セッション P09

IBM eServer iSeries

iHOPE
iSeries Hands-On Professional Education

EIM概説

特記事項

当資料で解説される項目の更に詳細な説明は、製品から提供されるマニュアル、オンライン・ヘルプ、Web上の情報を参照してください。

当資料は、2003年4月現在のIBMその他の製品情報に基づいて作成されております。この資料に含まれる情報は可能な限り正確を期しておりますが、日本アイ・ビー・エム株式会社による正式なレビューは受けておらず、当資料に記載された内容に関して日本アイ・ビー・エム株式会社および日本アイ・ビー・エム　システムズ・エンジニアリング株式会社が何ら保証をするものではありません。したがって、この情報の利用またはこれらの技法の実施はひとえに使用者の責任においてなされるものであり、当資料の内容によって受けたいかなる被害に関しても一切の保証をするものではありませんのでご了承ください。

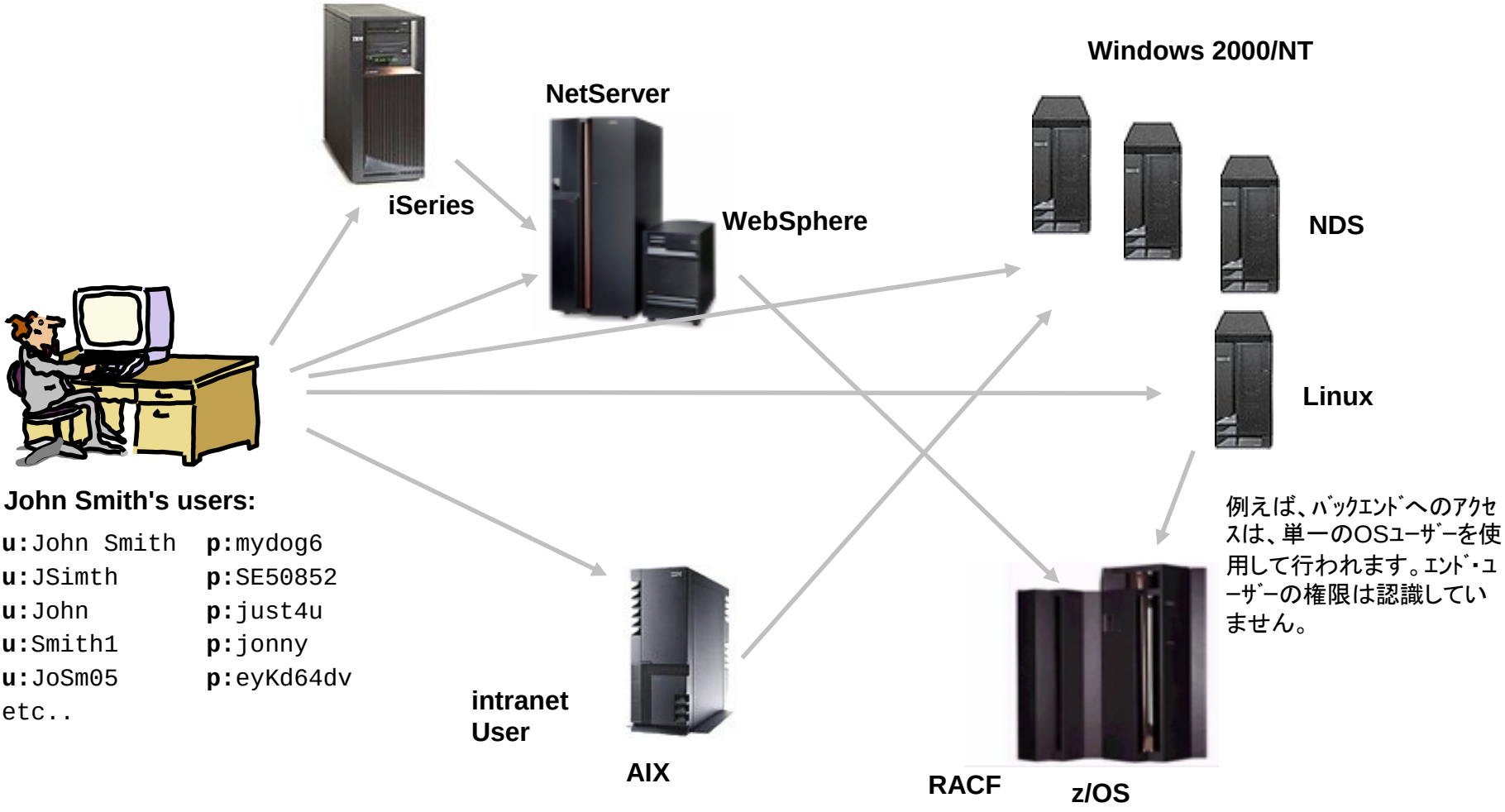
オリジナル・コンテンツ（英語版）作成：Erik Larsson (IBM Sweden)
オリジナル・コンテンツ（英語版）監修：Thomas Barlen (IBM Germany)

目的

- Enterprise Identity Mapping (EIM)の紹介
  - EIMとは
    - EIMが解決する問題とその必要性
  - 関連コンポーネントの理解
    - LDAPディレクトリー, Kerberos, API, etc.
- Kerberosの紹介
  - Kerberosの説明, その目的と稼動の仕組み

Enterprise Identity Mapping (EIM)

現在の標準的環境



:

Notes:現在の標準的環境

システムはそれぞれ、独自のユニークなユーザー・レジストリーを持っており、ユーザーIDやパスワードも独自のルールを持っています。ユーザーは結局、複数のユーザーIDとパスワードを持つことになります。ユーザーが、複数システムで同じパスワードを使うことによって、システム環境を簡単にしようとすることは、ごく一般的なことです。

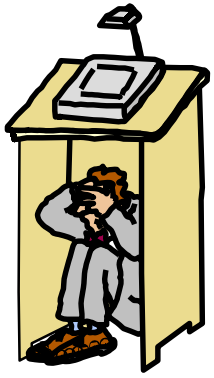
アプリケーション開発者は、お客様のデータが多くの異なるタイプのシステムに分散されていることを知っています。これらの全てが、独自のユーザー・レジストリーと関連するセキュリティー論を持っています。分散アプリケーションを提供することは、その管理の影響に関わらずそのアプリケーションのためのユーザー・レジストリーを提供することでもあります。

クロス・プラットフォームに分散されたアプリケーションは、特定のユーザーが誰であるかに応じます。OSの保護された資源がアクセスされる時、アプリケーションは、アプリケーション・ビューのユーザーをOSビューのユーザーのマップします。バックエンド・システムは、フロントエンド・システムを信用しなければなりません。

パスワードは、時には、クリアー・テキストで送信されます。

現在の問題

- 全てのサーバー・プラットフォームは、ユニークなユーザー管理メカニズム（ユーザー・レジストリー）を持っており、管理を複雑化している。
- 全てのシステムのユーザーをトラックし続けることが困難である。
- ユーザーは、各々のシステムのユーザーIDとパスワードを憶えておかなければならない。
- アプリケーション開発者は、独自のユーザー・レジストリーを作成し、バックエンド・システムへアクセスするために安全でない技術を使う
- Tivoliのような単一点での管理ツールは、管理者の問題を解決するが、ユーザーやISVの問題を必ずしも解決しない。



Notes:現在の問題

現在の、パーティション化されたサーバーや複数のプラットフォームを持つ異種ネットワークにおいて、管理者，ユーザー，アプリケーション開発者の全ては、企業ネットワーク内で個々のユーザーのために複数のユーザー識別が作成されることに対処しなければなりません。ユーザは、使用するシステムで、それぞれのユーザーID，パスワードを憶えていなければなりません。管理者は、パスワードのリセットや、ユーザーIDとパスワードの同期を行わなければなりません。また、各ユーザーがアクセスするネットワーク内の全てのシステムを憶えておかなければなりません。アプリケーション開発者は、この問題を解決するためにセキュアーでないテクニックを無理に使うか、セキュリティー論に関連した独自のユーザー・レジストリを実装したアプリケーションを書くために巨額を投資しなければなりません。これらの問題は、すぐに関連者全てを巻き込む管理上の大きな問題となります。

シングル・サインオン環境を扱うための1つのアプローチとして、全てのユーザー・パスワードとユーザーIDを含むサイド・ファイルを作成する方法があります。パスワードは、引き続きシステム上で管理される必要があり、レジストリーでは、ユーザー/パスワードが同期されている必要があります。パスワードは、通常クリアー・テキストで送信されます。そして、管理者が直接アクセス可能なクリアー・テキストまたは、解読可能なファイルとして保存されます。最終的には、このアプローチは、複数階層の異種アプリケーションを提供しようとするサード・パーティー・アプリケーション・プロバイダーには何もしません。

複数のレジストリー環境を管理することは、予算上重荷になります。ヘルプ・デスクへのコールの20-40%が、パスワードを忘れたことによるもので、その費用は、パスワード・リセット毎に14～26$掛かっています。

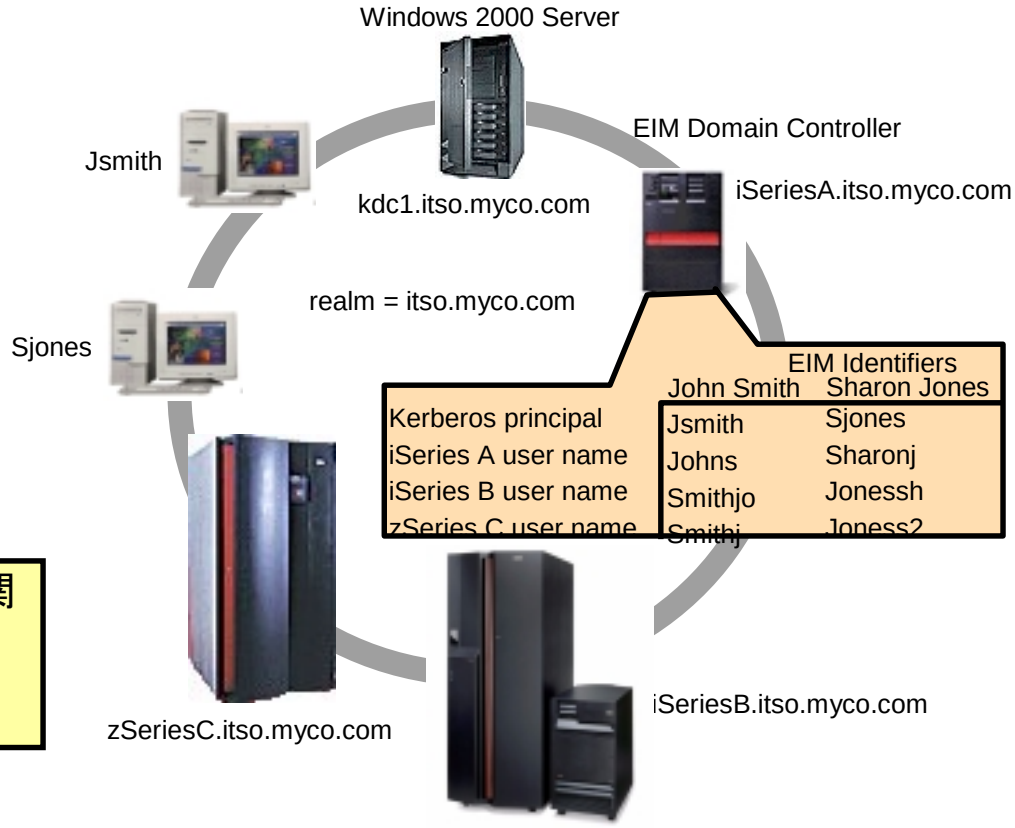
- 出展: Gartner Group.

EIMとは?

- Enterprise Identity Mapping (EIM)は、企業内の様々なレジストリー内で、人またはエンティティーを適切なユーザー識別(ID)にマッピングするためのメカニズムです。
- EIMは、アプリケーション開発者が、シングル・サインオン・ソリューションを提供するための費用を低く抑える基盤を提供します。

(Autonomic Computing)

EIMの定義: OSﾌﾟﾗｯﾄﾌｫｰﾑ, ｱﾌﾟﾘｹｰｼｮﾝやﾐﾄﾞﾙｳｪｱに関連するﾕｰｻﾞｰ・ﾚｼﾞｽﾄﾘｰを跨るIDの関連性です。



Notes:EIMとは?

Enterprise Identity Mapping (EIM)は、アプリケーション開発者が、シングル・サインオン・ソリューションを提供するための費用を低く抑える基盤を提供します。他のIBM　～　プラットフォームやIBMソフトウェアに沿って、EIMおよびKerberosのOS/400での開発は、シングル・サインオンの機能を提供するものです。これは、ユーザー，管理者，アプリケーション開発者に、基礎となるセキュリティー・スキーマを変更することなく、複数プラットフォームに跨る簡単なパスワード/ユーザーID管理のメリットをもたらします。

EIMは、OSプログラマーやISVが、特定のプロダクト・ベンダーからのサポート待つことなく、独立してシングル・サインオン環境を実装することを可能とします。

EIMは、ビジネスにおいて、今日より何百倍も複雑であるシステムやテクノロジー基盤を管理する能力を与えることを目標とするオートノミック・コンピューティングの一部です。

オートノミック・コンピューティングにおける自己管理サーバーは、お客様にとって究極のツールです。これらのツールには、自己最適化，自己構成，自己修復，自己防御があります。

EIMが提供するもの

- シングル・サインオン機能
- 管理の容易さ
  - 既存データのために正しい場所に存在するセキュリティー論を信頼
  - パスワードの紛失のための管理負荷を削減
  - クライアント・サイドのリスクを削減
    （キャッシュされたパスワード，付箋に記載されたパスワード，etc）
- よりよいアプリケーション・デザイン
  - 新しいユーザー・レジストリーを実装する必要なし
  - 追加のセキュリティー・メカニズムを定義、施行する必要なし
  - 分散された複数階層アプリケーションの開発者に最大限の柔軟性を提供
- ユーザーのプロセスを平易にする　アクセスは、見えないところで制御されます。
- iSeriesは、EIM機能を提供する最初のIBM　～　プラットフォーム
  zSeries, pSeries, xSeries, Java, Linux等
  の他のプラットフォームが続くことになります。

Notes:EIMが提供するもの

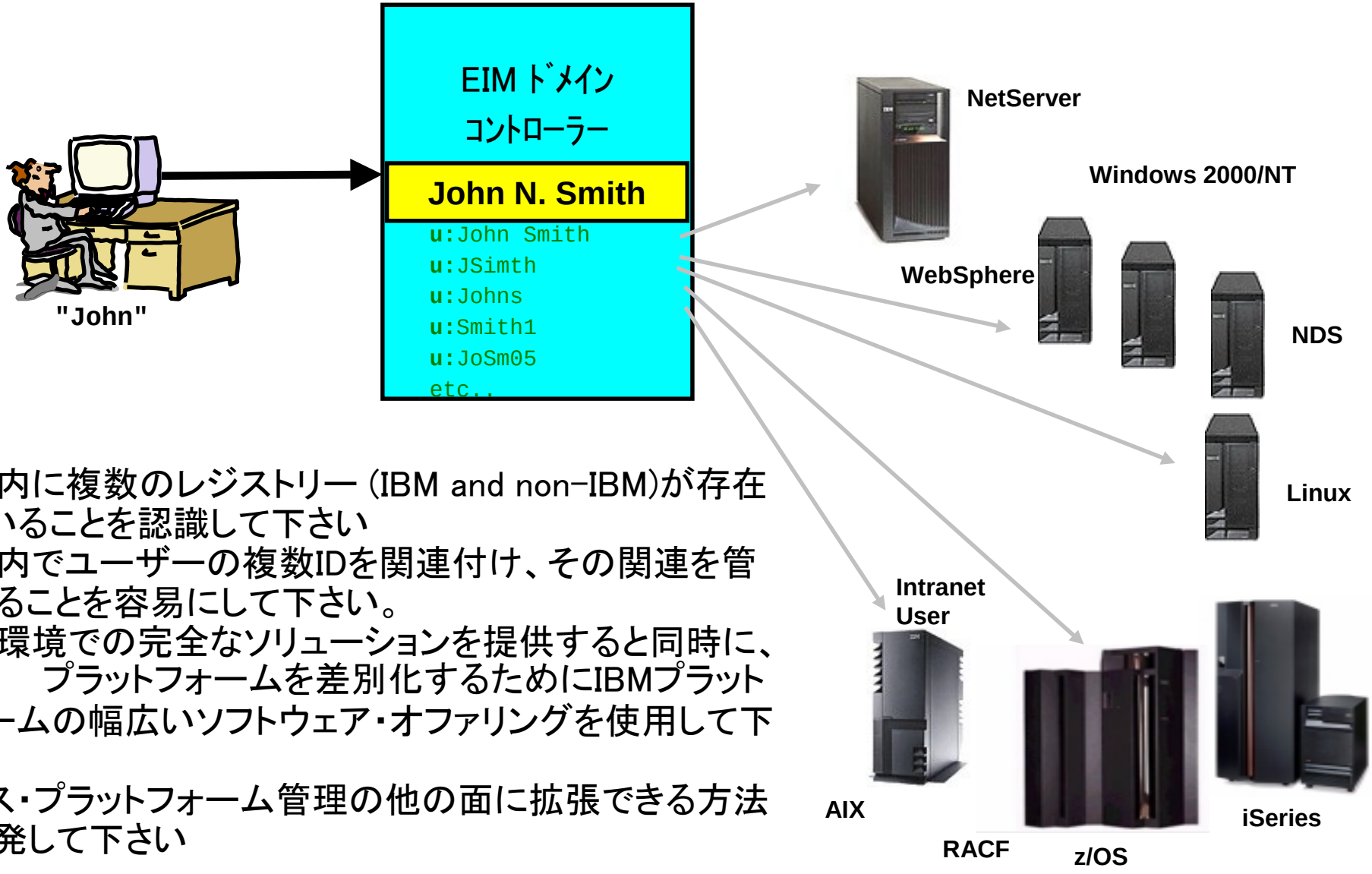
Enterprise Identity Mappingは、クロス・プラットフォームでのシングル・サインオン機能のメカニズムを提供します。企業内でシングル・サインオンが使用される時に、ユーザー，管理者，アプリケーション開発者に様々なメリットをもたらします。

iSeriesサーバーは、OS/400インターフェースがNetwork Authentication Service（例えば、Kerberos）によってユーザーを認証できるようEIMを使用します。アプリケーションはOS/400同様に、Kerberosチケットを受け入れることが可能であり、Kerberosチケットが表しているユーザーのユーザー・プロファイルを見つけるためにEIMを使用します。

他の　^　プラットフォームは、現在EIM実装に取り組んでおり、ほとんど基盤を完備しています。
アプリケーションは、OS自体に頼ることなく、EIM APIからメリットを享受することが可能です。

EIMは、共通サービスを持つプラットフォームやレジストリーの境界を越える時、IDを変換する必要があるアプリケーションやプラットフォームのランタイムの要求を扱います。

提案されるアプローチ



- 企業内に複数のレジストリー (IBM and non-IBM)が存在していることを認識して下さい
- 企業内でユーザーの複数IDを関連付け、その関連を管理することを容易にして下さい。
- 異種環境での完全なソリューションを提供すると同時に、
  - ~ プラットフォームを差別化するためにIBMプラットフォームの幅広いソフトウェア・オファリングを使用して下さい
- クロス・プラットフォーム管理の他の面に拡張できる方法で開発して下さい

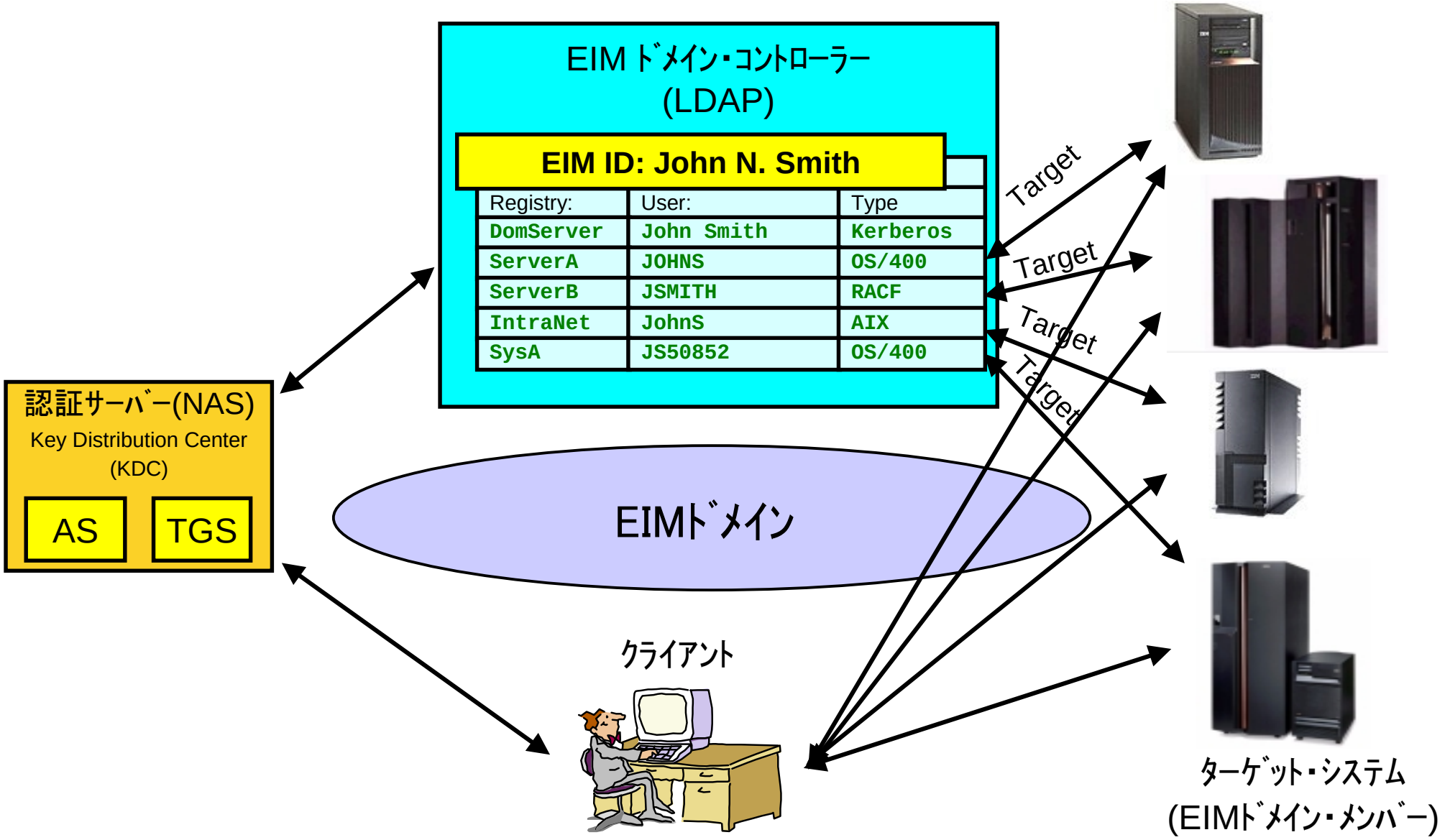
Notes:提案されるアプローチ

新しいユーザー・レジストリーを作り出す、または複数ユーザー・レジストリーおよび関連するセキュリティー論が存在し、今後も存在し続ける事実を無視しようとするよりむしろ、既存のプラットフォームを跨るユーザーIDを同等に扱うためにEIMを使用して下さい。

ユーザーが1つのユーザー・レジストリーで既に認証されている場合、他のユーザー・レジストリー内に存在するどのIDが同じ人を表すのかを決定することができます。

EIMは、企業内における個人やエンティティーとユーザー・レジストリー内の関連するID間の関係を管理するためにデザインされています。

EIMのコンポーネント



Notes:EIMのコンポーネント

■EIMドメイン
EIM データを含む Lightweight Directory Access Protocol (LDAP) サーバー内のディレクトリーです。

■EIMドメイン・コントローラー(LDAP)
少なくとも 1 つの EIM ドメインを管理するために構成された Lightweight Directory Access Protocol (LDAP) サーバーのことです。

■EIM ID
EIMドメイン内の個人やエンティティーを表します。EIMドメイン内でユニークでなければなりません。

■認証サーバー(NAS –KDC)
ユーザー認証用のユーザー名およびパスワードの代わりに Kerberos チケットを発行します。これにより。ネットワーク内の別のサービスに対して自分の ID を証明することができます。

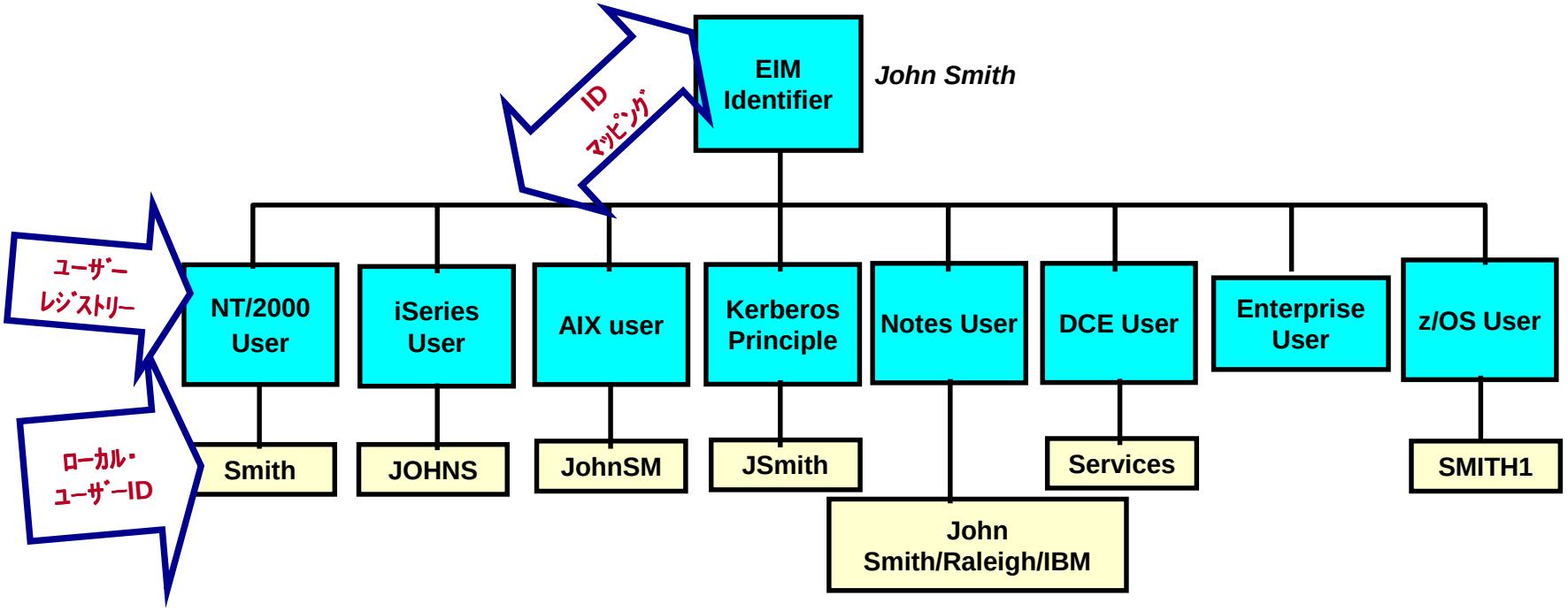
■ターゲット・サーバー(EIMドメイン・メンバー)
EIMドメインに参加し、シングル・サインオンを許可するサーバー(サービス)です。

■クライアント
EIMドメインに参加し、シングル・サインオンを要求します。

EIMコンポーネント: EIM ID

EIM IDは、EIMにおける実際の人やエンティティーを表します。

IDの関連性（マッピング）は、マッピングにアクセスするためのプラットフォームを跨る共通サービスと共に、LDAPのような周知のロケーションに保管されます。



Notes:EIM ID

EIM IDは、EIMにおける実際の人やエンティティーを表します。人やエンティティーのためのユーザーIDは、EIM IDと関連付けられます。これらのIDのマッピングは、人やエンティティーが企業内で持っている全てのユーザーIDをトラックし続ける管理タスクを簡単にします。

EIM IDは記述を持っており、その記述はIDが表す人やエンティティーを詳細に定義します。また、EIM IDの別名を作成することもできます。これは、マッピングを見つけるための操作が行われる時、特定のEIM IDの所在場所を定めることを援助するものです。

企業内で異なる人が、同じ名前をシェアすることは、稀ではありません。EIM ID名は、IDがどの個人に属するかを混乱させるので、EIMドメイン内でユニークでなければなりません。別名は、EIM管理者が任意でユニークなEIM ID名を持つことを可能にし、EIM IDが属する個人に関する追加の情報を提供することも可能にします。

例えば、名前がJohn S. Smithである2人のEIM IDは、John S. Smith1 と John S. Smith2 かもしれません。John S. Smith1の別名は、 John Samuel Smith とすることができ、 John S. Smith2 の別名は、 John Steven Smith とすることができます。

各々のEIM IDは、どちらの John S. Smith を表しているかを見分けるために使用できる、複数の別名を持つことができます。もう1つの別名は、所属する組織番号を含む2人の個人のためのEIM IDそれぞれに追加されるかもしれません。

EIMコンポーネント: LDAP

- EIMは、EIMドメイン・データと一緒にIDを保管するためにディレクトリー（LDAPサーバー）を使用します。
- ディレクトリー・サーバーは、EIMドメイン構成へのアクセス制御も行います。
- EIMドメインを作成するために基本のディレクトリー構成が必要となります。
- ユーザーは、LDAPディレクトリー・ツリー内のEIMドメイン・データを直接処理するべきではありません。
  - EIMドメインを管理するために、EIM APIが提供されます。

Notes:LDAP

Enterprise Identity Mapping (EIM)は、少なくとも基本構成されているディレクトリー・サービス(LDAP)サーバーを必要とします。
1つも存在しない場合、EIMウィザードが構成します。EIM管理の観点でディレクトリーに直接アクセスする必要はありません。

しかし、社員情報の保管のような他の機能のため、もしくはレプリケーションやSSLのような高機能を構成するためにディレクトリーを使用する計画があるなら、まずはLDAPディレクトリー・サーバーを熟知すべきです。LDAPを構成しようとする前に、iSeries Information Center内、計画情報の"Plan your LDAP directory server"を参照して下さい。ディレクトリー・サービスを熟知しており、LDAPの計画段階を終えているなら、構成プロセスを開始するために"Install and configure Directory Services"(Information Center内)を参照して下さい。

もう1つのiSeriesディレクトリー・サービスの実装と使用の優れた情報源は、IBM Redbook *Implementation and Practical Use of LDAP on the IBM ~ iSeries Server*, SG24-6193 です。

ディレクトリー・サービスは、EIMドメインおよびEIMに含まれる情報へのアクセス制御と同様にドメイン・コントローラーの情報，権限のためのコンテナです。

本番環境では、ディレクトリー・サービスをSSLを使って構成することを勧めます。

EIM APIを使用せず、EIM情報を変更しないで下さい。

EIMコンポーネント: NAS

- Network Authentication Service (NAS)は、iSeriesが認証のためにユーザーIDとパスワードを使うことに代わりにKerberosチケットを使うことを可能にします。
- アプリケーションは、ユーザーを特定し、安全にIDを他のサービスにパスします。
- NASは、Kerberos Network Authentication Service (RFC1510)に基づいています。
- APIを使うことによって、EIMは、他の目的のために使用されるNAS無しで使用することもできます。

Notes:NAS

iSeriesサーバーを含む多くのプラットフォームが、既に認証のためのKerberos（Network Authentication Serviceとして知られる）をサポートしています。

Network Authentication Serviceは、iSeriesやiSeries Access for Windowsのようないくつかのiシリーズサービスが、ユーザーを認証するためのユーザー名，パスワードに代わって、Kerberosチケットを使うことを可能にします。Kerberosプロトコルは、ユーザーまたはサービスが、危険なネットワーク内で他のサービスに対し、IDを証明することを可能にします。

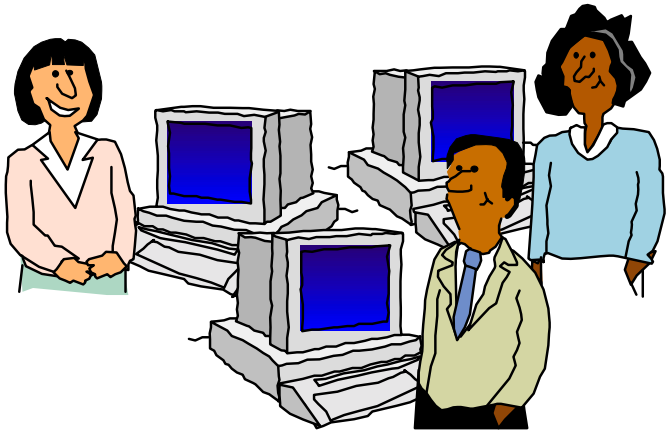
プリンシパルの認証は、key distribution center (KDC)と呼ばれる中心に置かれたサーバーと通して完了されます。KDCはKerberosチケットでユーザーを認証します。これらのチケットは、ネットワーク内の他のサービスに対し、プリンシパルのIDを証明します。これらのチケットによりプリンシパルが認証された後、ターゲット・サービスと暗号化されたデータが交換されます。Network authentication serviceは、ネットワーク内のユーザーまたはサービスのIDを確認します。アプリケーションは、安全にユーザーを認証することができ、ネットワーク上の他のサービスに安全にIDをパスすることができます。一度ユーザーが分かれば、ネットワーク資源を使用するためのユーザー認証を確認するための別の機能が必要とされます。Network authentication serviceは、以下の仕様を実装しています。:

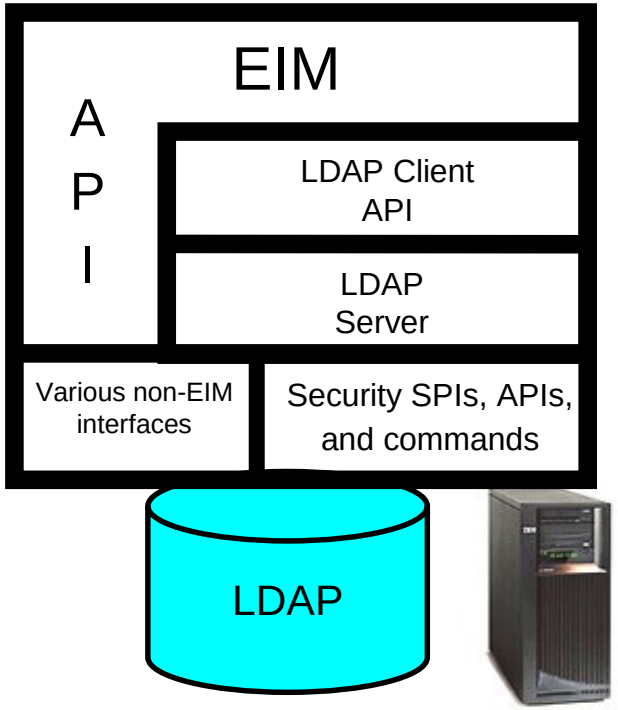
- Kerberosバージョン5 プロトコル Request for Comment (RFC) 1510
- 今日の業界で一般となっている多くの事実上標準のKerberosプロトコル API
- RFC 1509, 1964, 2743として定義されているGeneric Security Service (GSS) API

iSeries上のNetwork authentication serviceは、Microsoft's Windows 2000 Security Service Provider Interface (SSPI) APIのようなこれらのRFCに準拠している認証，委任，データ機密性サービスと相互運用します。

EIM API

- EIMは、ドメイン情報を保管するためのディレクトリー・サーバーにアクセスするAPIコレクションを使用します。
- 新しいプロトコルでなく、LDAPが使用されます。
- EIM APIは、GUIユーザー管理ツールのようなサード・パーティー製品のために使用されます。
- IBMは、ISVが製品にバンドルするためのEIM APIとJavaパッケージを自由に配布します。





Domain Management

- **eimGetHandle()**
- **eimConnect()**
  - authenticate caller (U1) in registry A (REGA)
  - ..
  - eimGetTargetFromSource(U1, REGA, REGB, associated_identity)
  - setuid(associated_identity)
  - perform task as local identity
  - get next request
- **eimDestroyHandle()**

Notes:EIM API

EIMで使用されるAPIのカテゴリー

- EIM “ハンドル” オペレーション – 共通
  - EIMサービスのインスタンスであるトークンを管理　起動者が“ハンドル”への接続を保持する責任を持つ他のサービスに対するコンセプトに類似
- ドメイン・オペレーション – EIM管理
  - EIMドメインの作成とEIMドメイン・コントローラーの確立
- レジストリー・オペレーション – EIM管理
  - システムまたはアプリケーション レジストリーのEIMインスタンスへの参加
- EIM IDオペレーション – EIM管理
  - ユーザーのための“anchor”ポイントの管理
- EIM コア・マッピング・オペレーション　– 実行時
  - 異なるレジストリーを跨りユーザーIDの判別をサポート
- システム・オペレーション – システム/EIM管理
  - EIMドメインへの接続
- ユーザー管理オペレーション – 管理
  - このサービス・セットの定義は、進行中
  - 以下を記述するXMLマークアップを定義する方向:
    - レジストリー内のユーザーとAPIに渡すデータの定義
    - 複数レジストリーを跨るユーザーの追加/変更/削除を可能にする

Notes:EIM API (続き)

EIM 開発プログラミング・モデル –以下のようなEIM APIを利用するアプリケーションを書く場合:

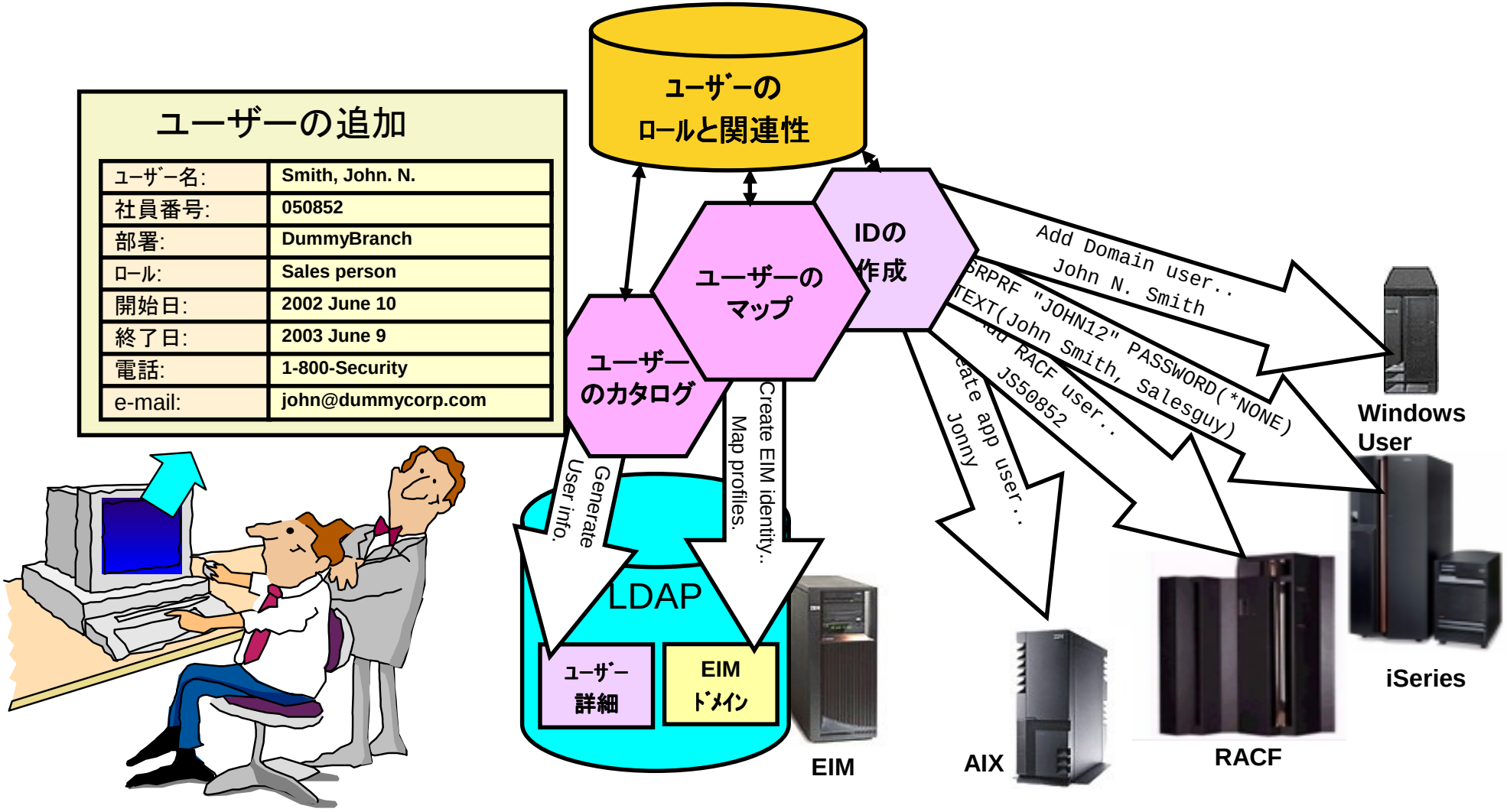
eimGetHandle()
eimConnect()
- レジストリーA (REGA) 内のコーラー (U1) を認証
- ...
- eimGetTargetFromSource(U1, REGA, REGB, 関連ID)
- setuid(関連ID)
- ローカルIDとしてタスクを実行
- 次のリクエストの取得

eimDestroyHandle()

Blank

ユーザー管理ツールを作成するためのAPIの使用

EIM APIの考えられる使用法は、ユーザーを管理する管理ツールです。



Notes:ユーザー管理ツールを作成するためのAPIの使用

この例では、管理者は企業内のユーザーを作成しています。これは、既存環境でのロールに添ったプロファイルとIDの作成から始めます。これらのIDは、各々のレジストリーのルールに添ってユニークなIDが作成されます。
これは、ユーザーIDをEIMドメイン・データベースのIDにマップすることも行います。最後に、このプロセスは会社のディレクトリーに、ユーザーのためのエントリーを作成します。

これらのユーザーIDは、非常に複雑なパスワードと共にセットアップすることができます。また、iSeriesのケースでは、パスワード認証の使用をディセーブルにすることができます。(password を *NONE に設定)

EIMを使うことによって、管理ツールは、ユーザーの除去/ディセーブルまたは会社内でのユーザーのロールの変更、のような機能を持つこともできます。
ユーザーが管理者の役割を持つ場合、管理システムへのアクセスが可能になります。

Note: Java JarファイルおよびLinux EIM APIのオープン・ソース・バージョンは、ビジネス・パートナーが自由に彼らのアプリケーションにAPIをバンドルすることが可能であり、認められています。

アプリケーション提供者のEIM開発

- 開発者はアプリケーションの作成とメンテナンスに多くを投資:
  - ユーザー・レジストリー特定のアプリケーション
  - セキュリティー論に関連
  - IDマッピング特定のペアワイズ・アプリケーション
- EIM:
  - 開発コストの削減
  - 管理コストの削減
- ISV製品オポチュニティー:
  - 企業ビューでのEIM管理のための製品(eServerやeServer以外)
  - 関連性の作成やユーザー・レジストリーの定義プロセスを自動化
  - EIM内のID関連情報を利用するシステムやユーザー管理ツール(例えば、ユーザーに関連したID全ての削除)
  - 複数階層、異種アプリケーションを安価に作成

Notes:

開発者は、アプリケーションの作成、メンテナンスに多くを投資

- ユーザー・レジストリー特定のアプリケーション
- セキュリテー論に関連
- IDマッピング特定のペアワイズ・アプリケーション

これは、アプリケーションのコストやこれらのアプリケーションを展開するIT環境の管理コストを増加させます。

EIMは、開発コストを削減:

EIMは、複数階層、異種アプリケーションの開発コストを顕著に削減します。

- 新しいユーザー・レジストリーの実装が不要
- 追加のセキュリティー論の定義、施行が不要
- 分散、複数階層アプリケーション開発者に最大限の柔軟性を提供

EIMは、管理コストを削減:

EIMは、管理コストを顕著に削減します。 -- セキュリティー管理を容易にします。

- 管理者は、新しいユーザー・レジストリーを管理する必要なし
- 既存データのための既存セキュリティー論を信頼
- 人またはエンティティーについての情報と関連するIDを提供

ISV製品オポチュニティー:

IBMは、以下によりeServerおよびnon-IBMプラットフォームでEIMを広く利用可能にします。:

- 全てのIBM　～　プラットフォーム (xSeries、Linuxを含む) で6ヶ月以内にそれぞれでEIM基盤を提供 (予定)
- Java実装の提供 (予定)

ISVが製品にEIM APIを自由にバンドルすることは、ある程度許可されています。

IBMは、EIMを業界標準の主要部分と考えています。

iSeries上でEIM/NASが利用できるサービス

- V5R2 iSeriesサーバーでは、以下のアプリケーションにシングル・サインオン可能です。:
  - iSeriesナビゲーター
  - ホスト・サーバー
  - iSeries Access for Windows
  - PC5250エミュレーター (Telnetサーバー)
  - DRDA, ODBC, JDBC, DDM
  - NetServer
  - QFileSvr.400
- EIMでは、以下のユーザー・レジストリー・タイプが事前定義されています。:
  - OS/400
  - AIX
  - Kerberos
  - Kerberos (ケース・センシティブ)
  - LDAP
  - RACF
  - Windows 2000
  - Novel Directory Services
  - Policy Director

Notes:EIMを利用できる機能

V5R2では、iSeries Navigator, ホスト・サーバー, ODBC/JDBC/DRDA, PC5250+Telnetサーバー, NetServer, QFileSrv400が、KerberosとEIMによるシングル・サインオンを利用することができます。

これは以下の内容を意味します。:
ユーザーは、Kerberosが利用可能なシステム(Windows2000等)にログインすることができ、一度ログインすれば、二度とユーザーID, パスワードを入力する必要がなくなります。ユーザーがiSeriesナビゲーターのシステムをクリックする時、適切なユーザー・プロファイルの下でシステムに自動的にサインオンされます。ユーザー名またはパスワードの同期はありません。事実、OS/400ユーザー・プロファイルは、管理者が選んだ場合、PASSWORD ＊NONEで構成することができます。

SQLは、iSeriesや他の　～　プラットフォームであってもそれらのマシンからのデータにアクセスするためにiSeriesナビゲーター(または、認証にKerberosを使用するスタンドアローンのODBC or JDBCアプリケーション)経由でサブミットされます。繰り返しますが、この全ては、ユーザーID, パスワードの入力やSQLステートメント内にパスワードをコードすること無しで行われます。それでも、適切なセキュリティーは、適切なユーザーIDとネイティブ・セキュリティー論を使用し、それぞれのシステムで施行されます。この全てが、どんなプラットフォームでもエージェント・コード無しで稼動します。

PC5250は、ユーザーIDとパスワードを使用せず、サインオンのバイパスを許可します。

Kerberosを使用するよう構成されたNetServerを使用し、ユーザーはユーザーIDとパスワードを提供すること無しに、OS/400ファイル・システムをドライブにマップすることができます。繰り返しますが、このユーザーには適切なセキュリティーが施行されています。

QFileSrv400もまたユーザーID, パスワード無しのサインオンが可能です。ユーザーは、iSeriesナビゲーターで単一のiSeriesに接続することができ、実際は2台目のiSeriesサーバーにポイントしているQFileSrv400マウント・ポイントにアクセスすることができます。3つの異なるユーザーID(Windowsログイン, iSeries1のプロファイル, iSeries2のプロファイル)を持つことができます。ユーザーIDおよびパスワードの入力を促すこと無く、マウント・ポイントにアクセスすることができます。(iSeries1のマウント・ポイントへのアクセスが許可されていると仮定して)また、ユーザーIDおよびパスワードの再入力無しで、iSeries2のデータにアクセスすることもできます。(iSeries2のデータへのアクセスが許可されていると仮定)

これらは、KerberosおよびV5R2のEIMを利用するオペレーティング・システム・レベルのインターフェースです。

Notes: EIMを利用できる機能(続き)

ユーザー・レジストリー

ユーザー・レジストリーは、オペレーティング・システムのユーザーIDや既知または信頼されている、または既知で信頼されているアプリケーションのセットを表すエントリーのセットを包含しています。ユーザーIDのセットは、特別のアプリケーションと共に使用されるシステム・ユーザー・レジストリー全部、またはシステム・ユーザー・レジストリーのサブセットとすることができます。特にRACFユーザー・レジストリーにCICSのために定義されたユーザーのリストは、アプリケーション・レジストリーの例となります。ユーザー・レジストリーが、例えば、特定のiSeries上のOS/400ユーザー・プロファイルのリストを使用するためにオペレーティング・システムのために作成される時、ユーザー・レジストリーのタイプは、EIM内のシステム・ユーザー・レジストリーとして参照されます。ユーザー・レジストリーが、特別なアプリケーションを使用するために作成される時、ユーザー・レジストリーのタイプは、EIM内のアプリケーション・ユーザー・レジストリーとして参照されます。

EIMで処理するユーザー・レジストリーの大半は、システム・ユーザー・レジストリーです。

Blank

Kerberos

Blank

Kerberos

- Kerberosは、ネットワーク認証プロトコルです。
- 信用されないネットワーク上で、クライアントからサーバー（その逆も）へのセキュアーな認証を確立するためにデザインされています。
- クライアントが、確立された接続を安全にするための暗号法を利用できるようにします。
  - クライアントに依存します。iSeries Accessは現在、Kerberos暗号化をサポートしていません。
- 業界に広く普及しており、プラットフォーム間の相互運用を考慮しています。
- 信用管理を簡単にします。
- RCF1510で定義されています。

RFC1510

The Kerberos Network Authentication Service

Notes:Kerberos

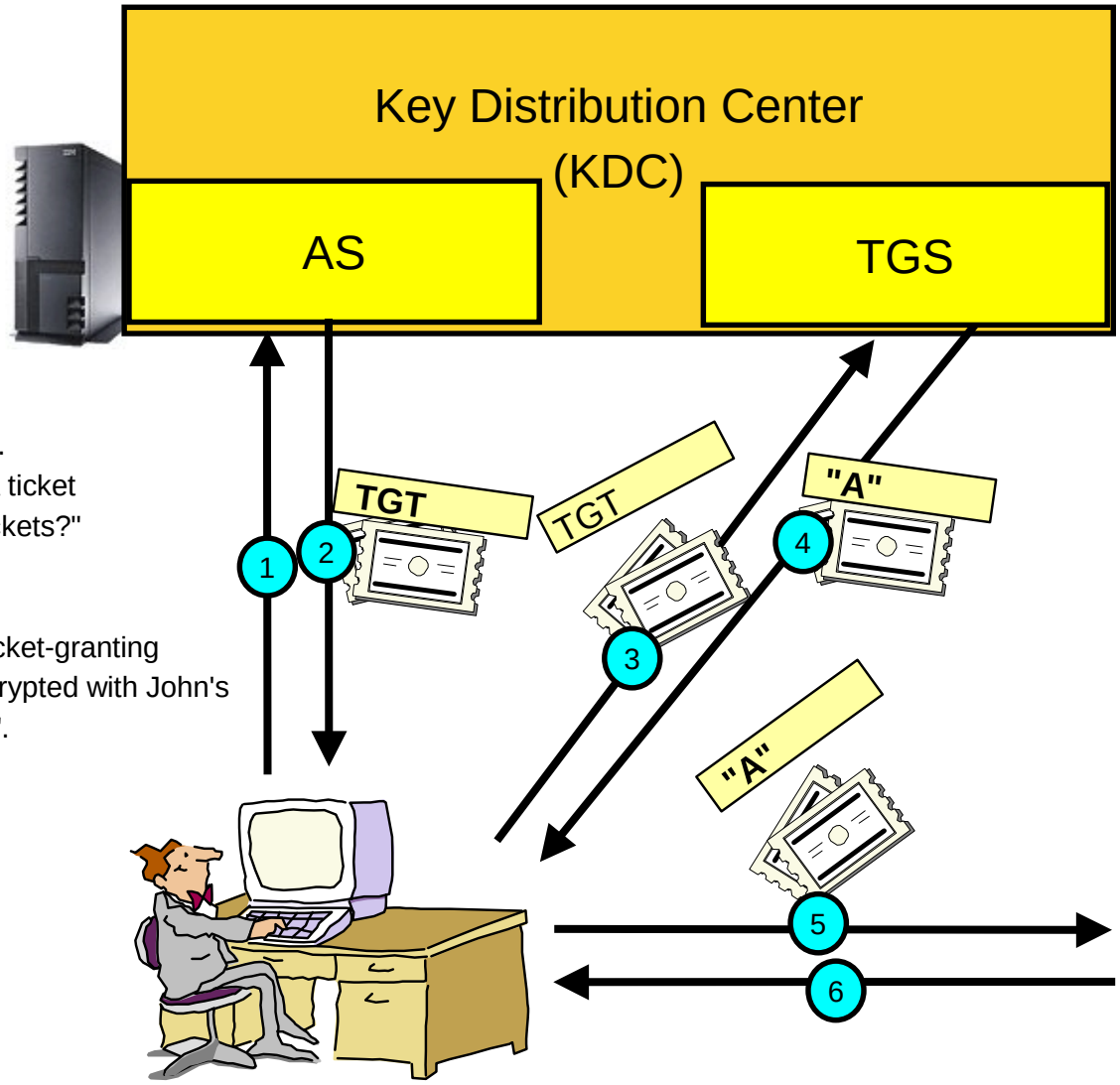
Kerberosシステムは、アテナ・プロジェクトの一部としてマサチューセッツ工科大学Massachusetts Institute of Technology (MIT)によって、1980年代にデザインされ開発されました。現在のKerberosのバージョンは5であり、それは、RFC1510 The Kerberos Network Authentication Service (V5) で標準化されています。詳細については、 http://www.ietf.org/rfc/rfc1510.txt を参照して下さい。

“Kerberosは、BSDオペレーティング・システムやX Windows Systemで使われているものに非常に類似した著作権許可の下、MITから自由に入手できます。MITはソース・フォームでKerberosを提供します。そのため、使いたい人は誰でもコードを調べ、そのコードが信頼できるものであることを確信するでしょう。加えて、専門的にサポートされた製品を好む人には、Kerberosは多くの異なるベンダーから製品として入手することができます。

まとめると、Kerberosはネットワーク・セキュリテーの問題に対するソリューションです。またそれは、全社に跨る情報システムを安全にすることを援助するための、ネットワークを越えた認証ツールと強力な暗号法を提供します。Kerberosが有用なものであることを見出して下さい。MITでは、Kerberosは、我々のインフォーメーション/テクノロジー アークテクチャーにとって非常に貴重なものです。“ 出展：MIT

Kerberos認証はそれ自身、セッションの残りが暗号化されることを自動的に暗示するものではありません。しかしながら、Kerberosは、セッション暗号化のためクライアント・プログラムによって使用される暗号キーの安全な交換を可能にします。例えば、iSeries Accessは、Kerberosの暗号部分を実装していません。しかし、iSeries Accessのトラフィックは、代わりにSSLによって暗号化されます。

Kerberos環境



1 **as_request:**
"Hi, I'm John.
Can I have a ticket
for getting tickets?"

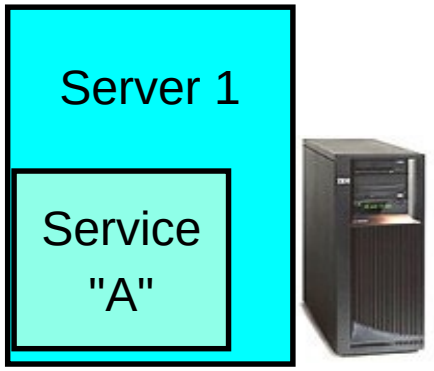
2 **as_reply:**
"Here's a ticket-granting
ticket, encrypted with John's
secret key".

3 **tgs_request:**
"Here is my TGT, could I have a
ticket for Service A? "

4 **tgs_reply:**
"Here's a ticket for Service A."

5 **ap_request:**
"Here is my ticket; let me use
your service. "

6 **ap_reply:**
"Welcome John! By the way,
here's the proof that I'm
Service A."



Notes:Kerberos

Kerberosプロトコルは、いくつかのサブ・プロトコルから成ります。クライアントがKerberosサーバーに証明書を要求することができる2つの順序があります。最初に、クライアントが接続を希望するサーバーのためのチケットの要求をクリアー・テキストで オーセンティケーション・サーバー Authentication Service (AS)に送ります。応答はクライアントのシークレット・キーで暗号化され送られます。たいてい、この要求は後ほどチケット・グランティング・サーバー ticket-granting server (TGS) で使用されるチケット・グランティング・チケット ticket-granting ticket (TGT) のためのものです。次に、クライアントがTGSに要求を送ります。クライアントは、Kerberos証明書を要求する他のアプリケーション・サーバーにコンタクトするのと同じ方法で、TGTをTGSに送ります。応答はTGTからのセッション・キーで暗号化されます。

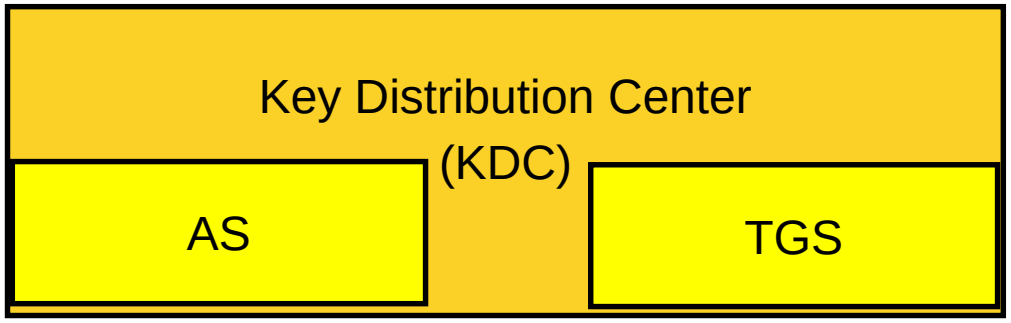
クライアントとサーバーは、最初は暗号化キーをシェアしません。クライアントが新しい立証者に対しそれ自身を認証する時はいつでも、新しい暗号キーを生成し、それを両者に安全に配布するためにオーセンティケーション・サーバーを信頼します。この新しい暗号化キーは、セッション・キーと呼ばれ、Kerberosチケットは、認証者に配布するために使用されます。

Kerberosチケットは、オーセンティケーション・サーバーによって発行された証明書で、サーバー・キーを使い暗号化されます。他の情報の中でも、チケットは、認証者に対するプリンシパルの認証、セッション・キーが発行されたプリンシパルの名前、セッション・キーの有効期限、のために使用されるランダムなセッション・キーを含んでいます。チケットは直接認証者に送られません。しかし、代わりにアプリケーション・リクエストの一部として立証者に転送するクライアントに送られます。チケットはサーバー・キーで暗号化されているので、オーセンティケーション・サーバーによってのみ理解されます。クライアントがチケットを変更することは不可能です。

Kerberosのコンポーネント: KDC

Key Distribution Center (KDC) は2つの主要なサービスを持っています。:

- オーセンティケーション・サーバーAuthentication Server (AS)
  - ASは、IDを証明する 共用シークレットを含みます。
  - 一度認証されれば、TGTが発行されます。
- チケット・グランティング・サーバーTicket-granting server (TGS)
  - サービス・チケットを発行する。(セッション・キーを含む)
  - チケット配布をトラックし続けない。



Notes:Kerberosのコンポーネント: KDC

Key Distribution Center (KDC)は、チケットとテンポラリーのセッション・キーを提供するネットワーク・サービスです。KDCはプリンシパル(ユーザーとサービス)および関連するシークレット・キーのデータベースを維持します。 それは、オーセンティケーション・サーバー Authentication Server (AS) とチケット・グランティング・サーバーTicket Granting Server (TGS) から成ります。KDCとして稼動する安全なマシンを使うことは重要です。誰かがKDCへアクセスした場合、全体のレルム(領域)が解決されます。

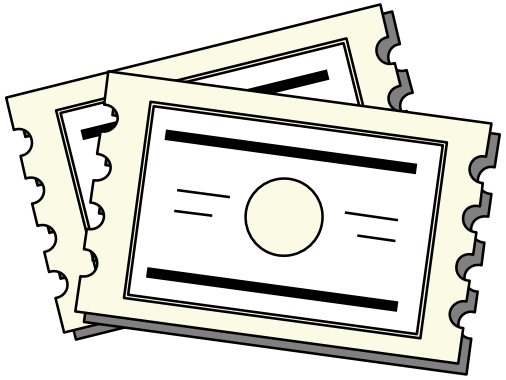
Note: KDCは、iSeriesサーバー上でサポートされません。

Kerberosのコンポーネント: チケット

- チケット: クライアントがそれ自身をサーバーまたはサービスに対して認証し、セッションを確立することを援助するレコード。
- チケット・グランティング・チケットTicket-granting ticket (TGT): 後にセッションのため使われるチケットを要求するためのチケット。一度正しい証明書がオーセンティケーション・サーバーに与えられれば、TGTが受信されます。

他のチケット:

- プロキシアブル/プロキシー・チケット: バックエンド・サーバーに対してクライアントを表すためにサーバーによって使用されるチケット
- フォワーダブル/フォワーデッド・チケット: クライアントの代わりにサービス・チケットを獲得するタスクを委任するチケット



Notes:Kerberosチケット

チケット:
クライアントがサーバーに対し、それ自身を認証することを援助するレコード。それは、サーバーのシークレット・キーを用いて封印されたクライアントのID, セッション・キー, タイムスタンプ, 他の情報を含みます。最近作成されたオーセンティケーターAuthenticator(*) に沿って贈られる時にクライアントの認証を提供するのみです。

チケット・グランティング・チケット (TGT):
一度初期認証が行われれば作成されるチケット。これは、各々のリクエストのためにシークレット・キーを使用することに代わり、TGSとの通信のための一時セッション・キーの使用を考慮しています。TGTはたいてい、8-10時間の時間制限を持っています。シークレット・キーとして、通常はるかに長い有効期間を持つでしょう。またTGTは、サービスに対する認証のためのチケットを獲得するために、クライアントによって使用されます。

プロキシアブル, プロキシーチケット:
プロキシアブル・チケットは、TGTに含まれるネットワーク・アドレス以外のアドレスを持つサービスのためのチケットを入手できるようにするチケット・グランティング・チケットです。フォワーダブル・チケットと違い、現在のTGTから新しいTGTを代理することはできません。サービス・チケットを代理するのみです。フォワーダブル・チケットは、ID全てを他のマシンに転送させ、プロキシアブル・チケットは特別のチケットを転送させるのみです。プロキシアブル・チケットは、サービスがプリンシパルに代わってタスクを実行することを可能にします。サービスは、特別の目的のためにプリンシパルのIDを引き受けることができなければなりません。プロキシアブル・チケットはKDCに、元のTGTに基づき異なるネットワーク・アドレスに対する新しいチケットを発行することができることを伝えます。プロキシアブル・チケットでは、パスワードは要求されません。

フォワーダブル・チケット:
フォワーダブル・チケットは、サーバーがリクエスターの証明書を他のサービスにパスすることを可能にします。これを行うためには、初期TGTは、フォワーダブル・オプションを持ってリクエストされなければなりません。そして、サーバーは証明書を委任することを許されます。それが使われる例は、ユーザーがリモート・システムにログインし、そのシステムでの認証を必要とする時、ローカルにログインしているようになることです。

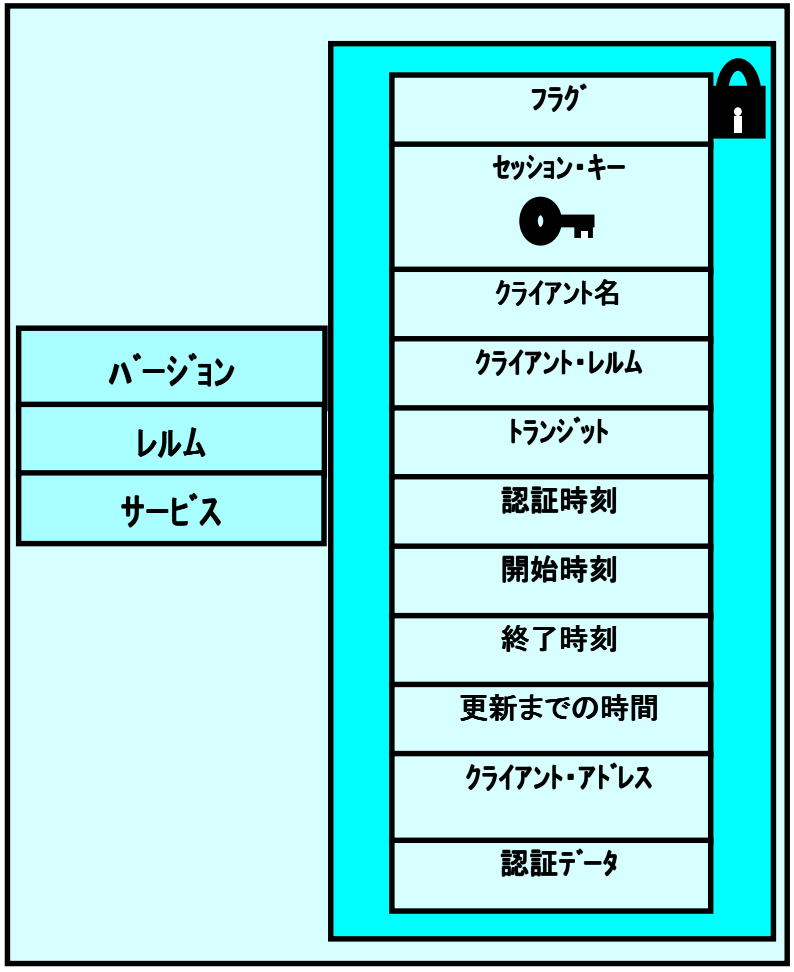
(*)オーセンティケーター: この情報(オーセンティケーター)が、クライアントとサーバーでのみ知られるセッション・キーを使用し、最近生成されたことを示す情報を含むレコードです。オーセンティケーターは、右のテーブルにリストされたフィールドから成ります。
(正確な仕様については、RFC1510を参照して下さい。):

- フィールド -	オーセンティケーターの値
authenticator-vno	バージョン・フォーマット
crealm	レルム
cname	クライアント名
cksum	リクエスト内のアプリケーション・データのチェックサム
cusec	タイムスタンプのマイクロ秒, 0-999999
ctime	ホストの時刻
subkey	この特定のセッションのための別のキーを含むことができます。
seq-number	シーケンス番号

'チケット'

この構造は、全てのチケットで同じです。
（正確な仕様については、RFC1510を参照して下さい。）

- **tkt-vno:** 使用されたKerberosのﾊﾞｰｼﾞｮﾝ（v.5）
- **Realm:** ﾁｹｯﾄを発行したﾚﾙﾑ名
- **Sname:** ﾁｹｯﾄが意図するｻｰﾊﾞｰ/ｻｰﾋﾞｽ名

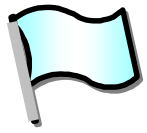


- **Enc-part:** ﾁｹｯﾄの暗号化された部分を保管（受領者のｼｰｸﾚｯﾄ･ｷｰを用いた）
- **Flags:** ﾁｹｯﾄが発行された時にｾｯﾄされた様々なｵﾌﾟｼｮﾝ。これがどのようなﾀｲﾌﾟのﾁｹｯﾄかを指示する。
- **Key:** ｸﾗｲｱﾝﾄとｻｰﾊﾞｰ間で使用されたｾｯｼｮﾝ･ｷｰ
- **cname:** ｸﾗｲｱﾝﾄ･ﾌﾟﾘﾝｼﾊﾟﾙのID名
- **crealm:** 最初にｸﾗｲｱﾝﾄを認証したﾚﾙﾑ
- **transited:** ﾕｰｻﾞｰ認証に参加するﾚﾙﾑ
- **Authtime:** 認証時刻
- **StartTime:** ﾁｹｯﾄが有効となる開始時刻
- **EndTime:** ﾁｹｯﾄが期限切れとなる時刻
- **RenewTill:** ﾁｹｯﾄが継続可能な場合、最終版の終了時刻
- **CAddr:** ﾁｹｯﾄが使用可能なｱﾄﾞﾚｽ
- **Authorization-data:** （ｵﾌﾟｼｮﾝ）発行者からｻｰﾊﾞｰ/ｻｰﾋﾞｽへﾃﾞｰﾀをﾊﾟｽするために使用されるﾌｨｰﾙﾄﾞ

Notes:チケット内のフラグ

以下のテーブルは、Kerberosバージョン5で使用されるフラグのリストです。

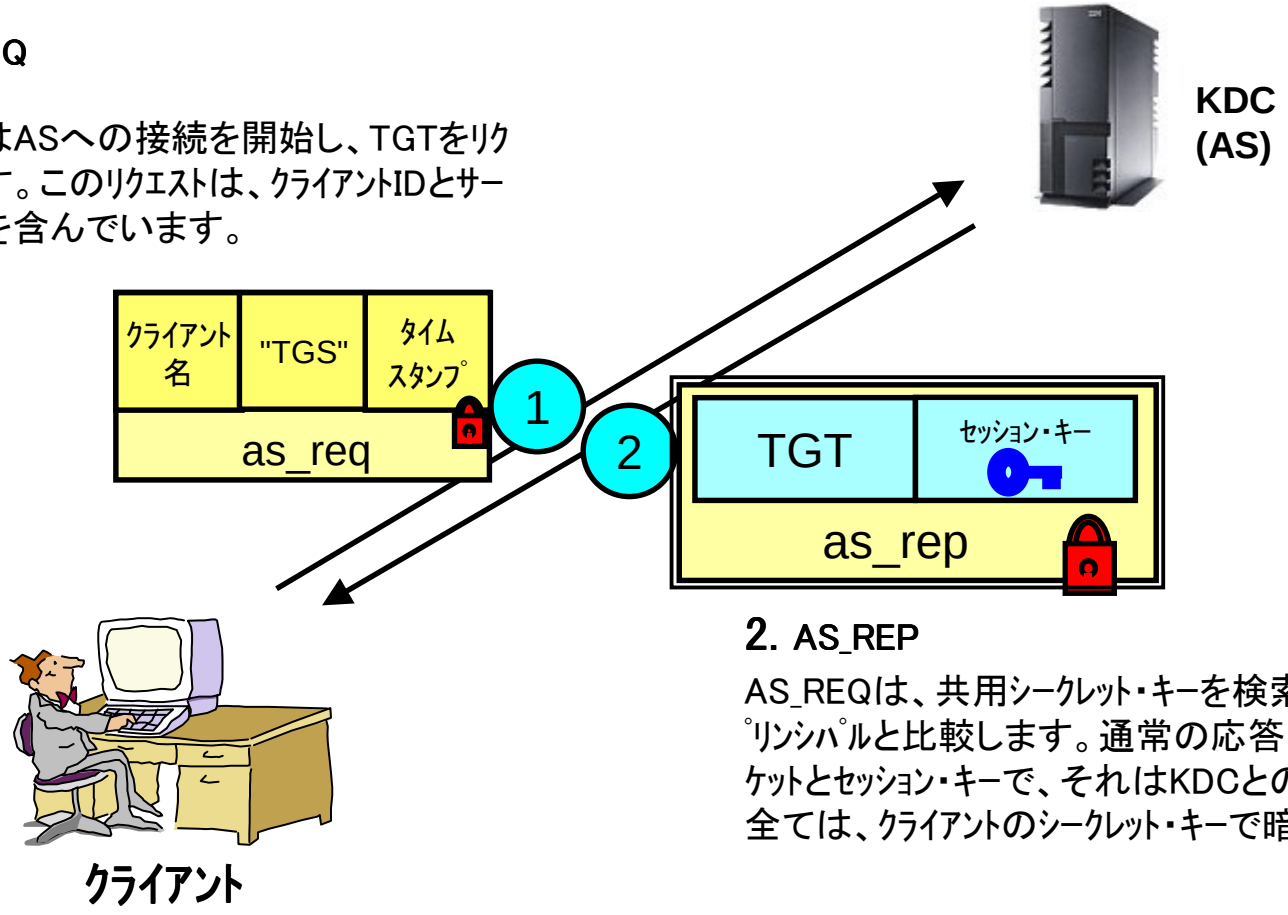
ビット	フラグ	記述
0	RESERVED	このフィールドは、将来の拡張のため予約
1	FORWARDABLE	FORWARDABLEフラグは、通常TGSによってのみ解釈されます。そして、終端のサーバーによって無視されます。このフラグがセットされる時、贈られたチケットに基づく異なるネットワーク・アドレスと共に新しいticket-granting ticketを発行することがOKであることをticket-granting serverに伝えます。
2	FORWARDED	このフラグがセットされる時、チケットが転送されたか、転送されたticket-granting ticketを含んでいる認証に基づき発行されたかを示します。
3	PROXIABLE	PROXIABLEフラグは、通常TGSによってのみ解釈されます。そして、終端のサーバーによって無視されます。PROXIABLEフラグはticket-granting serverにnon-ticket-granting ticketsのみが異なるネットワーク・アドレスと共に発行されること伝えることを除き、FORWARDABLEフラグと全く同じ解釈を持っています。
4	PROXY	このフラグがセットされる時、チケットがプロキシーであることを示します。
5	MAY-POSTDATE	MAY-POSTDATEフラグは、通常TGSによってのみ解釈されます。そして、終端のサーバーによって無視されます。このフラグは、ticket-granting serverに、先付け日付のチケットがthis ticket-granting ticketに基づき発行されるかもしれないことを伝えます。
6	POSTDATED	このフラグは、このチケットが先付け日付になったことを示します。エンド・サービスは、最初の認証がいつ発生したかを確認するために、authtimeフィールドをチェックすることができます。
7	INVALID	このフラグは、チケットが無効であることを示し、使用する前にKDCによって有効にされなければならないことを示します。アプリケーション・サーバーは、このフラグがセットされたチケットをリジェクトしなければなりません。
8	RENEWABLE	RENEWABLEフラグは、通常TGSによってのみ解釈されます。そして、終端のサーバーによって無視されます。(いくつかの特に慎重なサーバーは、継続チケットを許可しないかもしれません。)継続可能なチケットは、後に期限切れとなる後継チケットを入手するために使用することができます。
9	INITIAL	このフラグは、チケットがticket-granting ticketに基づいて発行されたのではなく、ASプロトコルを用いて発酵されたことを示します。
10	PRE-AUTHENT	このフラグは、初期認証の間、クライアントがチケットが発行される前にKDCによって認証されたことを示します。事前認証方式の強度は示されませんが、KDCで受け入れられます。
11	MAY-AUTHENT	このフラグは、初期認証に使われたプロトコルが、明示されたクライアントによって単独で処理されることを期待して、ハードウェアの使用を要求していることを示します。ハードウェア認証手法は、KDCによって選択され、手法の強度は示されておりません。
12-31	RESERVED	将来の使用のため予約



セッションの例

1. AS_REQ

ｸﾗｲｱﾝﾄはASへの接続を開始し、TGTをﾘｸｴｽﾄします。このﾘｸｴｽﾄは、ｸﾗｲｱﾝﾄIDとｻｰﾊﾞｰのIDを含んでいます。



2. AS_REP

AS_REQは、共用ｼｰｸﾚｯﾄ・ｷｰを検索するために存在するﾌﾟﾘﾝｼﾊﾟﾙと比較します。通常の応答は、ﾁｹｯﾄ・ｸﾞﾗﾝﾃｨﾝｸﾞ・ﾁｹｯﾄとｾｯｼｮﾝ・ｷｰで、それはKDCとの通信に使用されます。全ては、ｸﾗｲｱﾝﾄのｼｰｸﾚｯﾄ・ｷｰで暗号化されています。

Notes:セッションの例

1. AS_REQ:

クライアントは、ASへの接続を開始し、TGTをリクエストします。オプションとして、サーバーは、クライアントがタイムスタンプを暗号化するためのシークレット・キー(*)の使用により、それ自身を事前認証することを要求することができます。送信されたリクエストは、クリアー・テキストとオプションの暗号化されたタイムスタンプで、クライアントのIDとサーバー(**)のIDを含んでいます。

2. AS_REP:

AS_REQは、共用シークレット・キーを検索するために存在するプリンシパルと比較します。通常の応答は、チケット・グランティング・チケットとセッション・キーで、それはKDCとの通信に使用されます。全ては、クライアントのシークレット・キーで暗号化されています。

TGTを使用することにより、クライアントは新しいサービスに対する証明書のためのリクエストが行われるたびに、自分自身のシークレット・キーを使用する必要がなくなります。

通常TGTは、8-10時間の有効時間を持ちます。

(*) *シークレット・キー*は、ユーザーがKerberosサービスに最初にサインインするときに入力するパスワードに由来します。Windows2000環境では、シークレット・キーは、ドメインにログオンする時に生成されます。クラアイントとシークレット・キー保存のセキュリティー・レベルを上げるために、バイオメトリックスやスマートカードを使用することもできます。

(**) TGSサーバーのIDは、“krbtgt”です。

セッションの例 (続き)

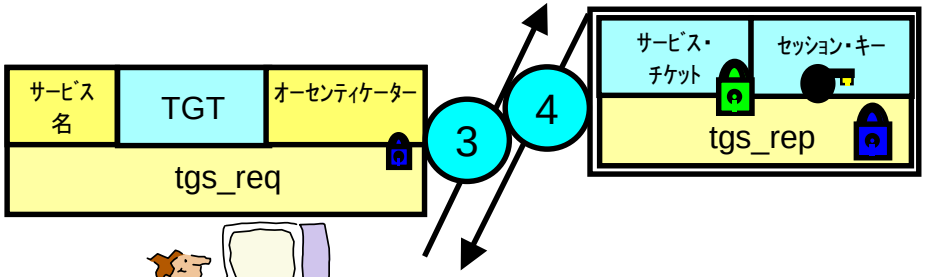
3. TGS_REQ

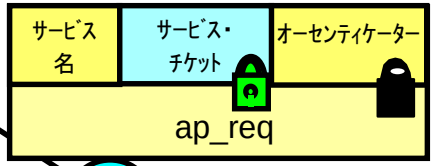
ｸﾗｲｱﾝﾄがｻｰﾋﾞｽへの接続を開始しようとする時、ｸﾗｲｱﾝﾄはまず、ﾁｹｯﾄ・ｸﾞﾗﾝﾃｨﾝｸﾞ・ｻｰﾊﾞｰからのｻｰﾋﾞｽ・ﾁｹｯﾄを要求します。この要求は、ｻｰﾋﾞｽ名, TGT, JohnのIDを提供するｵｰｾﾝﾃｨｹｰﾀｰから成ります。

KDC (TGS)

4. TGS_REP

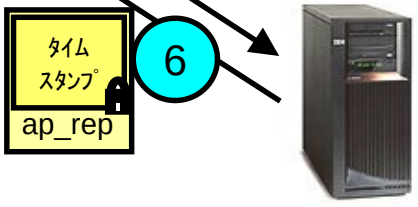
TGSは、要求されたｻｰﾋﾞｽのためのｻｰﾋﾞｽ・ﾁｹｯﾄと、ｾｯｼｮﾝ・ｷｰで応答します。この応答は、TGTと共に先に受信されたｾｯｼｮﾝ・ｷｰで暗号化されます。





5. AP_REQ

ここで、ｸﾗｲｱﾝﾄは、ｻｰﾋﾞｽ・ﾁｹｯﾄをｵｰｾﾝﾃｨｹｰﾀｰと一緒に転送することができます。ｻｰﾊﾞｰが、ﾁｹｯﾄが信用されたｻｰﾄﾞ・ﾊﾟｰﾃｨｰやKDCからのものであることを確認した後、ｾｯｼｮﾝが確立されます。



Server_A

6. AP_REP

ｵﾌﾟｼｮﾝで、ｸﾗｲｱﾝﾄはｻｰﾊﾞｰに、ﾀｲﾑｽﾀﾝﾌﾟを暗号化するためのｾｯｼｮﾝ・ｷｰを使用してそれ自身を認証することを要求することができます。

Notesセッションの例（続き）

これらのステップ（3～5)は、新しいサービスが要求される度に繰り返されます。

3. TGS_REQ

クライアントが、サービスとの接続を開始しようとする時、クライアントはまず、チケット･グランティング･サーバーからのサービス･チケットを要求します。この要求は、サービス名, TGT, JohnのIDを提供するオーセンティケーターから成ります。このトランザクションは、クライアントが、オーセンティケーターを暗号化するためにAS_REPから先に受信したセッション・キーを使います。

4. TGS_REP

TGSは、要求されたサービスのためのサービス･チケットと、セッション･キーで応答します。この応答は、TGTと共に先に受信されたｾｯｼｮﾝ・ｷｰで暗号化されます。最初のフィールドを除いて、クライアントはサービス・チケットを暗号化することができません。サービス・チケットは、目的のサービスに転送されるために使われるだけです。これは、セッション・キーがクライアントのためのチケットを除いて送られるからです。

5. AP_REQ

ここで、クライアントは、サービス･チケットをオーセンティケーターと一緒に転送することができます。サーバーが、チケットが信用されたサード･パーティーやKDCからのものであることを確認した後、セッションが確立されます。クライアントは、オーセンティケーターを暗号化するためにセッション・キーを使用します。そしてそれは、一度チケットがサーバーの共用シークレット・キーで復号化されれば、サーバーで読むことができます。

6. AP_REP

オプションで、クライアントはサーバーに、タイムスタンプを暗号化するためのセッション･キーを使用してそれ自身を認証することを要求することができます。これは、サーバーが実際にサービス・チケットをなんとか復号化したこと、および応答のためにセッション・キーを使用したことを証明します。

Kerberosの制限

- Kerberosクライアントが “セキュアー ”であることを必要とする
  - トロイの木馬や他のパスワード・スニフィング技術から保護できません。
  - クライアント・セキュリティー・ソリューションに全ての信用を置きます。
- オフラインでのブルートフォースや辞書攻撃には弱い
- あらゆるプラットフォームのあらゆる場所で利用可能でない
- クライアントは、同じ時刻に設定されなければならない
  (5分のずれ – 省略時)

Notes:制限

Kerberosには、適切に機能するための環境上のいくつかの前提があります。:

■“サービス妨害(Denial of service)” 攻撃は、Kerberosでは解決できません。これらのプロトコルには、侵入者が、アプリケーションが適切な認証ステップに参加するのを防ぐことができる余地があります。このような攻撃(システムにとっては普通の障害に見えるかもしれませんが)の検知と解決は、たいてい管理者やユーザーのような人に任されます。

■プリンシパルは、シークレット・キーを秘密にしておかなければなりません。侵入者が、どうにかしてプリンシパルのキーを盗めば、プリンシパルのふりをするか、正当なプリンシパルに対するサーバーになりすますことができます。

■“パスワード推測” 攻撃は、Kerberosによって解決されません。 ユーザーが平易なパスワードを選択していれば、攻撃者が辞書に連続的にエントリーされている単語やユーザーのパスワードから引き出されたキーの下で暗号化されたメッセージを使用し、繰り返し復号化を試みることにより、オフラインでの辞書攻撃を成功裏に開始するとが可能となります。

■ネットワーク上のホストはそれぞれ、他のホストの時刻と大まかに同期している時計を持たなければなりません。この同期を使用して、繰り返し検知を行う時のアプリケーション・サーバー保守の必要性を削減します。大まかさの度合いは、サーバー毎に構成することができます。時計がネットワークを越えて同期されている場合、時計の同期プロトコルは、ネットワーク攻撃者に対して安全なものでなければなりません。

■プリンシパルのIDは、短期間でリサイクルされません。アクセス・コントロールの典型的なモードは、特定のプリンシパルに対してアクセス権を与えるためにアクセス・コントロール・リスト(ACL)を使用することです。削除されたプリンシパルのための古いACLエントリーが残っており、プリンシパルIDを再利用する場合、新しいプリンシパルは、古いACLエントリーで指定された権限を継承します。プリンシパルIDを再利用しないことで、不注意なアクセスの危険性が取り除かれます。

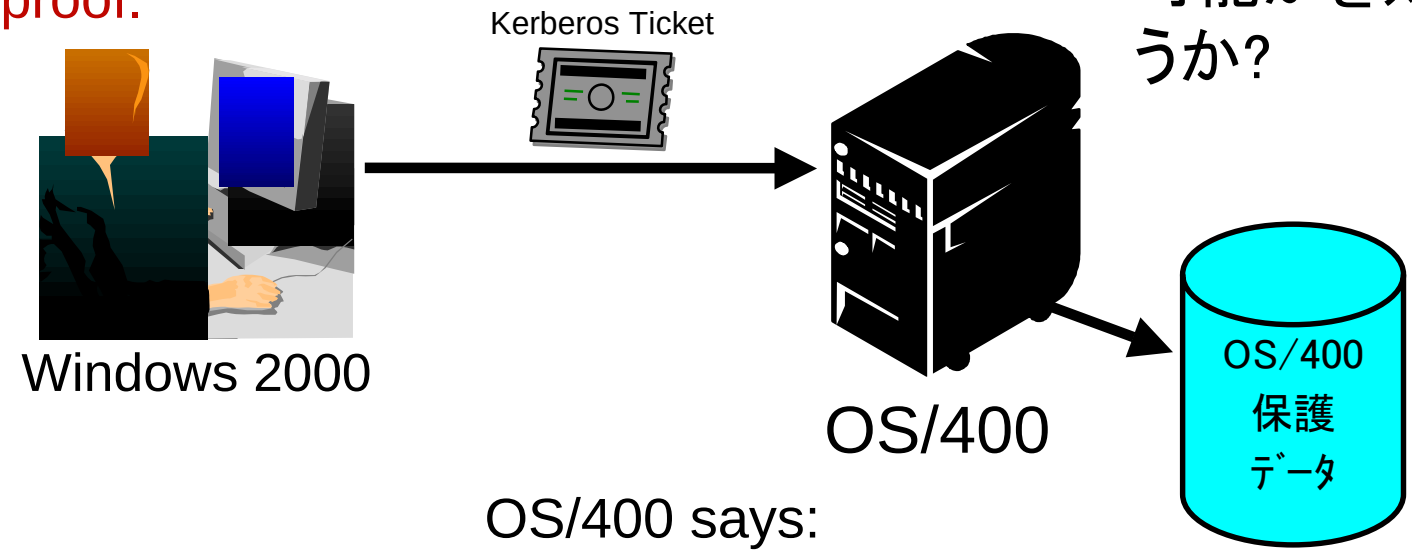
Kerberosの制限 (続き)

Kerberosが扱うのは認証のみです

OS/400はどのように、Windowsユーザーがどの OS/400資源にアクセス可能かを知るのでしょうか?

Client application says

"I am 'patriciaboats@MYCOM.WIN2KDOMAIN1' and here's proof. "



OS/400 says:

"I know who you are over **there**; but I need to know who you are over **here** to determine what you can access."

Notes:制限 (続き)

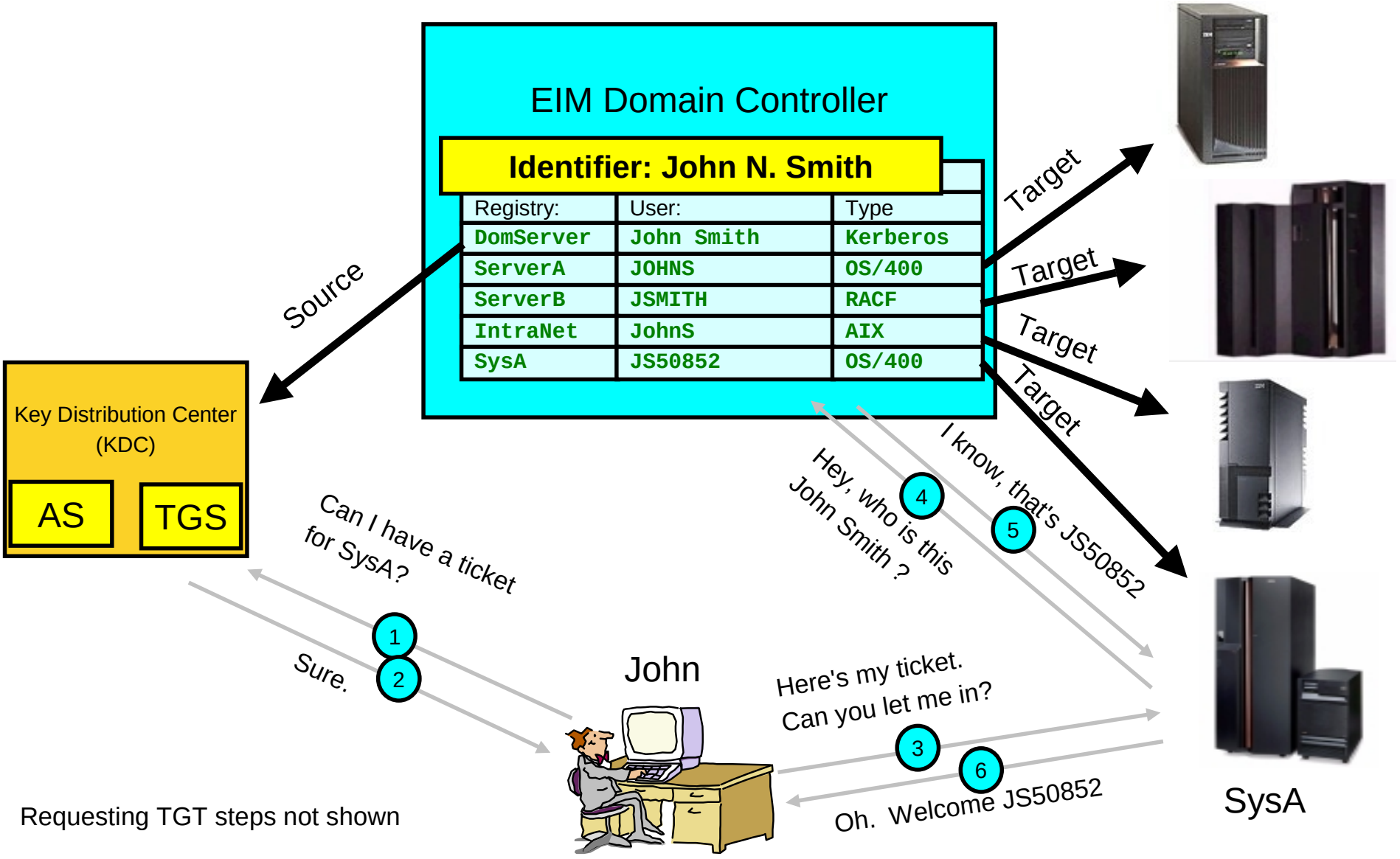
これは、認証(Kerberosが行うこと)と権限(オペレーティング・システムおよびアプリケーションのセキュリティー・レルム)の典型的な問題です。

Kerberos単独では、全てのユーザー・クラスのための複数のユーザー・レジストリーの全体的な問題を解決しません。

- 認証を扱う
- 管理者は依然、権限を心配しなければならない
- アプリケーション・プロバイダーは依然、以下のようなアプリケーションの作成を心配しなければならない。
  - 展開が容易
  - 管理が容易
  - セキュアーにすることが容易
  - 使うことが容易

テクノロジー・パズルのもう一つのピースが必要とされます！ ヒント: EIMとKerberosの協業は、次のページです。

EIMとKerberosの協業



Notes:EIMとKerberosの協業

以下のステップは、クライアントが既にTGTを持っていると仮定し、いかにEIMとKerberosがシングル・サインオンに利用されるかをサマライズしています。:

1.) サービスのための証明書がTGSから要求されます。

2.) *Sys_A* 用のサービス・チケットが返されます。

3.) クライアントは、サービス・チケットを使用し、SysA上のサービスへのアクセスを要求します。

4.) EIMリクエストを処理できる*Sys_A*は、EIMドメイン・コントローラーにユーザーIDを転送するためにEIM APIを使用します。
EIMコントローラーは、EIMデータベース内のIDを見つけるために、**“ソース”** のユーザーとレジストリーを見ます。

5.) EIMサーバーは、**“ターゲット”** のレジストリー・エントリーにあるユーザーIDを返します。

6.) *Sys_A*は、Johnのための接続を開き、OS/400ユーザーJS50852として妥当な権限を与えます。

サマリー

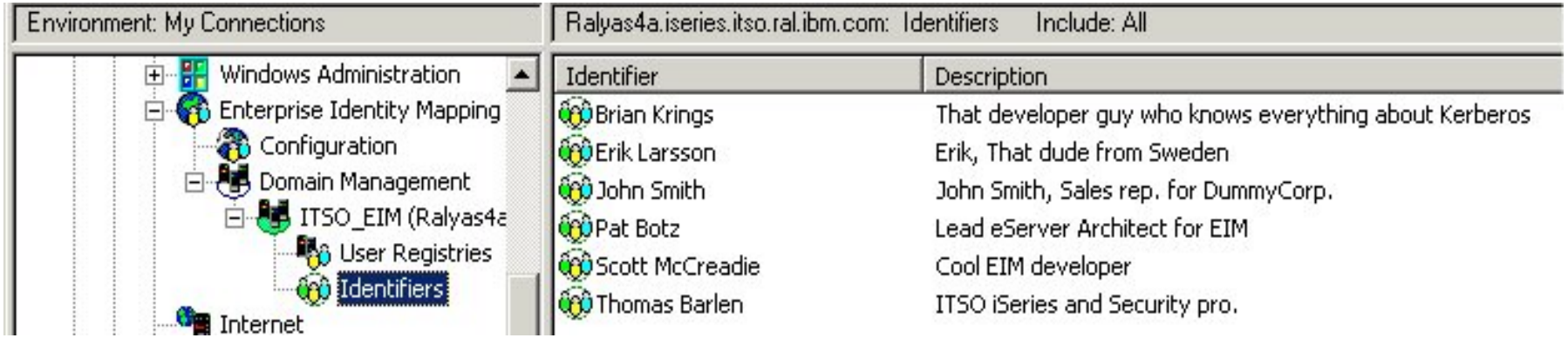
- シングル・サインオンのためにEIMをセットアップする時:
  - KDCにプリンシパルを追加
  - RALYAS4AにNASを構成
  - RALYAS4AにEIMサービスを構成
  - EIM IDの作成とユーザーのマップ
  - iSeries Navigatorの認証方法を変更
  - EIMを使用するためにRALYAS4Bを構成
- いくつかの考えられる次のステップ:
  - 多くのIDを追加
  - iSeries Accessが “Kerberized” Telnetを使えるようにする
  - NetServerがKerberosを使うようにする
  - ディレクトリーのSSLを可能にする

Notes:サマリー

今行われたステップは、EIM対応の環境への始まりを与えるものです。他のレジストリーがEIM対応になるように、レジストリーをEIMドメインに追加することができます。それは、既存の認証手法（ユーザーIDとパスワードであると思われますが）に影響を与えるべきでありません。コントロールされたペースでユーザーを取り入れることが可能です。一度ユーザーがEIMでマップされ、Kerberos認証を使用しているなら、iSeriesユーザー・プロファイルの**PASSWORD**値は、***NONE**にセットすることができ、ユーザーが他の手法を使用してサインオンすることを防止します。

EIMドメインに更にユーザーを追加するためには、ステップ4を繰り返して下さい。ターミナル・エミュレーションにiSeries Accessを使用しているなら、iSeriesナビゲーターと同じ認証手法を使うことができます。（*IBM Personal Communications* は現在、Kerberos認証をサポートしていません。）EIMとKerberosでサインオンをバイパスしたい時には、システム値 **QRMTSIGN**が、***FRCSIGNON** 以外にセットされていることを確認して下さい。

省略時には、ディレクトリーに接続する際、LDAPはユーザー名とパスワードをクリアー・テキストで送信します。従って、EIMドメイン内のシステムは、SSLを使用することを強く勧めます。オプションでEIMコントローラーでそれ自身を認証するためのKerberosが勧められます。



関連出版物

このセクションでリストされた出版物は、このワークショップでカバーされたトピックスをより詳細に調べるために特に適しているものです。

International Technical Support Organization Publications

- For information on ordering ITSO publications, visit us at **http://www.redbooks.ibm.com** (Internet Web site)
  or
- **http://w3.itso.ibm.com** (intranet Web site)

技術支援は、http://www.ibm.com/support and **http://w3.ibm.com/support を参照して下さい。**

Redbooks on CD-ROMs

- Redbooks are available on CD-ROMs.

CD-ROM Title	Collection Kit Number
System/390 Redbooks Collection	SK2T-2177
Networking and Systems Management Redbooks Collection	SK2T-6022
Transaction Processing and Data Management Redbook	SK2T-8038
AS/400 Redbooks Collection	SK2T-2849
RS/6000 Redbooks Collection (HTML, BkMgr)	SK2T-8040
RS/6000 Redbooks Collection (PostScript)	SK2T-8041
Application Development Redbooks Collection	SK2T-8037
Personal Systems Redbooks Collection	SK2T-8042

関連出版物 – 続き

他の出版物

- ***これらの出版物も、詳細情報源として妥当なものです。:***

Title	Publication Number
The Kerberos Network Authentication Service (V5), RFC1510	http://www.ietf.org/rfc/rfc1510.txt
Microsoft's Active Directory home page	http://www.microsoft.com/activedirectory
V5R2 iSeries Information Center, Security topics	http://www.iseries.ibm.com/infocenter
Kerberos, A Network Authentication System	ISBN 0-201-37924-4
Implementation and Practical Use of LDAP on the IBM ~ iSeries Server	SG24-6193