Global Information Assurance Certification Paper

Implementation challenges for an SSO aware Enterprise Data Warehouse

Elias Serrano
Security Essentials Certification (GSEC) version 1.4b
Option 1
August 22, 2003

ABSTRACT

Large enterprises evolve over time, diversifying their products and services, expanding overseas, etc. New corporate ventures create the need to develop or acquire new technologies, which not always necessarily integrate with the rest of the existing enterprise systems. Real or perceived silos develop, which make it difficult to come up with coordinated solutions in order to decrease costs, improve productivity and time to market, and increase customer satisfaction. However, sooner or later, the trend from diversification in the enterprise systems inevitably reverts to integration. Data, Process and Presentation integration becomes a necessity form an economic point of view. This leads to consolidation of processing centers, databases, and multiple instances of local applications.

In this paper, I attempt to identify some of the challenges in consolidating scattered data, coming from disparate systems (NT, UNIX, AS/400, Mainframes, etc.) into enterprise data warehouses for business intelligence analysis, as well as the security risks involved in providing secure web access to the data. Since large enterprises usually end up with multiple warehouses, logically separated by the company's different lines of business, I include a discussion on why an interactive Single Sign On (SSO) is desired in order to improve the customer experience.

INTRODUCTION

The main criteria behind the consolidation of data centers for global companies are, as with almost everything else in the modern business world, economics. Once a company expands internationally, it faces multiple problems, which in time start eroding its competitive advantage. Applications become more complex, or even worse, current applications are modified and deployed in different regions, increasing the cost of maintenance. Since different local groups usually manage these applications, the size and cost of the IT staff increases.

One of the preferred methods for company expansion in this day and age seems to be acquisitions. This and by itself creates its own set of problems. For example, organizational boundaries create silos, which have the potential of decreasing the cooperation between geographical business units, creating a de facto ineffective organization, where the quality of the service the end user

receives suffers. Management of budgets between many different IT organizations and data centers becomes a nightmare. Since the data is distributed among multiple locations, which not necessarily share the same technologies, global MIS customers cannot receive the information they need, in a timely manner, in order to increase their own competitive advantage.

In the process of consolidating international data and still follow the principles of confidentiality, integrity and availability, one important thing to consider is local regulations. For instance, the European Union (EU) created a privacy law back in 1998 that "prohibits the transfer of personal data to non-European nations that do not meet the European adequacy standard for privacy protection"[1]. In this case, all US companies must abide by a "Safe Harbor" policy before any data can be transferred out of the EU.

Benefits of Data Consolidation

Cost reductions when consolidating data in single repositories can be achieved by a number of means, including:

- Server consolidation. One example could be the implementation of VMware. "When deployed in software development and test environments and in corporate IT operations, VMware server products consolidate applications and infrastructure services running on diverse operating systems onto fewer highly scalable, reliable enterprise-class physical systems"[2].
- Improved quality service for global customers. Instead of receiving multiple data feeds from regions all over the world, or having to logon to multiple sites all over the world to view their own data, customers receive a single, global picture of their own clients, allowing them to perceive consumption patterns, and negotiate more favorable contracts with their vendors (e.g.: better airline fares).
- Reduced cost in IT expenses. Even with today's trend towards labor globalization, where technologies development and maintenance is transferred offshore, minimizing the number of duplicate operations has a positive effect in a company's overall results.

Creating the Warehouse.

Even after consolidating the data from multiple regions, it is usually processed by legacy systems, which were built over time to carry out the different aspects in the business of the organization. These systems were built somewhat like islands, independent of each other, except for regular batch feeds between some of them. Security of the data has always been important, but it was not as involved as it is today, particularly when the data is exposed to web access. Most of the exposure of the data to manipulation was limited to internal employees, using dumb terminals, applying online or batch updates, or processing external feeds, regularly received via tape or using a transmission vehicle (FTP, XCOM,

NDM, etc.), into secure receiving areas. The organization of the file systems was directed to quick online transaction processing, or for the use of queries already optimized for the individual application. Since the systems were created over time, they used several different DBMS, relational or otherwise, different programming languages, operating systems, as well as a collection of platforms for multiple vendors.

Accessing legacy data to provide ad-hoc query access and data mining is sometimes difficult. Legacy databases are not organized for easy ad-hoc query access, and as a result most queries not only run extremely slow, but they may interfere with the normal response time required for transaction processing of the legacy systems, many of them being mission critical. Also, the data is spread across many different platforms, possibly in different formats, so creating efficient reports required by the organization involves too many resources from the MIS department. In response to this situation, data warehouses were created.

A data warehouse is usually a large, de-normalized database containing data transferred from other legacy systems in the organization, and is the foundation for information analysis across the enterprise. Before the data is applied to the warehouse, it is cleaned and integrated with other data coming from different applications. Extract, Transform and Load (ETL) tools maybe used for this purpose, and selecting one is not an easy task. “Companies that are developing their own data warehouses often struggle to determine the best methodology and implementation strategy for performing ETL”[3]. The data is massaged in a way that makes it easier, or at least possible, to provide data mining and reporting capabilities, either directly through SQL created by internal users, or more commonly through the use of specially designed tools. These Decision Support System (DSS) tools provide a logical view of the database, processing logic, and business rules, normally stored in a special database (metadata). This allows the average user, or executive users, to perform sophisticated analysis of the organization’s data without the need of knowing a programming language.

Creating a data warehouse, or a set of data warehouses organized by line of business, is a very desirable goal for big enterprises, but it poses a number of challenges, including: warehouse modeling, population, and selection of an appropriate user interface to facilitate forecasting, simulation, analytical processing, data mining and analysis of trends.

In the process of building a data warehouse, it is important to remember the cornerstones of data security: Confidentiality, Integrity and Availability. Based on the fact that warehouses are read only databases (from the end user’s point of view), and that usually warehouse test environments do not replicate the production environment (number of processors, amount of hard drive available to load the entire production warehouse for stress test purposes), it is a common temptation to point development DSS tools to the production warehouse.

Creation of a test warehouse with scrubbed production data, and pointing development DDS tools to it is imperative in order to achieve the security objectives.

Warehouse Modeling.

The start point in the creation of the data warehouse is the analysis of the various data sources and processes in the organization, identifying the key elements and the relationships between them. The output of this exercise is the enterprise data model, containing a single enterprise schema. Since the multiple enterprise data sources have been developed using standards varying over time, or due to time constraints not much attention was paid to existing standards, the data sources usually contain different definitions for comparable entities, including varying field sizes, names, and so forth. In addition, similar fields may contain different codes meaning the same thing (e.g. “AA” in one application identifies American Airlines, and “001” in a separate application also means American Airlines). All of these issues are resolved during warehouse population.

The next step in warehouse modeling is the grouping of the enterprise entities in Facts, Dimensions and Attributes. These groups constitute the basis for the creation of the actual physical tables. Facts are numerical values that represent performance measures of a business activity, e.g. prices, profits, sales, etc. Dimensions are characteristics of the facts, which describe the context of those facts. By the way, one of the most popular dimensions, associated to most facts, is the Time dimension. Finally, Attributes are used to classify facts, and can be arranged in hierarchies and used to analyze data at different aggregation levels, e.g. Country, Region, State, City).

While legacy systems are usually normalized, a data warehouse must be highly de-normalized to minimize the number of necessary joins for efficient query processing. In the real world, some degree of normalization is necessary to facilitate data manipulation through DSS tools. Summary tables can help performance of specific queries and reports. There are a number of schemas, which are appropriate for data warehouse modeling, including Star and Snowflake, as well as variations between them.

In a Star schema, fact tables are at the center of a number of dimension tables. The dimension tables contain the attributes used in the decision making process. For example, if the Region code were included in the fact table, one of the dimensions would include the Region Code and Description. Once the logical model is created and used by the DSS tool, the end user can manipulate related facts by looking at Region Descriptions instead of codes. Snowflake schema is a variation of Star schema, where dimensions are further normalized into “snowflakes”. Independent of the schema being used, special consideration must be paid to table relationships, especially in cases where there’s a many to many

relationship between fact and dimension tables. This may lead to duplicate rows in the results sets, and incorrect results in report totals.

Warehouse Population

Input from legacy systems is usually not ready for loading into the warehouse. Individual entities must be aligned so the data loaded into the warehouse is uniform, and data must be transformed to conform to the warehouse model. There are a number of factors to consider when loading a warehouse:

- In-house vs. external ETL tools: Extract, Transform and Loading tools support the process of consolidating, massaging and loading data from disparate systems into the warehouse. A good ETL tool can reduce dramatically the amount of time and money required for maintenance of the warehouse loading process.
- Load Periodicity: When building a warehouse that will only support monthly MIS reporting and data mining after a specified period closes, it certainly does not make much sense to include it in a hub-and-spoke architecture where warehouse updates are almost instantaneous. However, if the warehouse will support day-to-day analysis where the source data changes everyday, nightly updates may be appropriate. In some cases it will be necessary to provide for updates not tied to specific intervals. Data purges should be included within the normal warehouse load window; so online query performance is not impacted.
- Loading time: Loading a warehouse can be a very intensive process. When a warehouse is designed for ad-hoc reporting, numerous indices are created over the physical tables in order to improve performance of the queries. Having to maintain many indices on the fly slows down the load process, with the possibility that users will need to access the warehouse before the load process ends, and this has a great impact on their own online queries. There are ways to deal with this situation, including: deletion of indices before the load process starts and recreation once it ends, use of Symmetric Multiprocessing (SMP), etc.
- Synchronization: Loading the warehouse may depend on other processes successfully ending on some or all of the legacy systems used as a source of the warehouse data.[4]

DSS tools

DSS tools, or business intelligence tools, are user interfaces that connect to the data warehouse, in order to provide decision makers (internal or external) with a more or less user friendly way to improve their tactical and strategic decisions. Typically, a DSS tool will have a logical view of the warehouse, standard reports and templates, all stored in the metadata. Metadata is simply

"data about data", and it may include other objects, like server definitions, user profiles, etc.

Some DDS tools include means of achieving data confidentiality, by including filters designed to restrict the customer's view of the warehouse. For example, users may be restricted to look at data from a specific region, group of stores, a single business unit, etc.[5]

Providing web access to the data.

Data warehouses contain information that is critical for the competitive advantage and success of an enterprise. Even when the data is exclusively accessed by internal users and used for internal purposes only, it confronts a number of security risks from malicious internal access, server breakdowns, data corruption, natural disasters, etc. Enabling access to the data from the web raises enormous security issues for the IT personnel. E-business can only succeed when defense in depth is practiced, securing every layer of the access chain (web servers, application servers, network, database servers) from external attack, while at the same time providing secure access to authorized users. Recent events (MS blaster worm) teach us that threats come not only from unauthorized personnel trying to get access to the company's data, but from unknown hackers wreaking havoc on the computer resources, for no particular or declared reason against the enterprise. The threat is always out there, and good planning and preparation help minimize the risk.[6]

Web access to the warehouse requires a delicate balance between providing the right access to authorized users, and restricting access to the rest. The challenge here is to identify what information will be available to individual users, or groups of users, and who of those users will have access to individual warehouse tables or views, as well as the operations the user is allowed to perform on each warehouse object. In order to secure the integrity and availability of the warehouse, no internal or external users should have constant access to the database server tables outside of the authorized tools and applications. In other words, users and developers should not be able to logon to the warehouse box and use SQL to view or modify the tables, or use other access methods for the same purpose. Some shops Policies and procedures must be in place to grant access in case of pre-determined and accepted exceptions. All involved parties must sign confidentiality agreements.

Data level security restricts access to individual objects in the warehouse, and it is applied based on a list that includes all warehouse objects, the users having access to the object, and the type of access allowed to the user (read, change, etc.) Usually, only read access should be granted to the warehouse tables and views, and change access for temporary tables in a separate space. Application level security provides secure access to slices of information in the

warehouse, and it is usually bundled within the DSS tools. The use of secure communications, e.g. SSL, is necessary to protect confidentiality of the data.

User authentication and authorization must include both data level and application level security. When a user logs on the tool, the user id and password are authenticated against information stored in the tool's metadata. Once the user is authenticated, the tool displays only those objects where the user has been previously authorized (reports, templates, folders, etc.) When the user runs a report, the tool applies rules and filters that restrict the user to slices of the warehouse data (customer's data, regions, markets, etc.), and uses these rules to create and submit to the warehouse SQL statements that include those restrictions, achieving application level security. When the warehouse is contacted to execute the SQL from the tool, the id and password used by the tool to connect to the warehouse platform (which is also stored in the tool's metadata) is authenticated, and used to provide the right access to the tables included in the SQL statements. This achieves data level security.

A good architecture that provides secure web access to the warehouse should include all kinds of protection against attacks: firewalls, intrusion detection systems at the network and application levels, and anti-virus software in each box (with up-to-date virus definitions database). In addition, the architecture should include three tiers for external access to the data:

- First tier: Web Server, which contains the pages, used by to tool to interact with the users, and is protected by a firewall against external attack. The web server should not have any connections defined to any other boxes and the outside world, with the exception of the Application Server.
- Second tier: Application Server, where the intelligent part of the tool is installed. This tier may connect to other servers to store its metadata, file caches, etc.
- Third tier: Database Server, where the warehouse resides.

There should be a firewall between each tier. Also, clustering should be used when available, at least at the web and application tiers, in order to provide failover capability, which increases the site's availability.

Single Sign On (SSO)

The difference between a large enterprise with multiple secure sites with SSO, and one without SSO, is somewhat like the difference between a free country and a very closed dictatorship (we do not have a lot of those left). In a free country, you are authenticated at the port of entrance by showing a valid passport. Once you step inside the country, you are authorized to go just about anywhere you chose to. In a non-free country, you would have again to provide authentication at multiple times and places after successfully crossing the border.

SSO provides authentication and access control, allowing the user to connect to multiple secured sites within the enterprise (or even across different enterprises) by logging on once, even when every one of those sites may have different passwords for the same user. The system is normally deployed inside the firewall connecting to the Internet. Once the user successfully logs on to the SSO portal, authorization is granted to all of the sites stored in the user's SSO profile, even across different platform, applications, or domains. Sometimes, password synchronizations is confused with true SSO. In password synchronization, a main password is distributed and synchronized across multiple systems and domains, giving the appearance and the user experience of true SSO, which is intrinsically more secure.[7]

A typical implementation of SSO (e.g. SiteMinder) includes a central policy server, usually located in a separate server or cluster of servers, and contains information about the users and the sites where the users are authorized. The policy server provides:

- Policy management. Maintaining sets of access rules to specific resources, and associating these rules to users or groups of users.
- Authentication, based on the user id and password, or a variety of methods, including certificates, hard or soft tokens, etc.
- Authorization. Providing access to pre-defined URLs, and applying the access control rules.

Web agents are installed on the web servers, intercepting all requests to the web server and managing access to the server contents, according to the security policies.[8]

Conclusion

- Consolidating data and operations is a highly desirable goal for large enterprises, in order to reduce application complexity, cost, and time to market.
- Global data warehouses provide the means to analyze enterprise data by line of business. However, modeling and loading a warehouse is not an easy task.
- Every aspect of data security must be taken in consideration when creating an enterprise data warehouse, and particularly when exposing this data to the web, in order to minimize the risk to internal and external attacks.
- Providing a single sign on mechanism improves the user experience, and this is especially true for large enterprises with multiple secure sites.

References

[1]U.S Department of Commerce. “Safe Harbor Overview”. URL:http://www.export.gov/safeharbor/sh_overview.html (Aug 5, 2003)

[2]Vmware Products. “Server Consolidation” URL:http://www.vmware.com/solutions/consolidation.html (Aug 5, 2003)

[3]Jennings,Michael F.”Strategies for Custom Data warehouse ETL Processing.”DM Review, June 2001. URL: http://www.dmreview.com/editorial/dmreview/print_action.cfm?EdiD=3573. (Aug 18, 2003)

[4]Baldwin,Dirk.”Data Warehouse Notes” September 1,1998.URL: http://www.uwp.edu/academic/mis/Baldwin/warehous.htm (Aug 18,2003)

[5]Power,D.J. “A Brief History of Decision Support Systems” URL: http://dssresources.com/history/dsshistory.html (Aug 19,2003)

[6]SANS Security Essentials 1.2.1 “Threat and the need for Defense in Depth” (Aug 19,2003)

[7]Taylor,Laura.”Understanding Single Sign On”May 28,2002 URL: http://www.intranetjournal.com/articles/200205/se_05_28_02a.html (Aug 20,2003)

[8]Netegrity”SiteMinder features/Benefits” URL: http://www.netegrity.com/products/products.cfm?page=SMfeatures (Aug 20,2003)