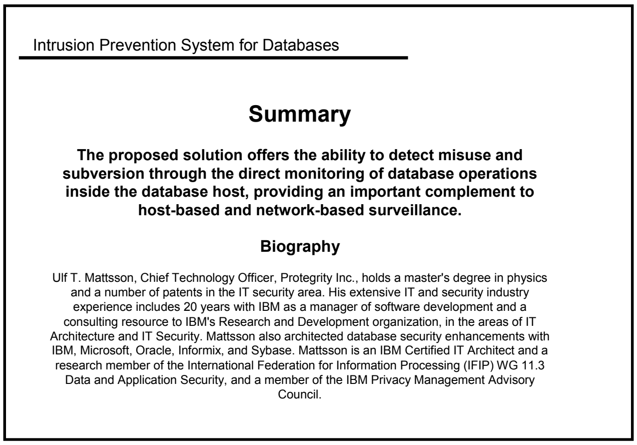
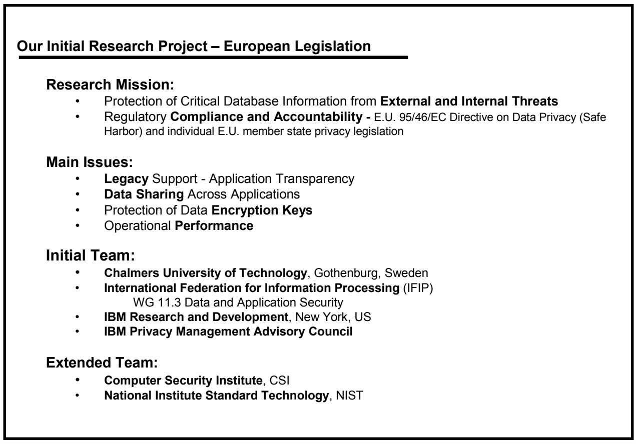

Intrusion Prevention System for Databases

Summary

The proposed solution offers the ability to detect misuse and subversion through the direct monitoring of database operations inside the database host, providing an important complement to host-based and network-based surveillance.

Biography

Ulf T. Mattsson, Chief Technology Officer, Protegrity Inc., holds a master's degree in physics and a number of patents in the IT security area. His extensive IT and security industry experience includes 20 years with IBM as a manager of software development and a consulting resource to IBM's Research and Development organization, in the areas of IT Architecture and IT Security. Mattsson also architected database security enhancements with IBM, Microsoft, Oracle, Informix, and Sybase. Mattsson is an IBM Certified IT Architect and a research member of the International Federation for Information Processing (IFIP) WG 11.3 Data and Application Security, and a member of the IBM Privacy Management Advisory Council.

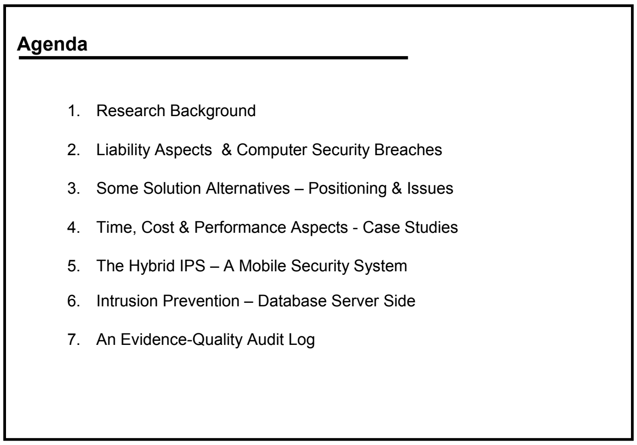

Our Initial Research Project – European Legislation

Research Mission:
- Protection of Critical Database Information from **External and Internal Threats**
- Regulatory **Compliance and Accountability -** E.U. 95/46/EC Directive on Data Privacy (Safe Harbor) and individual E.U. member state privacy legislation

Main Issues:
- **Legacy** Support - Application Transparency
- **Data Sharing** Across Applications
- Protection of Data **Encryption Keys**
- Operational **Performance**

Initial Team:
- **Chalmers University of Technology**, Gothenburg, Sweden
- **International Federation for Information Processing** (IFIP)
  WG 11.3 Data and Application Security
- **IBM Research and Development**, New York, US
- **IBM Privacy Management Advisory Council**

Extended Team:
- **Computer Security Institute**, CSI
- **National Institute Standard Technology**, NIST


Agenda

1. Research Background
2. Liability Aspects & Computer Security Breaches
3. Some Solution Alternatives – Positioning & Issues
4. Time, Cost & Performance Aspects - Case Studies
5. The Hybrid IPS – A Mobile Security System
6. Intrusion Prevention – Database Server Side
7. An Evidence-Quality Audit Log

Project Requirements: Privacy Legislation & Industry Initiatives

Privacy Legislation:

- U.S. Gramm-Leach-Bliley Act, (GLBA) extended with the U.S. Office of the Comptroller of Currency (OCC)

requirements for the financial services industry

- U.S. Healthcare Insurance Portability and Accountability Act (HIPAA)
- U.S. Food & Drug Administration (FDA) 21CFR 11 Electronic Records; Electronic Signatures for Clinical Trials
- U.S. State of California SB 1386 Disclosure Law
- E.U. 95/46/EC Directive on Data Privacy (Safe Harbor) and individual E.U. member state privacy legislation
- Canada's Personal Information Protection and Electronic Document Act (PIPEDA)

Industry Initiatives:

- ISO 17799 Code of Practice for Security Management
- American Express Merchant Data Security Standards
- MasterCard Site Data Protection Service
- VISA Cardholder Information Security Program (CISP)
- VISA 3D Secure specifications for cardholder data protection
- U.S. Software and Information Industry Association (SIIA) - A method for securing credit card and private consumer data in e-business sites

Typical Compliance Requirements:

User Access Control & Audit

Data Integrity

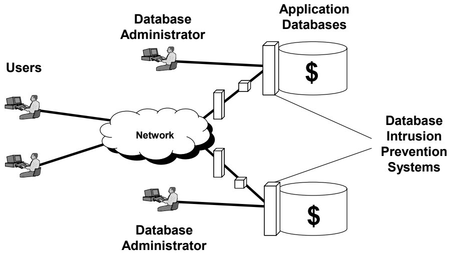
Administrator Access Control & Audit

Response when unauthorized access is suspected or detected

Data Confidentiality

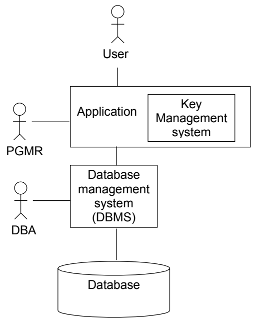
The Database Intrusion Prevention System

The proposed solution locks down the database to both enforce correct behavior and block abnormal behavior. The default policy ensures rapid deployment.



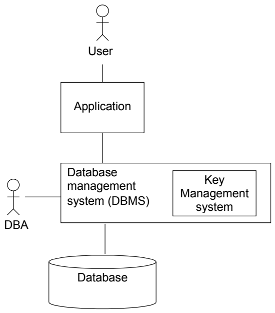
Case Studies - 4 Server Solution Alternatives

Encryption Keys exposed in the application environment.



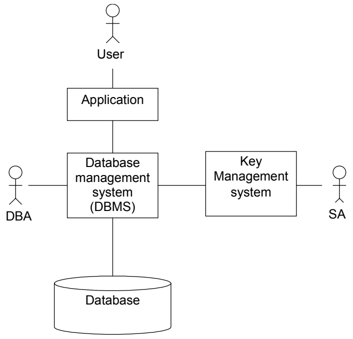
Case Studies - 4 Server Solution Alternatives

Encryption Keys exposed in the database environment.

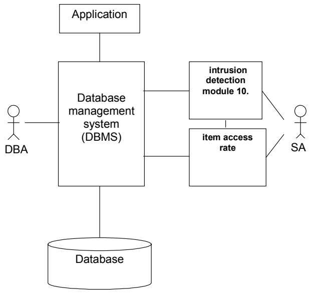


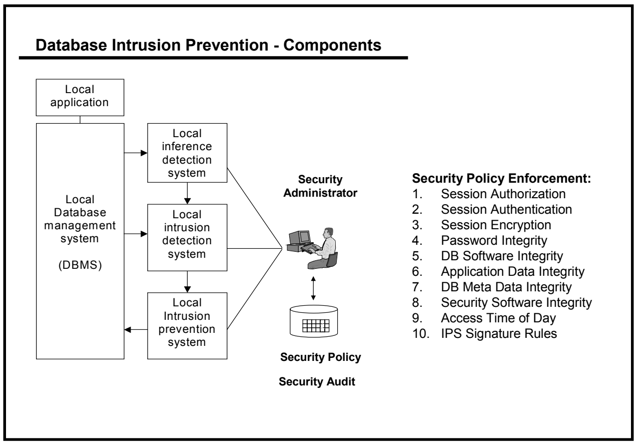
Case Studies - 4 Server Solution Alternatives

Encryption Keys managed securely separate from the database environment



Case Studies - Solution Alternatives


Application
Database management system (DBMS)
intrusion detection module 10.
item access rate
DBA
SA
Database



Database Intrusion Prevention - Components

Local application

Local Database management system

(DBMS)

Local inference detection system

Local intrusion detection system

Local Intrusion prevention system

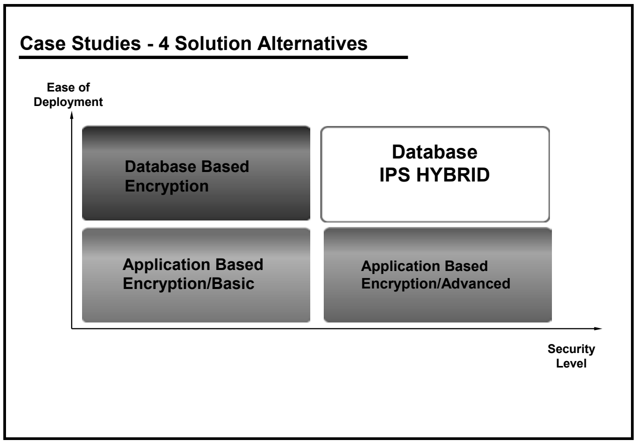
Security Administrator

Security Policy

Security Audit

Security Policy Enforcement:

1. Session Authorization
2. Session Authentication
3. Session Encryption
4. Password Integrity
5. DB Software Integrity
6. Application Data Integrity
7. DB Meta Data Integrity
8. Security Software Integrity
9. Access Time of Day
10. IPS Signature Rules



Case Studies - 4 Solution Alternatives

Ease of Deployment

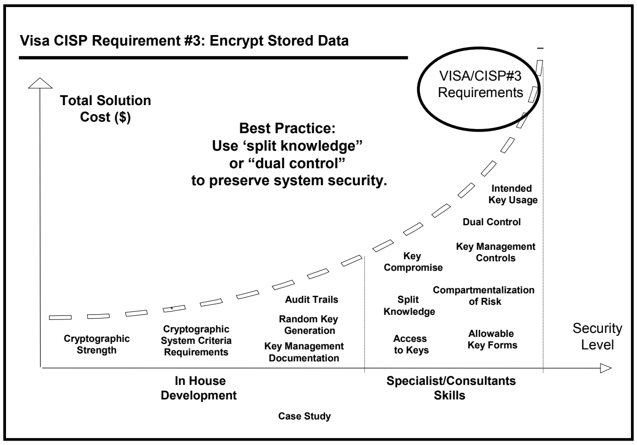
Database Based Encryption

Database IPS HYBRID

Application Based Encryption/Basic

Application Based Encryption/Advanced

Security Level



Visa CISP Requirement #3: Encrypt Stored Data

Total Solution Cost ($)

VISA/CISP#3 Requirements

Best Practice:
Use 'split knowledge"
or "dual control"
to preserve system security.

Intended Key Usage

Dual Control

Key Management Controls

Key Compromise

Compartmentalization of Risk

Audit Trails

Split Knowledge

Random Key Generation

Cryptographic Strength

Cryptographic System Criteria Requirements

Key Management Documentation

Access to Keys

Allowable Key Forms

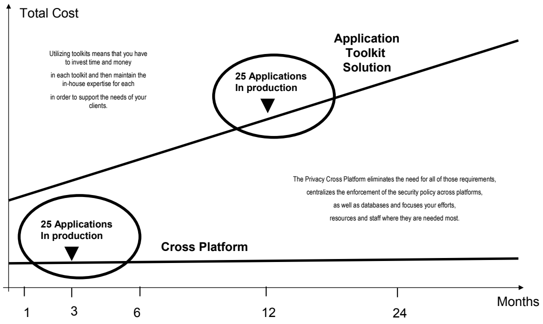
Security Level

In House Development

Specialist/Consultants Skills

Case Study

Implementation Time: 25 Applications Visa Compliant



Utilizing toolkits means that you have to invest time and money in each toolkit and then maintain the in-house expertise for each in order to support the needs of your clients.

The Privacy Cross Platform eliminates the need for all of those requirements, centralizes the enforcement of the security policy across platforms, as well as databases and focuses your efforts, resources and staff where they are needed most.

Case Study

Visa/CISP#3 – Case Study – Development

Training, analysis, design, programming, test, documentation, and installation:

- **Application Integration Development: 4 man-weeks/application**
- **Cryptographic Solution Development (man weeks):**

Cryptographic Vector Functions:	2
Key Management Control Functions:	12
Access to Keys Isolation:	11
Random Key Generation:	2
Allowable Key Forms Functions :	9
Intended Key Usage Functions :	10
Key Compromise Prevention Functions	10
Dual Control Functions :	6
Split Knowledge Functions :	8
Compartmentalization Functions:	10
Secure Audit System:	11

Case Study

Sensitive Information

What are Protegrity's clients protecting?

- The Investment Banker: While allowing each broker access to the corporate database, Secure.Data restricts permissions to the non-public personal information of clients belonging to other associates not required to view such sensitive data.
- The Communications Services Provider: Billing is charged to client credit cards on a monthly basis. Secure.Data was implemented to enforce the separation of duties between database administrators and the Accounts Payable department, by only allowing access to credit card information in Finance.

Sensitive Information

What are Protegrity's clients protecting?

- The Telecom: Adhering to the Telecom Act of 1996 by protecting client data through selective encryption.
- The Computer Software & Services Provider: Our client is using Secure.Data along with their Human Resources application to prevent salary information from being disclosed within any area other than HR.
- The Food and Beverage Company: In the soft drink space, providing access to sensitive formula information must be strictly controlled. Protegrity's Secure.Data protects this mission critical asset from both internal and external threats.

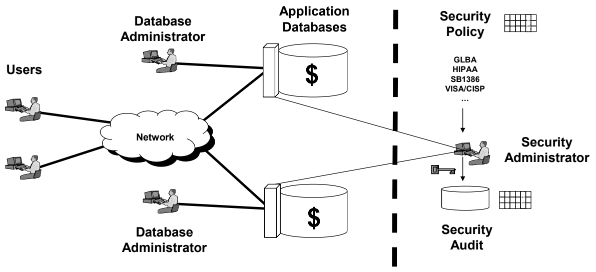
Sensitive Information

What are Protegrity's clients protecting?

- Human Services: As a solutions provider to state social services agencies, our client is required by law to protect the confidentiality and integrity of client data.
- Pharmaceutical: The research arm of one of our clients uses Secure.Data to protect the identities of chronically ill patients suffering from a deadly disease.
- Transportation: Our client in the railroad industry protects details regarding the cargo manifest and the shipping schedule. Especially today, protecting this information is a primary security concern.

Best Practice (Visa USA) – Dual Control

Use 'split knowledge" or "dual control" to preserve system security.



Case Studies - Solution Alternatives

Network-Based Detection - Network intrusion monitors are attached to a packet-filtering router or packet sniffer to detect suspicious behavior on a network as they occur. They look for signs that a network is being investigated for attack with a port scanner, that users are falling victim to known traps like .url or .lnk, or that the network is actually under an attack such as through SYN flooding or unauthorized attempts to gain root access (among other types of attacks). Based on user specifications, these monitors can then record the session and alert the administrator or, in some cases, reset the connection. Some examples of such tools include Cisco's NetRanger and ISS' RealSecure as well as some public domain products like Klaxon that focus on a narrower set of attacks.

Server-Based Detection - These tools analyze log, configuration and data files from individual servers as attacks occur, typically by placing some type of agent on the server and having the agent report to a central console. Some examples of these tools include Axent's OmniGuard Intrusion Detection (ITA), Security Dynamic's Kane Security Monitor and Centrax's eNTrax as well as some public domain tools that perform a much narrower set of functions like Tripwire which checks data integrity. Tripwire will detect any modifications made to operating systems or user files and send alerts to ISS' RealSecure product. Real-Secure will then conduct another set of security checks to monitor and combat any intrusions.

Case Studies - Solution Alternatives

Security Query and Reporting Tools - These tools query NOS logs and other related logs for security events or they glean logs for security trend data. Accordingly, they do not operate in real-time and rely on users asking the right questions of the right systems. A typical query might be how many failed authentication attempts have we had on these NT servers in the past two weeks." A few of them (e.g., SecurIT) perform firewall log analysis. Some examples of such tools include Bindview's EMS/NOSadmin and Enterprise Console, SecureIT's SecureVIEW and Security Dynamic's Kane Security Analyst.

Inference detection - A variation of conventional intrusion detection is detection of specific patterns of information access, deemed to signify that an intrusion is taking place, even though the user is authorized to access the information. A method for such inference detection, i.e. a pattern oriented intrusion detection, is disclosed in US patent 5278901 to Shieh et al. None of these solutions are however entirely satisfactory. The primary drawback is that they all concentrate on already effected queries, providing at best an information that an attack has occurred.

GLBA/OCC IT Requirements

1. **Access control** and authentication
2. **Encryption, including transit and storing**
3. Implementation to confirm modifications consistent with InfoSecPol
4. **Segregation of duties for access control management**
5. **Mechanism to protect the security by service provider**
6. **Monitoring system to detect actual attempted attacks**
7. Response when **unauthorized access is suspected or detected**
8. Response to **preserve integrity and security**

OCC Data Security Regulations II.A-B; III.A-D for GLBA

HIPAA IT Requirements

1. **Data to be Protected - “patient identifiable information”, not necessarily medical records**
2. **Healthcare is Data Driven & Data Intensive**
3. **Shorthand for security requirements:**
   - Confidentiality
   - Integrity
   - Individual Accountability
4. **Current Interpretation is Data at Rest as well as Data during Transmission**
5. **Protegrity provides trusted functionality (access control, integrity, confidentiality, audit trails) as required by HIPAA and as needed by business requirements**
6. **Protegrity provides the means for this functionality across several applications and platforms**

Visa USA CISP Requirements

1. Install and maintain a working network firewall to protect data accessible via the Internet
2. Keep security patches up-to-date

ISSUE

3. **Encrypt stored data**
4. Encrypt data sent across open networks
5. Use regularly update anti-virus software
6. **Restrict access to data by business “need to know”**
7. Assign unique ID to each person with computer access to data.
8. Don’t use vendor-supplied defaults for system passwords and other security parameters
9. **Track access to data by unique ID**
10. Regularly test security systems and processes
11. **Maintain a policy that addresses information security for employees and contractors**
12. Restrict physical access to cardholder information

***Best Practice:* Use ‘split knowledge” or “dual control” to preserve system security.**

Liability Issues executives need to consider

1. Class and individual action suits
2. Loss of network/database integrity and availability
3. Loss of intellectual capital
4. Loss of employee productivity
5. Defamation of brand name and reputation

Liability Coverage: Computer Security Insurance

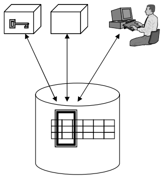
Customers utilizing the
Database Intrusion Prevention Technology
for data-privacy will qualify for up to a
40% discount on breach of
computer security insurance coverage.

Placed with Lloyd's of London, this policy provides the insured broad first party e-business Prevention for highly secure risks. Coverage includes Prevention against losses resulting from computer hacking, illegitimate use of computer systems and other Information Technology security risks.

INSUREtrust, Marsh McLennan, …
, …

Case Study: Application Encryption – Advanced

Applications



Source Code Changes

Prevention of Encryption Keys?

No Data Sharing with Application Packages, Database Utilities and Report Generators …

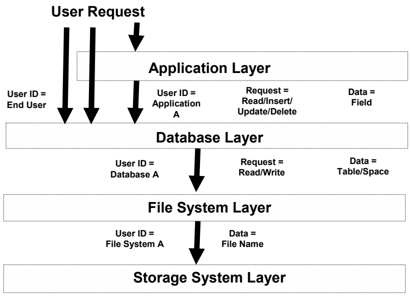
No Search on Encrypted Data

No JOIN on Encrypted Data

No Stored Procedures

Applications stop working …

Solution Layers – Information Request Granularity



User Request

Application Layer

User ID = End User

User ID = Application A

Request = Read/Insert/ Update/Delete

Data = Field

Database Layer

User ID = Database A

Request = Read/Write

Data = Table/Space

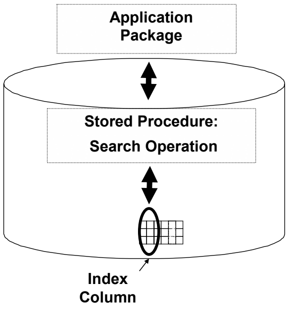
File System Layer

User ID = File System A

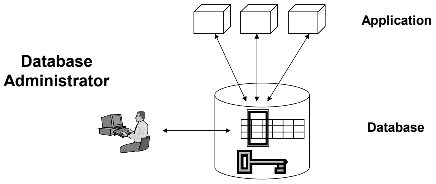
Data = File Name

Storage System Layer

Case Study – Issues with Application Level Encryption



Case Study: Database Encryption – Advanced



**Do NOT leave
'The Keys to The Store'
in the Database!**

Questions with Database Encryption

1. Is there there a concept of access control with Read, write, update, delete as separate functions, or will a user either has **100% access or 0%?**
2. Are **keys are stored in in clear text** for the duration of the session. This is readily accessible to any DBA! No point in locking the data if the key is accessible!
3. Is key storage password protected (requires second authentication), In on OS file (**unsecured from root**), or in the database in clear text (**accessible by the DBA**)? None of these are secure solutions.
4. Are keys generated by a **random number generator in the OS?** Not secure.
5. Is there a key recovery system? If you delete all the current users (private key and the associated copy of the "data" key) of a column will you have destroyed the keys and now have **unrecoverable data**?
6. Is there a **secure audit** around sensitive data or changes to access policy? Is there a central control of access, or can any defined user change access to the tables they own.
7. Is a private key required for key protection? Must the key be supplied to access data? This infers that **application changes** must be made to handle the key management. FIPS 140 level 3 support?
8. Is there **support for encrypted indexes acceleration?**
9. Is there **wizard support for automated deployment and migration of data and database definitions?**
10. Is there only **limited support of data types**, (or only Varchar2, raw or numeric (without parameters) are supported)?
11. Is the product **supporting all major database brands?**
12. Is the product **supported by major database vendors?**
13. Is the product **supported by major security vendors?**
14. Can I talk to multiple **reference customers in my industry segment?**

Case Study: Database Encryption – Advanced

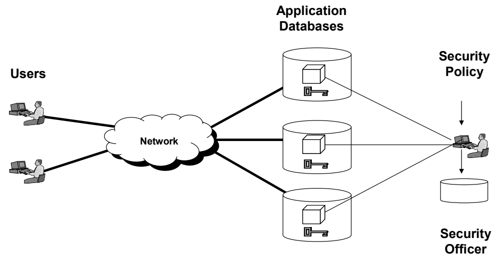
	The FAQ Scorecard (High Score is Most Favorable)	Hybrid Encryption	Database Encryption
Deployment	Do I need to change my applications?	100	0
	Support for several major database brands?	100	0
	Support for all major data types?	100	0
	Support for encrypted index?	100	0
Security	Are encryption keys protected exposure in clear text?	100	0
	Support for recovery of encryption keys?	100	0
	Support for random generation of encryption keys?	100	0
	Support for separation of users and encryption keys?	100	0
	Insert/update/delete/select support in security policy?	100	0
Audit	Audit support for all access to data?	100	0
	Audit support for all changes to security policy?	100	0

High Score is Most Favorable

Check Point UAA Integration Details

- **User requests secured application - A client attempts to access an application which is secured by a VPN-1 or FireWall-1 gateway and requires authentication.**
- **Gateway authenticates user, establishes VPN - Based on the security policy, the gateway authenticates the user.**
- **In this example, the user is requesting a connection through a VPN-1 Gateway and the policy specifies that a VPN be formed between the client and the Gateway.**
- **Application asks UserAuthority for user information - The application receives the connection request from the user. A user profile must be configured prior to a login request succeeding.**
- **Because this application leverages the UserAuthority API, it is a UserAuthority Client capable of making requests to the UserAuthority Server located at the Gateway.**
- **In this example, the UserAuthority Server knows about the user, so it responds to the application's UserAuthority Client request.**
- **A UserAuthority Server can also query other UserAuthority Servers, creating a chain of requests, until the UserAuthority Server which knows about the user is found and responds.**
- **Application makes intelligent authorization decision Based on information UserAuthority supplied. In this release the Secure.Server is able to make an intelligent authorization decision based on the authentication method supplied.**
- **Additional requests - Additional requests by this user to other applications do not require the user to authenticate. Rather, the UserAuthority-enabled application they want to connect to can make an inquiry to a UserAuthority Server.**

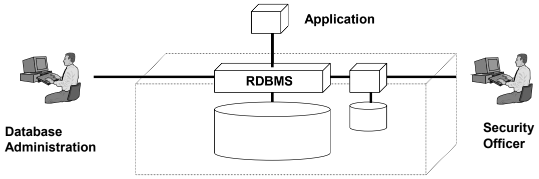
A Database Intrusion Prevention Solution



The Hybrid - Much more than data encryption

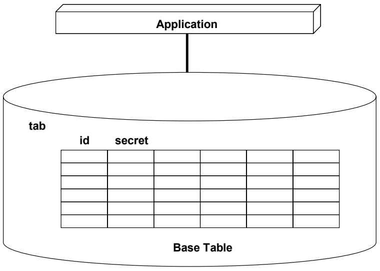
- The Database Intrusion Prevention provides an effective last line of defense
  1. Selective and highly secure, column-level data item encryption
  2. Cryptographically enforced authorization
  3. Comprehensive key management
  4. Secure audit and reporting facility
  5. Enforced separation of duties
  6. Interoperability with other security technologies
  7. Operational transparency to applications

Separation of Privacy Control Duties

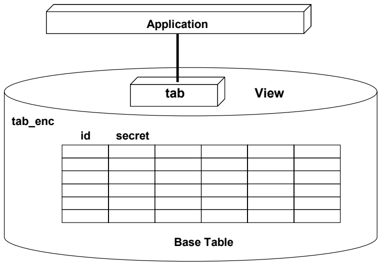


1. **Separation of duties for encryption key management**
2. **Separation of duties for integrity check of selected software executables**
3. **Separation of duties for access control policy**
4. **Strong authentication for the security administrator**

Secure.Data – Implementation - Sample



Secure.Data – Implementation - Sample



Secure.Data – Implementation – Tables & Views

The original base table ‘tab’ holds an identity ‘id’ column and a secret code column ‘secret’:

Create the new base table ‘tab_enc’ is defined as:

```
create table tab_enc (
                    id integer,
                    secret varchar (32) for bit data);
```

Create the new base table ‘tab_enc’ thet will hold encrypted values in the ‘secret’ column:

```
create table tab_enc (
                    id integer,
                    secret varchar (32) for bit data);
```

Create a view with the same name as the original base table ‘tab’:

```
create or replace view tab(id, secret) as
      SELECT id, decrypt('tab_enc.secret', secret)
      FROM tab_enc;
```

Secure.Data – Implementation - Triggers

Protegrity SQLdirector creates a trigger on the view ‘tab’ to be able to insert data:

```
create or replace trigger tab_insert
instead of insert on tab
for each row
begin
            insert into tab (
                        id,
                        secret)
                        values (
                                    :new.id,
                                    pty.ins_encrypt(‘secret’, :new.secret));
 end;
```

Protegrity SQLdirector creates a trigger on the view ‘tab’ to be able to update data:

```
create or replace trigger tab_update
instead of update on tab
for each row
begin
            update tab set
            id = :new.id,
            secret = pty.upd_encrypt(‘secret’, :new.secret))
            where id = :old.id;
 end;
```

Protegrity SQLdirector creates a trigger on the view ‘tab’ to be able to delete data:

```
create or replace trigger tab_delete
instead of delete on tab
for each row
begin
            pty.del_check(‘secret’);
            delete tab
            where id = :old.id;
 end;
```