北海道大学における統合認証システムと電子証明書発行サービスについて

尾形 かおり [1)]

1) 北海道大学 総務企画部 情報企画課

security@iic.hokudai.ac.jp

System of Integrated Authentication and Service of Digital Certificate in Hokkaido Univ.

Kaori Ogata [1)]

1) General Affairs and Planning Department, Hokkaido Univ.

概要

北海道大学の教職員、学生、その他の学内関係者、およびネットワークを一時的に利用する学外者の ID を統合する認証方法について、過去の実績と共に現在の状況を報告する。また、認証を安全に行うための暗号化通信に必要な電子証明書についても紹介する。

1 はじめに

近年、情報セキュリティの観点において、ネットワーク上のシステムは正当なユーザのみがアクセスし、安全に運用されることが益々求められている。北海道大学（以下：本学）においても、そのようなシステムに対して認証技術を用いているが、以前は各々のシステムが個別に認証を実施しており、ユーザはシステム毎に ID やパスワードを管理する必要があった。現在では、1 組の ID とパスワードで各システムの認証を行うシングルサインオンシステムや、教職員・学生・その他の学内関係者・一時的に利用する学外者それぞれの ID を統合する認証システムを構築している。本報告では、本学での統合認証を実用化する経緯および現在の統合認証システムの概要について述べる。

加えて、認証をセキュアな環境で利用するための暗号化通信に必要な電子証明書の概要、およびサーバ証明書の発行サービスにおける現況を報告する。

2 認証環境の変遷

以前、本学当課内の IT 推進グループでは学内ネットワークに接続したりメールアドレスを取得したりする ID として PID を発行し、大型計算機システムを使用するためには別の ID（大計番号）を発行していた。また、旅費システム、給与支給明細オンライン照会、電子届出システム等の全学事務系システム（教職員向け）はそのシステムの管理部署が各々ID とパスワードを管理していた。そして、当時の情報基盤課メディア教育担当（現：オープンエデュケーションセンター）では学部学生の ICT 学習環境の整備に ELMS-ID を運用していた。それぞれの部署が各々ID とパスワードを作成して管理していたため、利用者は複数の ID とパスワードを管理する必要があった。

2007 年、当時の情報企画課（現：当課情報環境推進本部担当）がシングルサインオンシステムを構築した際に、全学事務系システムの認証をまとめて SSO-ID を使用するようになった。

同時期に、IT 推進グループは学内ネットワーク登録および大型計算機システム利用のために、情報基盤センターポータルサイトを構築し、iiC-ID を作成した。iiC-ID は SSO-ID や ELMS-ID と同一のものもあったが、パスワード認証の連携をしていなかったため、ID が同一にもかかわらず、ユーザは依然としてパスワードを使い分けていた。

2010 年ごろから、共用 PC の Web 認証サービスや無線 LAN 認証を開始する際、既存の SSO-ID・ELMS-ID・iiC-ID のパスワードを横断的に参照できる統合認証システムを構築した。その後、他の

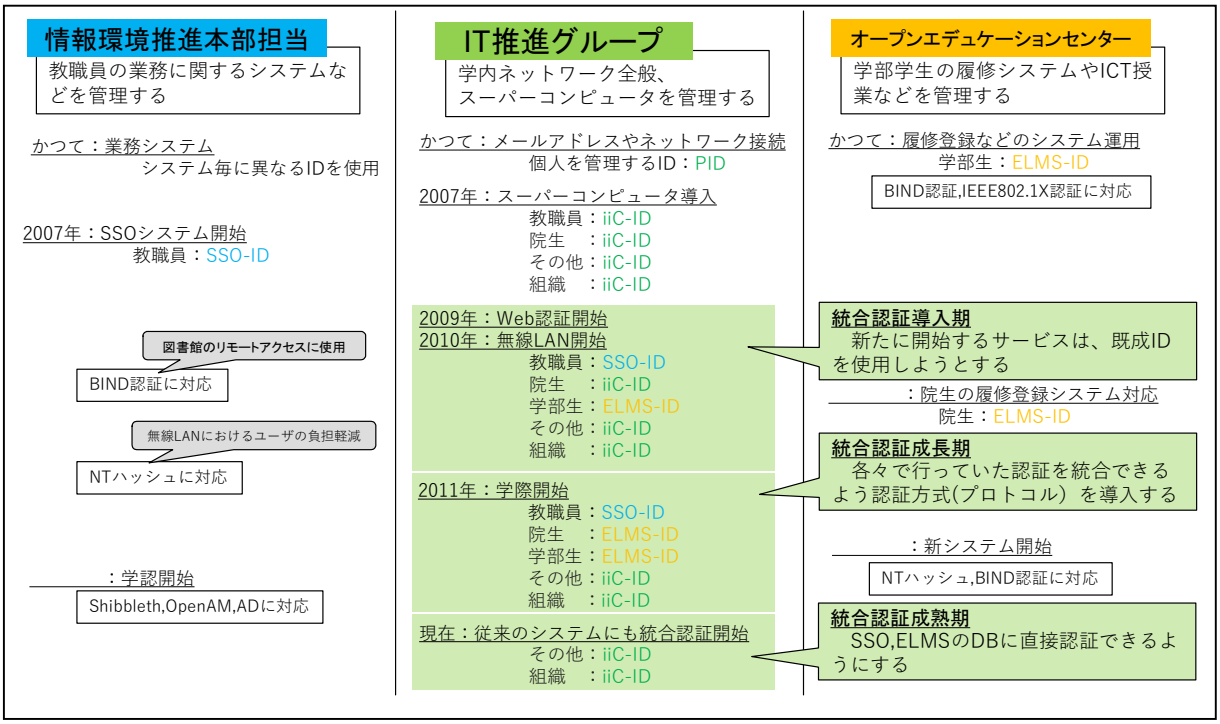

図 1　ID 種別における認証の変遷

システムもこの統合認証を実施するようになり、SSO-ID・ELMS-ID と重複する iiC-ID は運用を停止した。これにより、ユーザは複数の ID を管理する必要性がなくなり、1 つの ID で様々なサービスを利用できるようになった。

ID 種別における認証の変遷を図 1 に示す。

3　現在の認証環境

現在、教職員には SSO-ID を、学部学生および院生には ELMS-ID を、SSO-ID や ELMS-ID を所持しない学内関係者には iiC-ID を発行しており、ID 種別毎に別のデータベースで管理している。その他に、学会などで一時的に本学のネットワークに接続する学外者にはゲスト ID を発行し、これも独立したデータベースで管理している。使用しているデータベースは表 1 に示すように、それぞれ異なる。

対象者	ID 種別	DB
教職員	SSO-ID	ADAM
学生	ELMS-ID	OpenLDAP
その他の学内関係者	iiC-ID	OpenLDAP
一時利用の学外者	ゲスト ID	PostgreSQL

表 1.　ID 種別と使用する DB 一覧

本学での統合認証システムでは、ユーザからの認証要求に応じて、認証用サーバから種別に応じたデータベースに ID とパスワードを横断的に問い合わせ、認証結果を各システムに返す。それぞれのシステム、サーバの認証経路の概略を図 2 に示す。

全学事務系システムから SSO システムのサーバには主にリバースプロキシによるシングルサインオンを実現している。比較的近年に導入した学術認証フェデレーションや学生のメールシステムには SAML を用いている。また、無線 LAN コントローラで行っている IEEE 802.1X 認証の EAP も RADIUS プロトコルによって実装している。他、情報基盤センターポータルなどでは独自の認証形式を使用し、Web 認証システムにも RADIUS プロトコルを用いている。

そして、統合認証サーバから、ID 種別ごとのデータベースへは LDAP の BIND 認証を用いている。

認証連携サーバでは、Shibboleth, OpenAM 等を相互に連携させているが、その他の統合認証サーバは主に RADIUS プロトコルを実装するオープンソース FreeRADIUS を用いて、認証時に目的の種別の ID が格納されているデータベースを参照している。FreeRADIUS は、PostgreSQL 等の RDBMS や LDAP との通信も行えるサーバプログラムであり、本学では RADIUS による認証に対応したソフトウェアを比較的多く利用していたため、本学のシステムの統合認証に採用した。本学の統合認証における FreeRADIUS の設定について次に述べる。

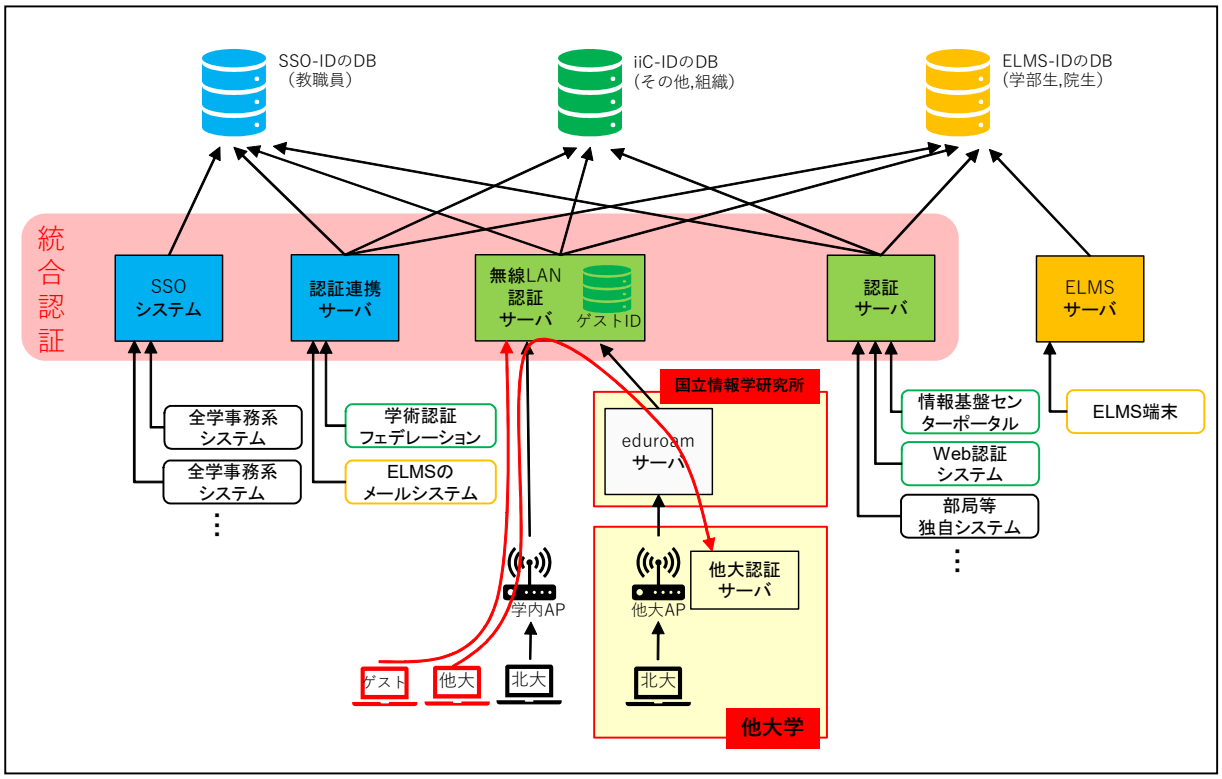

図 2　学内の統合認証の概略

4　FreeRADIUS の設定

FreeRADIUS は、/etc/raddb/ディレクトリにある設定ファイルを編集することによって統合認証を実現する。FreeRADIUS の主たる設定として、ID 種別や参照データベースの定義や RADIUS クライアント（図 2 に示す各種システム）の提示、ID 種別によるデータベースを振り分け設定を行う。

4.1 ID 種別と参照先データベースの定義

ID 種別の定義の設定例を以下に示す。

```
DEFAULT Prefix == "xxx", Auth-Type := XXX
DEFAULT Prefix == "yyy", Auth-Type := YYY
DEFAULT Prefix == "zzz", Auth-Type := ZZZ
```

ID 種別の形式に従い、利用する認可・認証モジュール（データベース）を振り分ける。なお、Auth-Type の値は、authenticate セクションで使用する。例では 3 通りの場合分けであるが、Prefix 毎に更にデータベースを振り分けることも、Suffix の指定も可能である。また、ID 毎の正規表現によりデータベースを振り分けることも可能である。

4.2 LDAP の設定

実際に利用する LDAP やデータベースの設定をする。主たるデータベースとして利用される LDAP の設定例は下記の通りである。表中の”xxx”は LDAP モジュール名であり、モジュールを複数設定して利用する際に用いる。FreeRADIUS 内で固定して使用するモジュール種別と管理者が指定するモジュール名を 1 行で記述する。なお、下記は Prefix == "xxx"に対応する ldap モジュールの設定例であり、他の ldap モジュールも同様に設定する。

```
ldap xxx {
    server = "xxx.hokudai.ac.jp" # LDAP サーバ
    identity = "cn=root,dc=xxx,dc=hokudai,dc=ac,dc=jp"
                                        # バインド DN
    password = rootpass
                                # バインド DN のパスワード
    basedn = "dc=xxx,dc=hokudai,dc=ac,dc=jp"
                                        # サーチベース
    filter = "(cn=%{Stripped-User-Name})"
                                # ユーザ ID とする属性
    start_tls = no # TLS オプション
    dictionary_mapping = ${raddbdir}/ldap.attrmap
                                # 属性名マッピング指定
    ldap_connections_number = 5
    timeout = 4
    timelimit = 3
    net_timeout = 1
}
```

filter 設定の cn ではユーザ名を示す属性名を指定する。また、%{Stripped-User-Name}は、RADIUS クライアントから送信されたユーザ名から Prefix 部分を削除したものを LDAP モジュールに引き渡すことを示す。Prefix も含めて引き渡すに

は%{User-Name}にする。なお、「LDAP モジュール名」は authenticate セクションで使用する。

4.3 認証モジュールの指定

使用する認証モジュールの指定を authorize と authenticate に記述する。

```
authorize {
    files # 認証の振り分けを実施
}
authenticate {
    Auth-Type XXX { # "xxx"の認証に ldap を使用
      xxx
    }
    Auth-Type YYY {
      yyy
    }
    Auth-Type ZZZ {
      zzz
    }
}
```

これで FreeRADIUS に送信されてきたユーザ ID の Prefix 毎に認証が振り分けられる。なお、ID 種別と参照先データベースの定義をする際に、すべてのデータベースを順に参照することも可能であるが、本学ではユーザ数が多く、バックエンドの認証サーバに余計な負荷がかかるため、参照するデータベースを予め ID の形式から振り分けるよう実装した。

5 電子証明書の有用性

セキュリティで保護された環境でユーザが認証を行うには、システムの Web サイトへは暗号化通信（SSL 通信）が必要である。ここでの暗号化通信には電子証明書の一つであるサーバ証明書を用いる。サーバ証明書は、個人情報などが盗聴・改竄されないようにクライアントとサーバ間（あるいはサーバ同士）での暗号化通信に使用する。また、サーバ証明書は、サーバを運営する組織が実在し、そのサーバが正当なドメインを適正に使用していることを第三者機関（認証局）が電子的に証明する。

Web サイトの閲覧者が、証明書を適用した Web サイト（サーバ）にアクセスを試みると、閲覧者の Web ブラウザ（クライアント）は Web サーバより証明書を取得する。取得した証明書と、認証局が公開しているルート証明書の 2 つの証明書によって、Web ブラウザは Web サーバの公開鍵が信頼のおけるものであると認識し、暗号化通信を確立する。

6 サーバ証明書発行サービス

北海道大学のドメイン(hokudai.ac.jp)を有するサーバに対して、当チームでは国立情報学研究所(NII)の UPKI サーバ証明書[1]を発行している。サーバ証明書発行サービスにおける業務内容を以下に述べる。

Web サーバの管理者よりサーバ証明書の申請を受けたら、サーバの管理状況を確認し、利用条件を満たす場合は証明書作成を許可する。次に、暗号化通信を行うために必要なファイルをサーバの管理者から受け取り認証局に提出する。その後、認証局からサーバ管理者に証明書が発行され、サーバ管理者が証明書をサーバに設定すると、サーバは暗号化通信が行える(図 3 参照)。

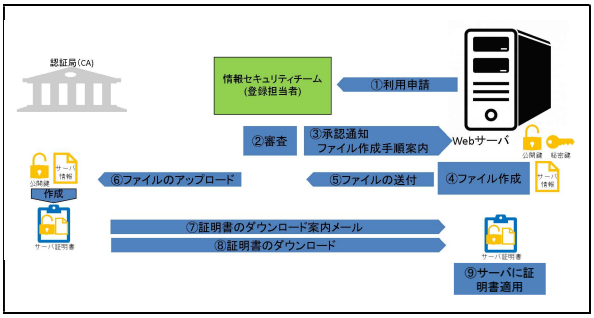

図 3 サーバ証明書発行の手順

7 今後の展開

7.1 統合認証技術

対応する認証プロトコルがシステムによって異なるため、プロトコルの特性を生かした認証方式を今後も検討していく。また、新規に導入するシステムや更新を行うシステムには、対応する認証プロトコルを都度確認し、SAML を主とする認証方式への統一を目指す。

7.2 電子証明書発行

現在、Web サーバへの暗号化を伴わない通信に対して『安全な通信ではない』と警告を促す Web ブラウザがあり、このような非暗号化通信を警告する Web ブラウザは今後増えると予想されている。認証を行わない Web サーバも暗号化通信を行うようサーバ管理者にサーバ証明書の取得を促すと共に、クライアント証明書およびコード証明書の発行も検討する。

参考

[1] https://certs.nii.ac.jp/