社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2006－DD－58（12）
2006－EIP－34（12）
2006／11／30

Web システムにおけるデータベース監査ログの課題と解決法

松永 豊† 　大場 みち子‡

†東京エレクトロン デバイス株式会社 CN 事業本部 〒107-8481 東京都港区赤坂 5-3-6
‡株式会社日立製作所 ソフトウェア事業部 新分野事業推進室
〒244-8555 横浜市戸塚区戸塚町 5030 番地
E-mail: †matsu@kabuki.tel.co.jp, ‡michiko.oba.cq@hitachi.com

あらまし　情報漏洩などの脅威への対応はますます重要性を増しているが、アプリケーションの開発技術や構造が複雑化しているため、ネットワーク主体の従来の防止法は限界が見えてきている。その結果、より根本的な対策が求められており、データベースの保護に関心が集まっている。一方、内部統制の要求による、業務システムの正しい利用を証明する為のデータアクセスの監視・監査について、今までの方法では解決できない問題が明らかになってきている。いずれの場合も中核となるのはアクセスを監視･記録する機能であり、このアクセス･ログについて Web 2.0 時代の要求に合わせて登場してきている新たな技術を調査・検討した。

キーワード　データベース, 監査, Web, 役割の分離, コンプライアンス, アクセス記録, 3 階層システム

The Issues and Solutions for Database Audit Logging in Web-based Systems

Yutaka MATSUNAGA† 　Michiko OBA‡

† Computer Network Division, Tokyo Electron Ltd.　5-3-6 Akasaka, Minato-ku, Tokyo, 107-8481 Japan
‡ Hitachi Ltd. Software Division, Emerging Business Development
5030 Totsuka-cho, Totsuka-ku, Yokohama-shi, Kanagawa 244-8555, Japan
E-mail: †matsu@kabuki.tel.co.jp, ‡michiko.oba.cq@hitachi.com

Abstract　Unauthorized accesses including information theft continue to be reported, but application structures are becoming more and more complex which makes the administrators difficult to apply the countermeasures. As a more essential way of mitigation, database protection is increasing its importance. Meanwhile, monitoring and audit of data access required for internal control in the legacy ways have demonstrated critical unsolved problems. For both cases, the functionality to monitor and record the accesses must be the core of the solution. We examind the access log technologies arising to meet the Web 2.0 systems.

Keyword　database, audit, Web, segregation of duties, compliance, access log, three tier system

1. はじめに

Web 技術を利用したシステム構築が一般的になり、さらに Web 2.0 と呼ばれる動向の中で Ajax を初めとする新しい技術が次々と使われ始めている。このためアプリケーション構造とネットワーク通信は複雑さを増し、性能や信頼性に対する要求と共に、セキュリティに対する要求も増大している。特に情報漏洩の事件が後を絶たず、データの不正アクセス防止に対する重要性が高まっている。

従来、こういったセキュリティの機能はファイアウォールなどのネットワーク機器と、アプリケーション開発時の対策でカバーしてきたが、事故の報告は一向に絶えないことから、他の部分での対策も求められている。その中でも大量のデータが構造化されて保存されているデータベース・システムへのアクセスを監査及び保護する技術の強化が必要になっている。

こういったデータベースに対する監査及びセキュリティの要求は、組織あるいはシステムの外部からの要因と内部からの要因の両方によって増大している。

外部からの要因としては、インターネットを通じた不正行為がエスカレートしており、ネットワーク・レベルのセキュリティ技術では防衛の限界を迎えていることがある。よりアプリケーションやデータに依存したセキュリティ対策を施す必要があり、データベースの保護が有効である。

内部からの要因としては内部関係者による不正行為が問題になっており、比較的手薄だった内部ネットワークへの攻撃や正規の権限の濫用がポイントとなる。さらにこうした不正行為の防止も含めて、各種規制への準拠とその対応のための内部統制の実現が急務となっている。その中でもデータの保護、特にデータベースのセキュリティは欠くことのできない要素となっている。

こうした要因を受けて、データベースのように重要なデータを保存しているシステム要素に対しては、データの利用が不正でないか常時チェックし（監視）、何か問題があった場合の原因調査や定期的な検査を行い（監査）、さらに不正な利用を遮断する（保護）仕組みが求められている。以上 3 つの機能、すなわち監視、監査、保護を実装するときに中核となるのがアクセスの内容を監視し、記録する機能である。ところが今までのデータベースに対するアクセス・ログの仕組みは必ずしも充分ではなかった。

本稿ではまず、データベースのセキュリティ要求を高めている外部及び内部からの要因を具体的に検証した上で、従来のアクセス監査ログ技術における機能的な問題点とその解決に求められる要件を洗い出す。そして浮かび上がった要件を満足する為の実現技術を提案する。

2. データベース監査・セキュリティ機能への要求

データベースの監査及びセキュリティに対する要求は、システムの外部から、内部から、及び規制準拠に対する要因から起こっている。

2.1. 外部からの攻撃による要求

インターネットからの脅威は従来、Web アプリケーションを通じた比較的単純な手段が使われる事が多く、防御はアプリケーション開発時の入力文字列検査など、基本的な対策を実施していればよかった。アプリケーションでの対策漏れや、対策回避の為の複雑な攻撃も、アプリケーション･ファイアウォールと言われるような、高機能なセキュリティ機器を Web サーバに適用する事で､多くを排除する事ができていた。

しかし新しく出てくるアプリケーション技術を悪用したり、まったく新しい侵入経路を利用したりする攻撃技術の進化で、Web レベルでの防御が難しくなっている。

アプリケーション技術としては、Web 2.0 と呼ばれる動向の中でより柔軟なユーザ･インターフェースが提供されるようになっており、それを悪用する攻撃が現れている。

2005 年 10 月にソーシャル･ネットワーキング･サービス内で使用された Samy による Ajax ワームは、Web 2.0 アプリケーションで代表的に使われている技術 Ajax を攻撃の手段として利用しただけでなく、従来のセキュリティ対策を無効にする工夫が多岐にわたって施されていた[1]。

新たな侵入経路としては、Web を経由しないでデータベースを直接攻撃したり、内部ネットワークの資源（端末など）を踏み台にしたりすることにより、従来のセキュリティ対策を迂回する事が行われている。イントラネットのサーバに対するインターネットからの侵入方法さえ報告されている。JavaScript で書かれた攻撃コードを Web アクセスやメール経由で内部端末で実行させることにより、その端末を介してイントラネット･サーバへの攻撃が可能になる(図 1) [2]。

こうした、外部からの攻撃手法の多様化は、「入り口で止める」タイプの対策が効果を失い始めていることを意味する。この傾向に対処する為には、よりデータベースに近いところでデータの動きを正確に把握することが重要となる。

さらに、インターネットからの侵入行為が次第に大きな金銭的利益を目的にし始めていることも、問題を大きくしている。例えばオンライン証券会社を標的とした一連の不正行為が報告されており､ある会社の被害額が 1800 万ドルを超えたという報告があった他、複数の会社が被害に遭っている[3]。これらの事件においては、攻撃の方法は明らかになっておらず、更なる被害の拡大が懸念されている。この事件についてセキュリティ分野の権威として知られる Richard Stiennon 氏は、口座の挙動を適切に監視していれば不正行為が完了する前に検知できたかもしれない、と述べている[4]。こういった監視のためにはデータベースのアクセス・ログは不可欠になる。

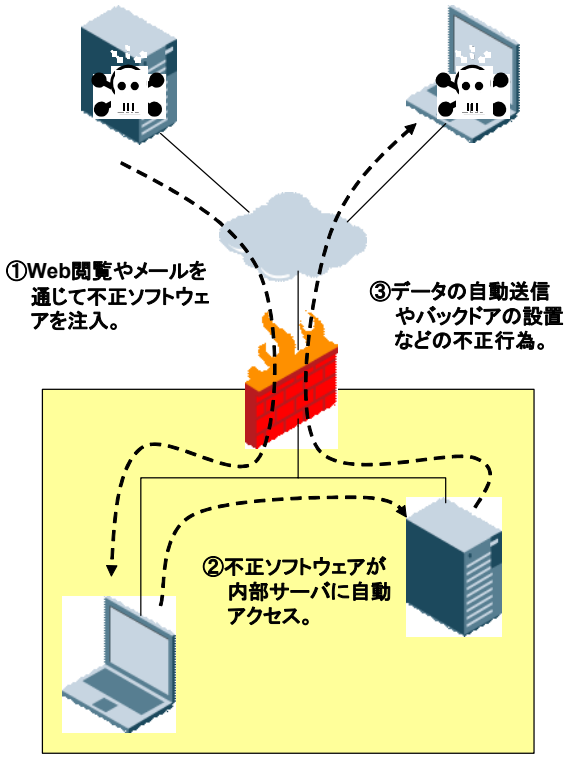

図 1 新たな侵入経路 - インターネットから イントラネット･サーバの攻撃

Fig.1 A new route of attack – attacking an intranet server from outside.

2.2. 内部要因から起こる要求

内部関係者による不正行為も次第に深刻になってきている。

2006 年 6 月には、国内電話会社の顧客データベースから 400 万件の個人情報漏洩が発覚した。この事件では、個人情報は内部で開発者が使用していた PC を使ってダウンロードされたとされている[5]。この事件では、正規のアクセス権限を持った内部関係者がその権限を濫用した可能性があり、こうした不正は部外者を締め出すことを目的とした従来のセキュリティ技術では防止できない。

内部システムではこうした権限の濫用の他に、システム設計として内部に対しては厳しいセキュリティ対策を施してない場合があること、内部関係者はシステムの弱点を情報として入手しやすいこと、などの問題がある。

2.3. 規制準拠の為の要求

さまざまな問題を受けて法規制や業界規制も厳格化の方向にあり、その中でもデータ保護の為の施策は多く求められている。

多くの企業が遵守を要求される Sarbanes-Oxley、COBIT、HIPAA、Visa の CISP といった規制に対する監査では､データベース･セキュリティの実装が厳しく求められている。具体的にはデータの参照や変更の可否を詳細に管理する必要がある[6]。このためには詳細なアクセス制御の技術と同時に、詳細なポリシーの策定手段や実際のアクセス状況のポリシー遵守状況確認手段が必要になる。

クレジットカード業界の標準規格である Payment Card Industry Data Security Standard (PCI DSS)では、カード保有者の情報について、情報種別ごとに具体的な保護技術を指定して導入を求めており、クレジットカードを取り扱う業者が違反すると罰金などの罰則がある。また、全てのシステム要素のログを最低でも日次で検査するように求めている[7]。従って、データベースに関しては特定の情報に対するアクセス状況を検査できなければいけない。

3. データベース･セキュリティの問題

高度化する一方であるデータベースの監査や保護に関する要求に対して、従来監査ツールなどで提供されてきたセキュリティ技術には問題点が多く指摘されており、システム構築の障害となっている。

3.1. ユーザ名の特定･記録

監視・監査・保護のすべてにおいて、特定のアクセスが誰によって行われているかという情報が基礎となる。ところが Web 技術を利用した 3 階層構造のアプリケーションでは、データ･アクセスのユーザ名を特定するだけでも大きな問題が存在する。

アプリケーションはしばしばコネクション･プーリングという技法を使い、データベースへの継続的な接続を 1 本（あるいは少数）作成して､ユーザからの問合せは全てこれを使用する。この仕組みは、要求があるたびに接続を作成する事による性能面での問題を解決する。ただしこの際、接続は複数ユーザで共有され、アプリケーションで決められたユーザ名を利用する。

このやり方はアプリケーションの動作には問題ないが、アクセス制御や監査にとっては大きな障害となる場合がある。これは、多くのデータベース管理システム(DBMS)付属のアクセ

ス制御機能や監査機能あるいはデータベース監査ツールの機能が、SQL で接続したときのユーザ名しか記録していないためである(図 2)。

従って、記録は中間のアプリケーションのユーザ名しか残されず、ユーザごとの権限によるアクセス許可も実施不可能となる。

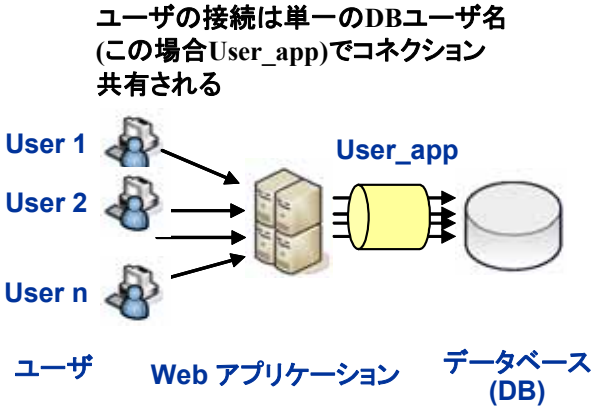

図 2 コネクション共有によるユーザ名の隠蔽

Fig.2 The connection pooling alters the usernames

3.2. 詳細なアクセス制御

仮にユーザ名が判明した場合も、詳細なアクセス制御の実装は困難な場合がある。アクセス制御を行う為には、個々のユーザについて、データベース内のデータ･オブジェクトごと､場合によっては列ごとに参照、更新などの権限設定を行う必要がある。これ無しには特定のアクセス要求が正しいかどうかの判断ができない。しかしユーザ数やデータベースの規模・複雑さによってはこの権限設定作業が莫大な工数を要求することになり、更に運用中のユーザ登録やデータ構造の変化に伴う権限設定の保守･管理は非現実的な場合がある。

このため、現実のデータベース･システムでは、詳細なアクセス制御は行われていないことも多い。

3.3. セキュリティ機能の迂回

監査やアクセス制御の仕組みが運用されているシステムにおいて、悪意のある攻撃者が仕組みを迂回する場合がある。大きく分けて、管理者権限の濫用による場合と、システムの脆弱性を利用する場合がある。

DBMS 付属の監査ログ機能を使っている場合、データベース管理者はその機能を一時停止したり、記録を改竄したりすることができる。またデータベース･サーバ上にインストールする形の監査ツールを利用している場合には、サーバの管理権限を持った管理者は、監査ツールの動作やログに対しても変更する権限を持つ事になる。こういった管理者が悪意を持った場合、セキュリティ機能の効果は保証されない。

システムの脆弱性については、サーバや DBMS を乗っ取ることでセキュリティ機能を迂回できる場合がある。例えば 2006 年 1 月に公表された Oracle データベースの脆弱性は TNS という通信プロトコルに内在し、ネットワーク中の通信内容を改ざんすることにより、DBMS の管理者権限を奪取できる[8]。この際、管理者権限で行われる操作は全て組み込み監査機能を迂回し、記録を残さない事が可能になる。

3.4. ログデータの管理

監査用のログデータは膨大な量になる事がある。保存やバックアップなどのシステム上のデータ管理はさまざまな技術が提供されており、費用を考えなければ最大規模のデータ容量も扱えるようになっている。しかしそのデータの利用については、テラバイト単位の情報を検索しなければいけない場合もあり、問題となる。

この点は法規制への遵守にも影響を及ぼす場合がある。例としては Sarbanes-Oxley 法の 409 条において、財政あるいは運営上の重要な変更を迅速に報告しなければいけないことが規定されている。この際、時系列による傾向を提示することや､情報開示は 48 時間以内に実施しなければいけないことが規定されている。

こういった期限を伴うログ分析を行うには、ログデータは単に保存されているだけでなく、高速な閲覧と検索が可能になっている必要がある。

3.5. 既存のデータベース監査ツールの種類と評価

既存のデータベース監査ツールには、3 種類の形態が存在する。

1. DBMS ネイティブの監査機能 – DBMS 自身が提供する機能。外部ツールでこの機能による監査ログを管理するタイプのものもある。
2. ホスト・ベースの監査ツール – データベース･サーバに専用のソフトウェアを導入して監視する。
3. ネットワーク型 – ネットワーク上で通

信を監視する

この中で、1.は DBMS に付属している機能を使うので導入負荷が最も低い。ただしデータベース・サーバの処理負荷が増大する点と、データベース管理者が監査の管理権限まで併せ持ってしまう点が問題。2.は 1.の問題点を解決する為にサード・パーティから提供される形態であるが、サーバの管理者からの権限分離は難しい点と、DBMS など関連するソフトウェアの互換性が問題になる場合があり、運用は簡単ではない。3.についてはデータベース・サーバから完全に分離できる為、権限の分離と処理負荷面を解決できるが、コンソールからのアクセス監査を何らかの方法で補助しないといけない場合がある点と、ツール導入の初期費用がかさみがちな点に考慮が必要となる。

表 1 監査ツールの形態による特徴
Table.1 Different types of audit tools

	1 DBMS 付属機能	2 ホスト・ベース	3 ネットワーク
権限の分離	できない	できない	できる
サーバへの負荷	高	中	無し
DBMS 互換性	問題なし	要注意	問題なし
導入コスト	低	高	高
運用コスト	低	高	低

4. データベース･セキュリティに求められる要件

今まで見てきたような問題点を解決し、有効なセキュリティ･システムを構築する為には､次のような要件を満たす必要がある。

4.1. Web ユーザの認識と追跡

監査ログとアクセス制御を有効にする為に、Web 技術を使った 3 階層構造のシステムでは、Web で接続しているユーザを識別し、記録する必要がある。さらにデータベースへの個々の操作についてこれら Web ユーザとの関連を突き止める方法を持たなければいけない。

4.2. 権限設定の自動化

アクセス制御に伴う問題点(3.2)を解決する為に、個々のユーザの権限設定を現実的な作業工数に収めるには、可能な限り自動化する必要がある。自動化の手段は、セキュリティ機能の構築時における初期設定作業と運用中の各種変更に対応するための調整作業の両方に対して有効でなければいけない。

この自動化が実現すれば、不正なアクセスを検知して管理者に通知したり、未然に遮断したりする機能も実用的になる。

4.3. 職務の分離 (Segregation of Duties)

サーバや DBMS の管理者による監査･アクセス制御の迂回を防止するためには、こうしたセキュリティ機能の管理を、サーバや DBMS の管理者と明確に分離する必要がある。つまり、こうしたセキュリティ機能は、サーバや DBMS の管理者の関与なしに導入、設定、運用できなければいけない。具体的には、

- 監査業務とデータベース管理者が分離していること
- 監査データの収集機能とデータベース･ソフトウェアが分離していること
- 独立した監査ツールを使用する際にも、DBMS やサーバにまったく依存せずにアクセス記録が行えること

の 3 点が必要になる[9]。

4.4. プラットフォームの保護

サーバや DBMS の脆弱性を通じてセキュリティ機能が迂回されることのないよう、プラットフォームを保護する事が必要になる。

大きく分けて、脆弱性の有無を検査する機能と脆弱性を悪用する通信の即時検知がある。脆弱性有無の検査については、脆弱性スキャンと呼ばれる走査を定期的に行う。即時検知については、侵入防御システム(IPS）と呼ばれる技術でネットワーク通信を監視する。

4.5. ログデータの高速検索

監査用に取得したアクセス･ログに対して、高速に閲覧･検索できる手段が提供されなければいけない。これには、データ容量を最小限に抑える（圧縮する）機能と、莫大なレコード数のログに対して検索処理を高速に行える専用ロジックが必要になる。

5. システム設計の提案

前章で明らかになった要件を満たす為には、適切な技術を使って最適なシステム設計を行う必要がある。ここでは、データ保護とコンプライアンスを満たすシステムを、現在入手可能な技術を使って構成する為のシステム設計を提案

する。

5.1. サーバ及び DBMS からの分離

職務の分離を実現する為には、監査機能は DBMS 及びそれが稼動するサーバから物理的・論理的に分離している必要がある。従って、監査機能はネットワーク上に独立して存在し、DBMS への通信を全て傍受分析可能で、導入や設定に際して DBMS に依存する機能を持たないことが条件となる。

5.2. Web 通信の監視と相関

Web ユーザを認識する為には、Web サーバに対する通信を常時監視し、Web への http リクエスト個々について、ユーザ名を判別できる機能が必要となる。このためには、

- ログイン通信の認識機能
- Web トラフィックにおけるセッションの追跡機能

の 2 点の機能をもつ監視ツールを導入する。

さらにデータベースへのクエリに対するユーザ名を特定するには、Web 通信とデータベースの監査ログ情報を相関分析し、結び付けられる機能を導入する(図 3)。

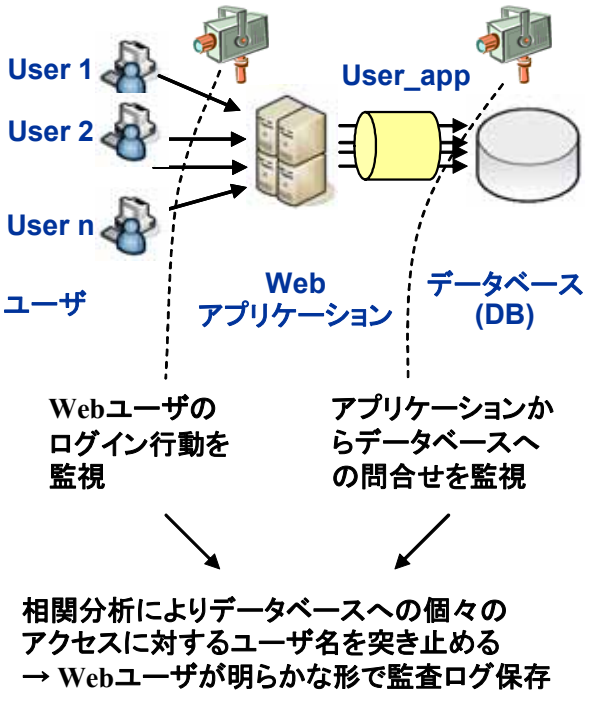

図 3 相関分析によるユーザ名の取得
Fig.3 Obtaining the username by correlated analisys

この他にも Web ユーザを認識する方法として、アプリケーションにユーザ名伝達の機能を埋め込む方法や、アプリケーション・サーバのミドルウェアが持つユーザ名伝達機能を利用する方法もある。ただしこれらはアプリケーションの書き換えを必要とする上に、ミドルウェアなど基盤の種類に依存する場合が多いので、変更が少なく長期間固定して利用するシステムに向く[10]。

既にそれぞれの技術を利用したツールが利用できるようになっている。

5.3. IPS (侵入防御システム)

プラットフォームの保護のために、IPS を導入する。この際、サーバ OS 及び DBMS への攻撃に関する検知機能を備え、攻撃パターンが随時更新されることが重要である。

5.4. 脆弱性スキャナ

プラットフォームの保護のために、脆弱性スキャナを導入する。この際、サーバ OS と DBMS 両方の脆弱性を検査でき、脆弱性リストが随時更新される必要がある。

5.5. プロファイリングによる権限ベースライン

権限設定を自動化するには、個々のユーザが許されるデータアクセスの内容を自動的に計測する。これには、運用中の通信を監視してその内容を統計化していくプロファイリングが有効である。一定期間の学習を行う事によって、権限設定のベースラインを作成し、基本的な権限設定として利用できる。

運用中の各種変更に対する権限設定の調整も、同様の技術で自動化することができる。監査機能の運用中に常時プロファイリングを行う事により、ユーザの増減やデータ内容の変化、あるいはユーザのアクセス行動の変化などを自動的に検知できる。セキュリティ管理者は、プロファイリング機能が報告してくるこうした変化を検査し、不正な性格の変化でない事を確認すれば良い。

こうして詳細なアクセス権限設定が運用可能な工数で実現できる事によって、ポリシーに違反するような行動や、報告義務のある重大な変化などの異常を自動的に検知することができる。

5.6. ログ管理システム

ある程度大規模なシステムのアクセス・ログを管理・利用する為には、ログ管理に特化したツールを使う必要がある。ログデータの性質を利用した圧縮技術を提供し、ログ検索にチューニングされた検索ロジックを持つツールを導入する。こうした機能を提供するログ管理専用のツ

ールが存在する。

5.7. 性能、スケーラビリティと一元管理

以上のような各種セキュリティ機能は、システムの業務に悪影響を与えない形で実装する必要がある。また、複数のデータベースが稼動するような大規模なシステムにおいては、タイムリーな管理判断と監査要求に応える迅速な分析機能を提供する為に、一元的な管理機能が求められる。

これらの要求を満たすには、

- システムの処理を遅延させないだけの分析性能
- ボトルネックを生じさせないための分散配置
- 一箇所から全ての管理･分析が行える一元管理機能

の 3 点を併せ持つ形でセキュリティ･システムを設計する。

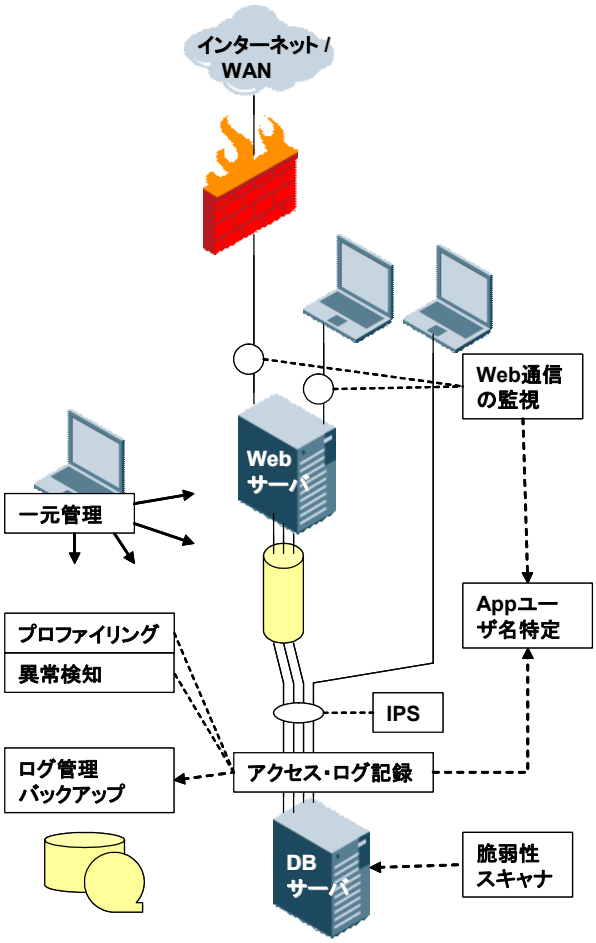

図 4 データベース・セキュリティの全体像
Fig.4 The overview of the proposed database security

5.8. セキュリティ･システムの全体像

今回提案した監視･監査･保護の機能を提供する､監査ログを中心としたセキュリティ･システムの全体構造を図 4に示す。

6. まとめ

インターネットとイントラネットいずれにおいても標準的になっている Web システムの中で、セキュリティや内部統制の面で重要度が増しているデータベースの監査とセキュリティについて、その中核となるアクセス・ログを中心に、求められている要件と新たに登場してきている技術を調査した。

従来の仕組みでは、ユーザ名の特定、アクセス制御、機能迂回の防止、ログデータの管理など、問題点が多くあったが、新たに登場している技術を利用することで、ほとんどが解決できることが分かった。

データベース･サーバとは独立した形で必要な機能を実装していくことで、情報漏洩対策や内部統制の実現に必要な要件を満たした監査ログ取得が行えるようになってきている。

ただしシステム構築の際には、必要な監査およびセキュリティの要件を理解し、関連技術から必要なものを選択し正しく実装することが今までにもまして必要である。

文　献

[1] 松永豊および大場みち子, Web 技術を悪用する攻撃に対するサーバ側セキュリティ要件, 情報処理学会研究報告, 2006-DD-056, Vol.2006 No.83 (2006)

[2] Jeremiah Grossman, Hacking Intranet Websites from the Outside, Black Hat Japan Briefings, October 2006

[3] Ellen Nakashima, Hackers Zero In on Online Stock Accounts, Washington Post, October 24, 2006; Page A01

[4] Richard Stiennon, New pump and dump scheme, Threat Chaos, August 31, 2006 http://blogs.zdnet.com/threatchaos/index.php?p=396

[5] 榊原　康,【KDDI情報漏えい続報】「アクセス･ログは 1 年間しか保存していなかった」, IT Pro, 2006/06/13 http://itpro.nikkeibp.co.jp/article/NEWS/20060613/240788/

[6] Phebe Waterfield, Security Begins at the Database Level, Yankee Group DecisionNote Market Analysis, October 2005

[7] PCI Security Standard Council, Payment Card Industry (PCI) Data Security Standard Version 1.1, September 2006

[8] Imperva Application Defense Center, Security Advisory: Oracle DBMS – Critical Access Control Bypass in Login Bug, January 2006 http://www.imperva.com/application_defense_center/papers/oracle-dbms-01172006.html

[9] Imperva Inc., What Auditors Want - Database Auditing 5 Key Questions Auditors Ask During a Database Compliance Audit, Imperva Whitepaper, September 2006

[10] 渡辺 亨靖, DB 監査ログの”穴”埋める製品相次ぐ, 日経コンピュータ, 2006/9/4