Global Information Assurance Certification Paper

Single Sign-on: Deployment Considerations

By:
Daryn Holloway
GSEC Version 1.3

Formatted

Summary:

As the size and scope of enterprises continue to grow, the pressure to maintain security across platforms and locations increases. The most common form of security is password protection. The result is a proliferation of passwords that users must remember to do their jobs effectively. Security and convenience become at odds with each other.

Bridging the security vs. convenience issues are Single Sign-on (SSO) products which offer administrators the ability to have tighter access controls and users to have one login that grants them access to all the resources they need. This paper discusses some of the issues surrounding an SSO deployment.

Scenario:

Pat's walking to the office through the customer service phone banks. Pat's eye was caught by a proliferation of brightly colored sticky notes taped to the sides of monitors. Moving in for a closer look, Pat discovers that the sticky notes are all user names and passwords to various mission critical severs and applications in the enterprise. Horrified, Pat rushes to the Director of Customer Service's office and demands that the sticky notes be removed as they represent a security risk.

With a blank unapologetic stare, she tells Pat she can't realistically do that. The number of passwords her employees have to remember to do their jobs is excessive and she cannot have them wasting time trying to remember them or calling the Help Desk. Each time there is a password change, productivity in her department takes a dip. She has graphs and time studies to prove this.

Within a matter of days, argument about security needs vs. productivity needs has climbed the ladder to the upper echelons of management where it rages fast and furious.

Sound familiar?

If it doesn't, it will.

The Problem:

As the organization has grown the complexity of its network has also increased. The result is that, in an effort to maintain network security, employees have several different user ID's and passwords that they must remember in order to effectively do their jobs. "Numbers from Stamford, Conn. -based Gartner indicate that these costs can reach $300 per user per year -- which probably is closer to reality than one might think, when user productivity costs are included in the calculation." (1)

In a homogeneous network, where there is only one login required, password policies are easily dealt with by end users. The moment these requirements are spanned across platforms and applications, security measures will be thwarted by a blizzard of sticky notes. Suffering from time crunches and productivity requirements of their own, users will always seek a workaround to security measures if they are too inconvenient or poorly understood.

Administrators are tasked to keep track of users and their permissions in order to enforce security policies as laid out by the organization. These administrative efforts most likely cross platforms, departments, as well as geographic locations.

What is difficult for the individual end user to deal with on a day -to-day basis can be debilitating to administrators and helpdesk personnel as they are forced to deal with access problems for the entire enterprise . Some analysts estimate that more than half of all helpdesk calls are related to forgotten passwords . Maintaining a secure hold on a network distributed across platforms and locations is a daunting task. Most likely there are multiple groups within an organization that are responsible for the administration and security of various severs and/or applications throughout the network(s) . The distribution of administration duties may be departmental or geographical. Changes affecting the nature of the business, its locations and the technologies employed by the organization increase the difficulty of maintaining constant user rights. Employees, partners, suppliers, etc. come and go , and the transient nature of these relationships adds another level to the difficulty to tracking and removing outgoing users IDs and access to systems across the enterprise. Given a system with tens of thousands of users, a practical mechanism must exist to rapidly assign and manage user access.

The solution:

Understanding the nature of end users and the requirements of security, the only realistic solution for large, distributed networks, is single -sign-on (SSO). "As a whole they all provide some form of Authentication, Authorization, Access control and password synchronization. SSO solutions are available for both organizations moving towards e -commerce as well as enterprise networked environments." (2)

A heterogeneous implementation of Microsoft Windows 2000 Active Directory can be considered a single-sign-on in its most limited form. Assuming all applications deployed are also Microsoft products, single sign-on is achieved for client/server applications and IIS based web applications. On the other end of the spectrum are internet wide solutions like AOL's Magic Carpet or Microsoft's Passport which work only with web based applications. These Internet-wide solutions, especially Passport, have had a number of security breaches within the last year, which severely limit their usefulness in a corporate environment. Another drawback is that their management and security is in the hands of another entity. As a result, for most businesses, internet-wide SSOs are not a viable solution yet.

At the enterprise level, SSOs divide themselves into two categories. There are those like Oblix NetPoint that use cookie based authentication schemes designed for web based applications. For organizations with client/server based applications, products like Netegrity's SiteMinder, Computer Associates' eTrust and RSA's Single Sign-on SDK are available

An SSO allows users to authenticate once to a central server, which then negotiates connections to authorized systems, hosts and applications for that user. SSO products integrate with leading directory services standards such as LDAP (*Lightweight Directory Access Protocol)* products like iPlanet Directory Sever or Windows 2000 Active Directory. ODBC (*Open Database Connectivity)* directory services connections like Oracle and SQL are also commonly supported. Integration with these directory services enables authentication credentials to be stored in one database.

Because most SSOs are more focused on web-based applications, their functionality is geared toward accessing resources from the internet/intranet. . With more and more information and services available to employees, customers, suppliers and business partners, it becomes increasingly important to organize and tailor that information to limit access and provide a mechanism for convenient data consumption.

Administrators can use SSO tools to implement security policies to protect resources including, but not limited to, web applications and website content. Administrators are able to develop and assign authentication schemes in which they can define and manage privileges for groups of users to specific resources.

Given the size of many enterprises, flexible delegated management is essential. Several administrative roles can be defined that allow for localized management of the security environment. The schema administrator can delegate authority based on security domains established within the SSO structure. This allows first and/or second tier administrators to continue controlling access to their department or group's resources without having any control over the resources of another department. All changes implemented within each security domain are

stored centrally and subject to controls set by the schema administrator thereby maintaining a cohesive access scheme across the enterprise.

SSOs are capable of resolving some of the productivity issues and security issues at the same time by using existing user data sources as the basis for an enterprise-wide security policy that can be as far reaching as individual web pages.

Customization to accommodate the SSO will be necessary regardless of the size of the organization. This is especially true if the organization is heavily reliant on client/server applications. If the SSO has not been completely planned for and correctly implemented, the deployment can cause serious headaches, as well as large and unplanned expenditures and productivity loss.

Regardless of the SSO chosen, to ensure a good deployment, there are four areas that must be focused upon; security, people, processes, and technology.

Security:

One of the two main reasons for investing in an SSO is the increased control of enterprise security. Because a successful SSO implementation requires the cooperation of diverse groups, including management, developers and administrators, security planning and implementation requirements are often given short shrift. However, to fully realize the benefits of an SSO deployment, security planning and analysis cannot be ignored. Some level of development will be necessary regardless of the SSO chosen and the resources being protected. By addressing security concerns up front, the organization can avoid recoding delays, cost overruns, and in some cases a deployment bereft of anything but basic authentication.

Challenges:

Foremost among the concerns of security minded administrators is that a single ID and reusable password represent the "keys to the kingdom" providing a hacker access to all the resources associated with that ID. Deploying an SSO increases the need for robust, secure authentication: fully encrypted logon at a minimum, and/or the use of one-time passwords, challenge response systems or biometric authentication.

There are two other concerns with central authentication service products. The central authentication server becomes a single point of attack and a single point of failure. If the central authentication server goes down, via a denial of service attack or for non-malicious reasons, there must be a secondary source of authentication. Redundant or mirrored servers should be in place and for the

most critical of applications alternate means of direct login should be available if the central authentication server fails.

Extra measures have to be taken to ensure that these serv ers are protected against any attack or misuse. Physical and network access to these servers must be as limited as possible. Where basic authentication is employed, all passwords, both the original logon password entered by the user , and any passwords stored in the central directory server for logon to target applications, should be encrypted in the server while stored and during transmission across the network.

Benefits:

Maintaining a security focus enables organizations to rea p the full benefits of SSO technology.

SSOs support a number of authentication schemes. Developers can select the authentication scheme appropriate to a specific application. Because it is easier to develop security into an ap plication than it is to recode to address security needs after the fact, it is important to consider the options before development begins. Commonly supported authentication schemes include the following:

- *Basic* – Identifies users based on a username and password. Most SSOs also support basic authentication over SSL.
- *X509 Client certificates* - Identifies users by verifying the users digital certificate. Certificate authentication can be combined with basic authentication for two-tier authentication.
- *HTML Forms* – Identifies a user with customized HTML forms that collect the user's credentials. Other information can also be collected if the form is built to support it.
- *Tokens* – identifies the user with hardware tokens that provide unique passwords. These passwords are generated by the token and changed frequently.
- *Proxy* – the SSO server authenticates users as a substitute for a third party proxy server.

- *Digest* – identifies users by compari ng an encrypted user attribute string stored in a server's directory against an encrypted string entered by the user.

- *Anonymous* – identifies non-registered users and assigns them a Global User ID that gives them access to specified resources.

Because basic authentication is still the most common authentication method robust password formats and password change requirements should be incorporated from the onset of planning. To make this effective the following must be taken into account:

- Users must have the option to change their passwords because users who create personal passwords are less likely to write passwords down.
- Passwords should require a minimum number of alphanumeric letters, numbers, punctuation, and non -printable characters to be valid.
- A minimum/maximum password length should be established. Users will be more likely to select a password that is more difficult for hackers to predict.
- Passwords should be protected against reuse. By forcing the use of different passwords, it becomes more d ifficult for unauthorized users to predict an SSO password.
- Most SSOs allow the establishment of a percentage difference from a previous password to make passwords more difficult to predict effectively preventing users from repeatedly using similar passwor ds.
- Passwords should have a limited number of attempts before the users account is disabled and users are forced to call the Help Desk for assistance. This limits the effectiveness of password cracking tools and alerts administrators to possible tamperi ng if account lockouts are consistently occurring.
- Passwords must have expiration dates that force a password change before the user can log on.

A robust password policy is not enough. Encryption of both passwords stored on the central authentication ser ver(s) and encryption in transit must be employed. Only by protecting passwords and their transmission, can a baseline level of security can be established for the SSO architecture.

People:

The roles and responsibilities of people within an organization are widely varied based on job function and responsibility levels. It is important to understand what

the job functions are of people throughout the organization so that appropriate levels of access can be granted.

Gathering information from administrators about current levels of access to resources is not enough to build a successful responsibility matrix. Information from other sources such as department managers all the way down to the people working the frontlines is also necessary, as their roles may have changed since the last time access concerns were addressed.

Role-based privileges allow an administrator to define classes of users and to add, modify, and delete resource privileges based on these classes. For example, roles can be established for groups of users with similar functional access profiles such as procurement users, administrative users, or logistics users. By using these groupings, administrators are able to develop security profiles for users that have similar security requirements.

Challenges:

The challenge for implementation is understanding the various roles of personnel and extrapolating those duties into roles that the SSO can administer.

While roles are already administered, it is likely that there are several sources for administration. Consistent permissions across a large network with distributed administration are unlikely. It is also unlikely that the administrators have a clear understanding of the functions performed by the various people with access to their systems. It is therefore necessary to consult with people outside the IT realm to fully grasp the roles of departments, groups and individuals.

Benefits:

With a clear understanding of the people and departments inside the organization basic, enterprise level roles can be established. These basic roles should be able to be effectively used across applications and departments within the organization. These simple roles form the foundation for the access control scheme. Additional layers of control can then be applied to further limit access to network resources.

Processes:

SSO has the capability to not only grant access rights based on roles but also on the flow of information. Wide, broadly based, groups can be further narrowed in scope to effectively restrict user access to resources specific to their actual role within the organization. These permissions can be further narrowed based on the informational requirements of their job.

Following an individual order and/or process through the organization will add an additional level of access control, making it possible to deny users access to functionality that does not specifically relate to their task. "The principle of least privilege states that only the minimum access necessary to perform an operation should be granted, and that access should be granted only for the minimum amount of time necessary." (3)

Challenges:

To include process flows as part of the overall SSO schema, cooperation across departments is required because the people least likely to have business process information are the administrators and developers tasked with the implementation.

Interviews with departmental managers and end users are necessary to define the flow of work through the organizations systems. While time consuming, it is important to conduct multiple interviews with employees at all levels within a department to ensure the accuracy of the process flow model. The SSO will use that model to further add an additional layer of access control over basic roles. If the model is inaccurate, the SSO deployment may not be tightened down enough to be reasonably secure. It is also possible that access to resources necessary for specific groups or individuals might be denied. An accurate model is necessary to protect resources and avoid potential productivity loss. Again, it must be mentioned that any kind of customized content makes for one stop shopping for a hacker and precautions against unauthorized use must be taken.

Benefits:

The process flow model, when applied on top of basic roles, allows the schema administrator to build an SSO scheme that offers a good level of access control. Basic roles are not enough because there are varying functions within departments and groups. As an example, not all employees working in the customer service department need to have access to the same information. Lumping all customer service employees into a single grouping many grant some users access to information not vital to their job function or deny users access to information they need. Integrating basic roles and the process flow model makes it possible to further narrow or broaden the privileges given to specific user groups.

Through a clear understanding of data flows and work processes administrators are better able to build the security model for the SSO and developers are better able to customize content for varying types of users.

Technology:

Focusing on security, roles and processes are all vital to creating a truly secure SSO deployment. Understanding the various technology these issues span is where it all comes together.

SSOs are driven by LDAP or ODBC databases and used to secure any number applications running on a variety of operating platforms and legacy systems.

It is here that much of the development will have to take place. This development will be based in large part on the authentication method(s) top be employed, information gathered about basic roles, the process model, and how the product was originally architected.

Challenges:

There are many technical concerns that must be taken into account before an SSO can be implemented. Issues that must be considered include any existing LDAP schema, development time, the cost to integrate application sign-on methods with the SSO, scalability and integration with legacy systems.

Directory services provide the capability to organize and access user attributes and information stored in a directory format. This information might include common user attributes such as public key certificates, e-mail addresses, or phone numbers. This means that the SSO's central directory services database is used in place of existing directory services databases to extract information about users and their group memberships. This centralized user data store is the core of the SSO.

The transition to a unified central repository in the form of a central directory service will aid in the smooth implementation of an SSO solution (among other benefits). Integrating the SSO into existing web applications and/or client/server applications will be a major cost driver in implementing SSO capability. Applications may have already implemented various sign-on methods, which are maintained, released, and administered independently. "Advanced SSO solutions provide SDKs so that internally developed or unsupported applications can be incorporated." (4) This process represents the bulk of the time and money spent implementing SSO.

Part of the development effort needs to focus on scalability. The SSO needs to be able to scale with existing applications and able to integrate with new ones. As transactions increase, the SSO needs to be able to handle the increased load. The SSO needs to be able to meet performance demands on a variety of web servers (e.g. Netscape, Microsoft's Internet Information Server (IIS), and Apache). The SSO should also be able to scale across a wide physical distribution. The SSO must also be able to cross Web domains, despite any limitations in the HTTP, without requiring users to repeatedly re-authenticate.

A means of access to existing databases that store user identification (UID) and passwords for legacy sys tems must be developed as part of a complete directory service infrastructure. At the very least a mechanism to interface these databases with the directory infrastructure will need to be implemented and understood fully by the development team.

Benefits:

A successfully implemented SSO provides the organization with increased security, increased productivity and decreased administrative overhead. Additionally, after implementation, the network architecture is better defined in terms of data flow and u ser access.

Adding and removing users from a centralized directory service reduces the amount of administrative overhead and the possibility that departing users are not removed from all systems.

After the implementation is complete, new resources, as part of their deployment, can be integrated with the SSO.

Conclusion:

Over all, if carefully planned and developed, the deployment of an SSO is worth the effort. By maintaining a security focus throughout planning, development and implementation, an S SO can help to solve two key business concerns. If implemented properly an SSO can increase the level of access control within the organization and assist in creating a more secure arch itecture. Users authenticate once and have access only to the resources their job requires .

An SSO ties together a variety of applications and systems to offer a more convenient sign-on for users, an increased level of security across the enterprise and, as a side benefit, a better understanding of roles and workflows insid e the organization.

References:

(1) Connolly, P.J. "Single Sign -On Dangles Prospect of Lower Helpdesk Costs." InfoWorld Test Center. 20 September 2000
URL: http://www.infoworld.com/articles/es/xml/00/10/02/001002esnsso.xml (3 February 2002)

(2) Kelly, Michael. "Is Single Sign on a Security Risk?" SANS Institute. 12 June 2001
URL: http://rr.sans.org/authentic/s so_risk.php (3 February 2002)

(3) McGraw, Gary. Viega, John. "Software Security Principles: Part 3." IBM Developer Works. November 2000
URL: http://www-106.ibm.com/developerworks/security/library/s_-priv.html?dwzone=security (1 February 2002)

(4) Carden, Phillip. "The New Face of Single Sign -On." Network Computing. 22 March 1999
URL: http://www.networkcomputing.com/1006/1006f14.html (29 January 2002)

Tervo, Timo. "Single Sign -On Solutions in a Mixed Computing Environment." Helsinki University of Technology. 4 December 1998

URL: http://www.hut.fi/~totervo/netsec98/sso.html (1 February 2002)

Loshin, Pete. "Single Sign -On" Computer World. 5 February 2001

URL:
http://www.computerworld.com/itresources/rcstory/0,4167,STO57285_KEY73,00.html (5 February 2002)

RSA KEON Single Sign -On SDK
http://www.rsasecurity.com/products/keon/datasheets/ds_keonsso.html

Oblix NetPoint

http://www.oblix.com/products/netpoint/

eTrust
http://www3.ca.com/Solutions/Subsolution.asp?ID=272

Netegrity SiteMinder® 4.6
http://www.netegrity.com/products/index.cfm?leveltwo=SiteMinder

Last Updated: May 22nd, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event