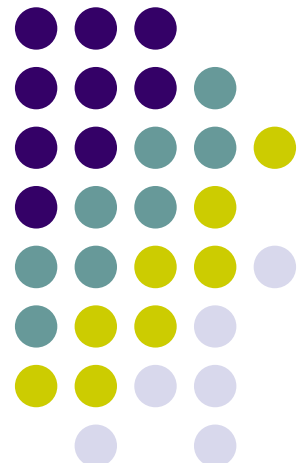
この方式では、安全性に対して保障がされていない

対策

ユーザは任意に、Delete **メッセージに、**credential－ID **を添えることができる。**

↳ Delete**時に、**credential－ID **を比較することによって メッセージの安全性が確保される**

4. Protocol Considerations



4.1 Secure Credential Formats

Credential **の作成、**uploaded **など、これまで解説してきた処理を信頼性を上げるために、明確な**credential－formats **を作成する必要がある**

現在、[PKCS12] **と** [PKCS15] **という2つの**
credential－format **が策定されている**

RSADSI**社が定める、公開鍵暗号技術をベースとした各種の規格群**

PKCS #12：Personal Information Exchange Syntax Standard
保存や送信のための個人情報(秘密鍵、公開鍵証明書等)の　フォーマット。

4.2 Authentication Methods

Authentication：

公認されたエンドユーザに対してのみ credential **の取引する行為を保障するための、きわめて重要なもの**

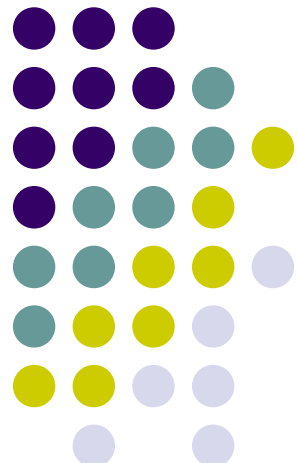
＜現行の”Authentication Methods”の概要＞

- **保障対象**
  - **サーバのクライアントに対する証明**
  - **クライアントのサーバに対する証明**
  - C/S**間の強力な暗号（例：**session**キー）による安全な交渉**
  - Session**キーによる、複数の**user credential **の交換の守護**

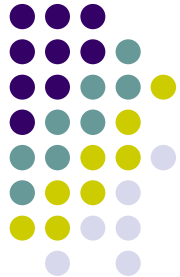
“Authentication Protocol”

- Strong Password Protocols
- TLS

5.Security Consideration



5.1 Communications Security



1. 秘密性

クライアントから credential サーバに対して通信が行われるとき、パスワード及びパスワード実行者を守ることによって、秘密性を守る

2. 完全性

クライアント命令を作り変えたり、古いcredential を使用することよってクライアントに成りすますのを防ぐために、C/S間で完全な通信を行わなければならない

3. 認証時の留意点

サーバは、attackerに対して誤ってcredential を公開しないように、クライアントに対して適切な認証を行う

クライアントは、attackerにパスワードが発覚するのを防ぐために、適切なサーバの身元照会を保障する

5.2 Systems Security

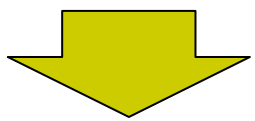
System Security は、サーバやクライアントの保護の度合いや、サーバに蓄積された情報に関係している

1. Client Security

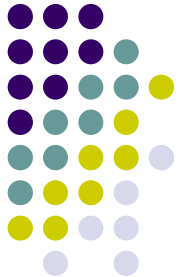
大部分のプロトコルでは、安全なクライアントを実現するためには、ユーザの行動にかなり依存することになる

例：パスワードを使用するプロトコルの場合

ユーザが、パスワードの重要性を知らずに、attacker の目前で堂々とパスワードを打ち込んでしまうことにより、パスワードが流出する



ユーザに対して、クライアント操作時の留意点を学ばせるなど、ユーザのセキュリティに対する認識レベルの向上を図る



2. Server Security

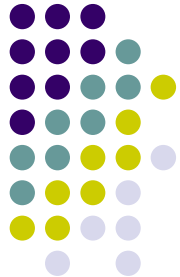
サーバでは、パスワード照会やユーザ認証は、ハイレベル・プロテクションで余裕を持たせる必要がある

(1) オフラインでのアクセスの場合

- 辞書攻撃を避けるために、サーバ認証キーには数段階の処理を必要とするプロテクションを採用する
- フィジカル部分でアクセスを制限する

(2) オンラインでのアクセスの場合

- 誤り回数と入力頻度を考慮して、一定レベルを超えるとそのcredential のアクセス権限を削除する



お わ り