データベースシステム　122－33
(2000. 7. 27)

Webtopシステムにおけるデータベースアクセスのセキュリティ実現
― LDAPによるユーザ管理の一応用 ―

小林智恵子　原嶋秀次　山田朝彦

(株)東芝　情報・社会システム社　SI技術開発センター
〒183-8512　東京都府中市片町3-22
TEL 042-340-6638
{ chieko , haras , yamada } @sitc.toshiba.co.jp

あらまし　現在、報告者らはWebtopシステムのセキュリティ機能の研究・開発に取り組んでいる。Webtopシステムにおいて、ユーザを正しく認識し、Webコンテンツ、Webアプリケーション、データベースなど、ユーザ権限に応じたアクセスの実現を目指している。本稿では、Webサーバへのログイン IDによる、Webアプリケーションからデータベースまでのシングルサインオンを実現する方式を報告する。ディレクトリサーバ(LDAPサーバ)で一元管理するユーザ情報を利用して、Webアプリケーションからデータベースへのログインを実行する。これによりユーザ権限に応じた安全性の高いシステムを構築することが可能になった。

キーワード　LDAP、Directory Server、情報管理、アクセス制御、シングルサインオン、データベース

User Authorization for Accessing Databases in Webtop Systems : Management of User Information using LDAP

Chieko KOBAYASHI　Shuji HARASHIMA　Asahiko YAMADA

TOSHIBA CORPORATION
INFORMATION AND INDUSTRIAL SYSTEMS & SERVICES COMPANY
SYSTEM INTEGRATION TECHNOLOGY CENTER
3-22, KATAMACHI FUCHU-SHI, TOKYO 183-8512　JAPAN
TEL +81-42-340-6638
{ chieko , haras , yamada } @sitc.toshiba.co.jp

Abstract　We are researching and developing security functions of Webtop system. In our Webtop system, we aim that a user is recognized correctly and that the access to the information resources, such as Web contents, Web application, and database, is permitted only if the user is authorised to use them. This paper reports implementation of the system which realizes delegation and single sign-on from Web application to database. Database system authenticates the user with the authentication information passed by the application which is stored in directory server (LDAP server) with confidentiality. This enables us to build the highly secure system in which a user can access the information if and only if he or she have the right to.

Key words　LDAP, Directory Server, information management, access control, single sign-on, database

1. はじめに

インターネットを用いた電子商取引(以下 EC)は、時間、距離の壁を取り払った取引を可能とし、販売者、顧客に多くのメリットをもたらした。一方で個人情報の漏洩やデータの改ざんなど、セキュリティ面の多くの問題を抱えている。このように、EC をはじめ Webtop ベースのシステム(以下 Webtop システム)ではセキュリティは必須の要素となっている。

弊社では、Webtop システムの標準的なソリューション体系を提案しており、そのセキュリティ機能として、ユーザ認証、アクセス制御、秘匿、監査などの機能の研究・開発に取り組んでいる。具体的には Webtop システムにおいて、ユーザを正しく認識し、Webコンテンツ、Web アプリケーション、データベースなどの情報資源へのアクセスをユーザ権限に応じて可能にすること、これらの情報資源の不正利用や漏洩を防止すること、問題発生時には原因究明(ログ解析)を行うことである。

ユーザ権限などのセキュリティ情報の一元管理には、ディレクトリサーバの活用が有効である。TCP/IP 上でディレクトリにアクセスするためのプロトコル LDAP (Lightweight Directory Access Protocol)[4]が標準化されてからディレクトリサーバの利用は急速に一般化した。ディレクトリサーバはこれまでのような単なる電話帳検索的な使い方から、情報を一元管理するレポジトリとして、情報システムの中核となるコンポーネントとして利用されるようになっている。報告者らは、ディレクトリサーバ上のユーザ情報をもとに Webtop システムへのログイン ID によるクライアント(ブラウザ)から Web サーバ、Web アプリケーションサーバ、データベース管理システム(以下 DBMS)までのシングルサインオンを実現した。シングルサインオンとは、アプリケーションごと、サーバごとに複数回発生するユーザ認証を、エンドユーザに対しては一度しか行わせずにシステムが代行する機構である。

本稿では、この中で Web アプリケーションサーバ上のアプリケーションから DBMS へのアクセスにおいて、ブラウザを利用するユーザの権限でログインを実現するメカニズムについて報告する。ディレクトリサーバで管理された認証情報を、今回作成したライブラリが利用して DBMS にログインする。この時、アプリケーションは認証情報を知ることはなく、安全性の高いシステムを構築することができる。

2. 背景

インターネットによる EC は、企業－消費者間で 1999 年の 180 億ドルから 2003 年に 1080 億ドル、企業間取り引きでは、1999 年の 1090 億ドルから 2003 年に 1.3 兆ドルになると言われている(フォレスター・リサーチによる米国における市場規模予測)[1]。このように、我々の一般生活においても企業活動においても、一般的な商取引手段となっている。しかし、EC におけるセキュリティについては、技術的課題の解決、関連法規の整備、社会的ルールの普及など、いずれをとっても十分であるとは言えない。EC においても、データベースは一般の企業システムと同様に非常に重要な位置を占め、また、Webtop システムが主流になることから Webtop システムにおけるデータベースの利用が重要になる。

DBMS でも、ユーザ認証、アクセス制御が早い段階から製品レベルに組み込まれていた。また、セキュリティを中心とした研究[2][3]がなされているほか、Trusted Oracle[6]などの製品も出荷されている。 Oracle8i[7]では Kerberos[5]にもとづく認証サーバでのユーザ認証が可能となっている。しかし、現状では認証サーバが限定されることや SQL Server など他の DBMS が存在した場合の統合管理が困難であることなどの問題がある。

現在、一般的に行われている Webtop システムの運用では、Web アプリケーション構築時に、あらかじめ、Web アプリケーションが DBMS にアクセスするためのログイン情報を DBMS に登録しておき、Web 上でアクセスを許可されたユーザ権限とは別に、このログイン情報で Web アプリケーションは DBMS へアクセスしている。この方式では DBMS アクセス用のユーザ情報が知られやすく、データの安全性を保証することは困難である。

報告者らは、ユーザデータの安全性を確保するために、Web アプリケーションにログインユーザの認証情報をあらかじめ記述することなく、Web 上でアクセス

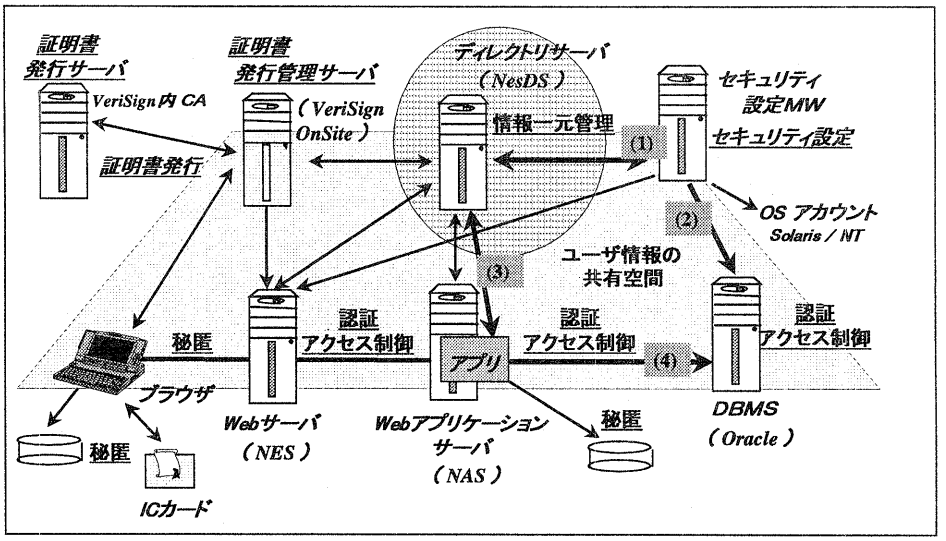


図 3.1　Webtop セキュリティ機能

を許可されたユーザの権限で DBMS へアクセスする方式を提案する。

3.　データベースアクセスのセキュリティ

3.1.　Webtop セキュリティ機能

Webtop セキュリティ機能は、弊社の提供するソリューション体系において、Webtop コンピューティングを実現させる基盤環境"Webtop プラットフォーム"の一機能と位置付けられる。Webtop プラットフォームはインターネット、イントラネット上で業務システムを開発／運用するための基盤技術であり、UNIX[1]、Windows[2]が混在するオープンな環境下において、アプリケーション開発や運用に必要な標準機能を提供し、開発したアプリケーションに対して信頼性や拡張性を保証する。Webtop セキュリティ機能は、ディレクトリサーバとセキュリティ設定ミドルウェア(以下、セキュリティ設定 MW)を軸にして統合的なセキュリティ管理を実現している。セキュリティ設定 MW については第 3.2 節で説明する。

図 3.1 に Webtop セキュリティ機能を示す。独自に開発したセキュリティ設定 MW を通して、ユーザ情報のディレクトリサーバへの格納をはじめとするセキュリティ情報の設定を行う。IC カードを利用することにより Webtop システムでのシングルサインオンを実現している。図中の(1)から(4)が本稿で報告する機能である。図中の(1)(2)は DBMS へのシングルサインオンを行うための登録系処理であり、セキュリティ設定 MW からディレクトリサーバと DBMS へ情報を登録する機能である。また、図中(3)(4)は DBMS へのシングルサインオンを行うための実行系処理であり、Web アプリケーションサーバ上で動作する Web アプリケーションに組み込んで利用するライブラリの処理である。ライブラリは DBMS に接続するために必要な認証情報をディレクトリサーバから取得し、DBMS へ接続する。この際、Web アプリケーションには DBMS への接続情報のみが渡され、ユーザ情報はライブラリから Web アプリケーションに渡されることなく、安全なシステムとすることができる。詳細はそれぞれ第 3.2 節、第 3.3 節で説明する。

今回の開発では、Web サーバは Netscape Enterprise Server[3](以下 NES)、ディレクトリサーバは Netscape Directory Server[4](以下 NesDS)、Web アプリ

[1] UNIX は The Open Group の米国およびその他の国における商標。
[2] Windows、WindowsNT は Microsoft 社の商標。
[3] Netscape Enterprise Server は Netscape Communications 社の商標。
[4] Netscape Directory Server は Netscape Communications 社の商標。

ケーションサーバは Netscape Application Server[5](以下 NAS)、DBMS は Oracle[6]を対象とした。

3.2. セキュリティ設定 MW

第 1 章で述べたように、セキュリティ情報の一元管理にはディレクトリサーバの活用が有効である。しかし、ディレクトリサーバにおいて、膨大な量のセキュリティデータの管理が必要なこと、シングルサインオンに向けて OS などシステムコンポーネントとの連携・統合が必要なことなどの問題がある。例えば、ソフトウェアによってはデータベースのログイン情報のように LDAP 未対応のものもあること、ユーザ情報は LDAP サーバで管理できても Web サーバに見られるようにアクセス制御情報(以下 ACL 情報)の LDAP 上での管理までは至っていないこと、などが課題としてあげられる。

そこで、報告者らが開発したセキュリティ設定 MW は、以下に述べるような方法で前記の課題を一部解決する。セキュリティ設定 MW は分散した複数サーバにおけるユーザ／グループ情報やアクセス制御情報など、セキュリティ情報を一元管理するための支援ツールであるが、LDAP 上のユーザ情報と連携して DBMS へのデータ登録／変更／削除を行う。その他に、WindowsNT や Network Information Service (NIS)管理の OS アカウント情報やファイルアクセス制御情報、また、Web サーバ上のコンテンツに関する ACL 情報を統合的に管理することができる。セキュリティ設定 MW はサーバ／エージェントモデルで実装しており、OS や Web サーバや DBMS はエージェントとなる。従って、エージェントとなるコンポーネント用の設定コマンドを作成／追加することが可能であり、情報の一元管理を容易にしている。これにより、統一したポリシーでセキュリティ情報を管理することができる。

3.2.1. ディレクトリサーバへの登録／変更／削除

DBMS と連携させるために必要な情報を表 3.2.1 の通り定義した。ディレクトリサーバには個々のデータベースのスキーマ情報を持ち込まず、セキュリティポリシーのみを管理する方式とした。ディレクトリサーバには LDAP-API を用いて情報を登録／変更／削除する。なお、DBMS ユーザ用パスワード情報は暗号化して登録する。後述する DB セキュリティライブラリが情報を取得する際、パスワード情報が漏洩することを防ぐためである。暗号化については第 3.3 節で説明する。

3.2.2. DBMS への登録／変更／削除

表 3.2.1 に示すデータを DBMS に対して登録／変更／削除する。ただし、フラグ情報は、ディレクトリ上の情報と DBMS 上の情報との整合性管理のために、ディレクトリで利用するものである。例えば、情報の管理として、ディレクトリサーバから DBMS に関する情報を削除するが DBMS 側には残したい場合は、削除フラグを false にすることにより、DBMS 側には情報をそのまま残し、ディレクトリからのみ削除できる。DBMS には SQL*Net などのネットワークソフトにより通信し、SQL 文を実行する。

属性	型	備考
ユーザ名	文字列	
パスワード	バイナリ	ディレクトリには暗号化して登録する.
インスタンス名	文字列	
接続名	文字列	
オーナーテーブル	文字列	選択肢 −／R／W／RW
その他のテーブル	文字列	選択肢 −／R／W／RW
オーナープロシージャ	文字列	選択肢 −／EX
その他のプロシージャ	文字列	選択肢 −／EX
ロール	文字列	
ユーザ表領域	文字列	
一時的な表領域	文字列	
DBMS ユーザ作成フラグ	文字列	選択肢 true／false
DBMS ユーザ削除フラグ	文字列	選択肢 true／false

表 3.2.1 スキーマ定義

3.3. データベースセキュリティライブラリ

データベースセキュリティライブラリ(以下 DB セキュリティライブラリ)利用する API を提供し、ディレクトリサーバで一元管理している認証情報を使って DBMS への接続を可能にする。

図 3.3 に DB セキュリティライブラリの処理を示す。

[5] Netscape Application Server は Netscape Communications 社の商標。
[6] Oracle は Oracle 社の商標。
[7] VeriSign OnSite は VeriSign 社の商標。

3.3.1. ディレクトリサーバからの情報取得

図3.3の1、2、3の処理で実現される。ユーザ認証ID (user ID) は Web サーバから Web アプリケーションサーバを経て、Webアプリケーションが取得する。その後、以下の処理を実行する。

1．Web アプリケーションから userID を受け取る。
2．取得した userID をキーに、ディレクトリサーバを検索する。
3．ディレクトリサーバに一致する userID が存在すれば、暗号化されたパスワードと接続名を取得する。なお、一致する userID が存在しない場合には DBMS への接続は許可されない。

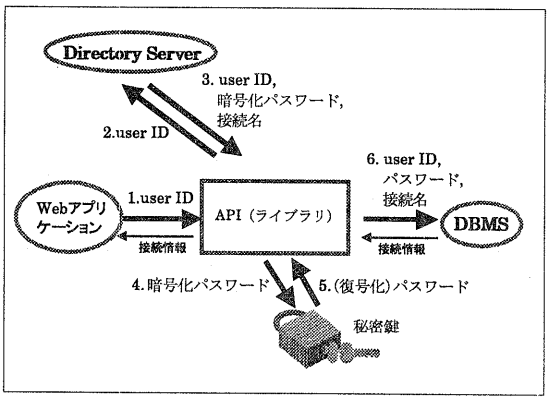


図 3.3　DB セキュリティライブラリの処理

3.3.2. DBMS への接続

図 3.3 の 4、5、6 の処理で実現される。ディレクトリサーバから取得した DBMS への接続情報を用いて、認証／接続を行う。なお、ディレクトリサーバから取得するパスワード情報が漏れないように、トリプル DES[8]を用いて暗号化している。暗号化されたパスワードは DB セキュリティライブラリの中で秘密カギを使って復号化する。DBMS 接続時は復号化されたパスワードを使ってアクセスする。以下に API の処理概要を示す。

4．ディレクトリサーバから取得した暗号化されたパスワードを秘密カギを用いて復号化する。
5．復号化されたパスワードを取得する。
6．userID、(復号化された)パスワード、接続名を使って DBMS へ接続する。

4. 考察

第 3 章で説明したように、セキュリティ設定 MW および DB セキュリティライブラリにより、Webtop システムにおけるデータベースまでを含めたシングルサインオンを実現できた。これにより、安全性の高いシステムの構築が可能となった。以下に現在検討中のテーマについて述べる。

4.1. セキュリティ設定 MW

(1)グループ概念の導入

すべてのユーザを DBMS に登録して、そのユーザでログインすることが適当でない場合もある。このために、ユーザのグループを定義し、グループ ID で DBMS にログインし、グループの権限でアクセスするという方法も考えられる。このためには、ディレクトリサーバのスキーマ変更と共に、データベースアクセスライブラリの変更が必要となる。グループとユーザの併用も含めたDBMSのテーブルに対して、グループおよびユーザの権限を定義する必要がある。

(2)データベーススキーマ情報の利用

今回は、ディレクトリサーバに、データベースのスキーマを持ち込まない方法を選択した。大量のスキーマ情報をディレクトリサーバ上に持つのは、望ましくないと考えたことと、運用とともに変化するスキーマ情報に各ユーザの権限設定を自動的にあわせるのは困難と考えたためである。しかし、用途によっては、スキーマに応じた権限設定が必要なケースも考えられ、その対応方法が必要である。

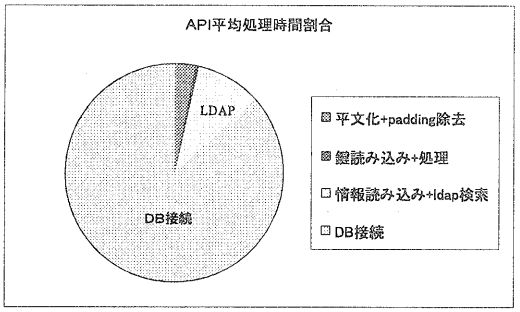
(3)複数インスタンスへの対応方法

データベース中に定義された複数のインスタンスに対して、各ユーザのアクセス権限を設定する方法が必要である。

4.2. DB セキュリティライブラリ

今回作成した DB セキュリティライブラリの性能について検証した。暗号化方式には、トリプル DES を利用した。トリプル DES は対称暗号方式の一つで最も利用されているものである。暗号処理の欠点としてアルゴリズムの実行に時間がかかる点があげられるが、実際に処理時間を測定して影響を調べてみた。

図 4.1 に連続 100 回実行した各処理時間の平均値を示す。この図に示すように、トリプル DES の処理時間（平文化+padding 除去と鍵読み込み処理）は全体の3%程度であり、ほとんど影響しないことがわかった。



平文化+padding除去	鍵読み込み+処理	情報読み込み+ldap検索	DB接続	全体処理
4.62	0.7	14.83	130.67	150.82

連続100回実行の平均値
（単位：ミリ秒）

図 4.1　DB セキュリティライブラリ処理時間

5.　今後の課題

本方式により、Web アプリケーションから DBMS へのアクセスにおいて、ブラウザを利用するユーザの権限でアクセスを可能にし、Webtop システム上のシングルサインオンを実現した。しかし、システム全体でのセキュリティ確保の観点から引き続き検討・開発を行う必要がある。特に下記のテーマについて重要であると考える。

5.1.　データベース内におけるデータの安全性

データベース内におけるデータの安全性を確保する方法である。現状ではデータベースにログインできれば、そのログインユーザの読み出し権限のあるデータはすべてみることが可能である。しかし、より高い安全性が必要なシステムにおいては、データを暗号化して格納するなどの方法が必要となる。

5.2.　より複雑なシステム構成への対応

システムにおけるディレクトリサーバの位置付けの明確化である。今回は、ユーザ認証情報の管理を中心に用いたが、一般にはシステム構成情報など他の用途にも利用する。従って、全体としてのディレクトリサーバへの負荷の検討や、複数のディレクトリサーバによる負荷分散などの検討が必要となる。

最後に、より複雑なシステム構成への対応を検討する必要がある。ディレクトリサーバのみでなく、複数の Web サーバ、Web アプリケーションサーバ、DBMS が存在する場合の構成や、複数の Web アプリケーションが存在する場合への対応などが必要である。

6.　まとめ

本方式により、Web アプリケーションにログインするユーザの認証情報を第三者に知られること無く、DBMS へのログインが可能となり、Web アプリケーションサーバ内でのユーザデータの安全性を高めることが可能となった。また、本方式では、セキュリティ設定MWが DBMS にユーザ情報を設定するので、複数の種類の異なる DBMS が存在する場合への対応が容易である。

さらに、データベースへのアクセス結果に応じて処理を変える仕組みを組み込むことによって、ユーザごとの処理を、データベースの登録内容で変更するといった利用方法も考えることができる。これらをふまえ、今後、Webtop システムにおけるデータベースのセキュリティという観点で研究・開発を進める予定である。

<参考文献>

[1]　前川：暗号技術と電子商取引、電子情報通信学会誌 Col.83, No.2, pp.96-100, 2000.2

[2]　F.Cuppens, G.Trouessin: Information Flow Controls vs Inference Controls: An Integrated Approach, Computer Security ESORICS 94, Springer-Verlag, pp.447-468, 1994.

[3]　K.R. Dittrich et.al. : Current Trends in Database Technology and Their Impact on Security Concepts, Database Security VIII, North-Holland, pp.11-33, 1994.

[4]　Lightweight Directory Access Protocol(V3) RFC2251, 1997.12

[5]　J.G.Steiner, C.Neuman, J.I.Schiller : Kerberos: An Authentication Service for Open Network Systems, proc. of. the Winter 1988 Usenix Conference, 1988.2.

[6]　Trusted Oracle7 Server Administrator's Guide : Introduction to Trusted Oracle7

[7]　Oracle テクニカル・ホワイトペーパー:Oracle8i™ でのデータベースセキュリティ, 1998.11. ,Oracle8i のセキュリティ:新機能と機密保護ソリューション, 1999.11

[8]　B. Schneier, "Applied Cryptography Second Edition", John Wiley & Sons, Inc., 1996