helpsystems

GUIDE *(Cybersecurity)*

A Guide to Practical Single Sign On

The Case for Managed SSO

Executive Overview

The high cost of managing user access to data and software applications with user IDs and passwords is a vexing business issue. The number of users within an organization sets the baseline cost, multiplied by each server and application in your IT environment that requires authentication. Multiply that again by the number of times users are required to change their passwords, then factor in the time spent solving password problems by both end users and your support group, and you're suddenly looking at a substantial expense.

Organizations can, however, eliminate a large part of this recurring cost while investing a surprisingly small amount of time and money up front. The key to achieving this success is to use a fresh, direct business approach to managing authentication.

Like any business problem, your objective should be to identify and implement the solution that provides the best long-term return on investment (ROI). Single sign on (SSO) projects treated as a business problem tend to cost much less, take less time to implement, and provide greater ROI in a shorter period of time.

It turns out that most businesses get the best ROI by using technology they already own to eliminate 60 percent, 80 percent, or more of the business problem—i.e. the high cost of managing passwords—over their entire multi-platform network. Typically, the extra cost involved in achieving 100 percent "single sign-on nirvana" is simply not justified by the estimated ROI.

Since the best solutions involve integrating existing technology across disparate platforms and environments, many IT shops lack the in-house expertise to understand and evaluate the options. When the decision to train, hire, or outsource is ROI-driven, most companies find it most cost-effective to contract SSO experts with a deep understanding of the technology and familiarity with many platforms, environments, and SSO products.

A knowledgeable SSO expert should be able to evaluate your IT environment for SSO within a few hours and identify the best solution within hours or days. Then, depending on the recommendation, implement a solution and train your administrator(s) in anywhere from a single day to several weeks.

Introduction

SSO projects are often started for their own sake. That is, the purpose of the SSO project is to provide a "single sign-on environment" for end users. In turn, these projects are often thought of as a convenience intended to "make end users happier" with the organization's IT services. While this is an important objective for IT, rarely will management approve spending for a project whose only objective is improving the perception of IT services by end users. Even when SSO is regarded as a way to improve productivity, only the productivity of end users is usually considered. The overall productivity of the entire company is not typically a goal of SSO projects.

This incomplete view of what SSO is and the issues it addresses typically leads to solutions that fail, fall short of expectations, or are unnecessarily costly.

However, you can eliminate these problems if you understand three things:

1. the real issue addressed by SSO
2. the various technical approaches to SSO
3. how to calculate ROI for SSO projects

With that foundation in place, you can implement a rational, cost-effective SSO strategy for your organization that maximizes return on investment in a surprisingly short period of time.

The Real Problem Solved by SSO

The average systems administrator assumes the purpose of SSO is to allow end users to access any IT computing resource after logging into the network a single time—a true single sign on.

But this narrow view of SSO leads to the erroneous assumption that any SSO solution that requires end users to authenticate more than once is not a suitable solution. This purely technical view of SSO will lead to over-priced, overly complex solutions that actually undermine the true value of SSO.

The real purpose of SSO is to significantly reduce the high cost of managing passwords across the organization. Re-entering a user ID and password is annoying. But from an organizational cost point of view, it doesn't take much time or money—assuming the user knows and enters the right user ID and password the first time. No, the real cost is all wrapped up in remembering/finding, changing, resetting, and getting help with their passwords . . . all the frustrating parts.

THE COST OF MANAGING PASSWORDS

Why is it so costly to manage passwords? To answer this question, we need to provide a little background on user IDs and passwords. The IT organization is responsible for providing secure access to the company's computing resources. This is done, almost universally, by providing each employee a user ID and associated password for each system (and sometimes also for applications) they need to access.

Because each person has multiple user IDs and passwords, managing them has a multiplier effect. Rather than spending time to manage X number of users, we actually have to manage X number of users times Y number of user IDs per person. You might think the cost of managing each user ID is the same whether the person it represents has one user ID or 50. That is, the costs scale linearly. However, the cost to manage passwords tends to scale more exponentially than linearly. In other words, the cost to manage 50 passwords is more than 50 times the cost to manage a single password. This is because complexity increases non-linearly with the increase in 1) the amount of information to remember and 2) the number of different types of transactions required to change all of them. What seemed trivial in the days of mainframes and terminals explodes into a large and significant cost for the entire organization in a distributed computing environment.

What costs are associated with user IDs and passwords? It's the costs of the create, manage, and delete operations on each of them. Management operations include the cost to change the user ID and/or password and the costs associated with debugging failed logins, resetting passwords, etc.

Of these, the cost to manage passwords is, by far, the single most costly aspect of using user IDs and passwords. Why? Because you spend much more time managing passwords than any other attribute of a user ID. Once a user ID is created, the only attribute that changes, typically, is the password. If any other attribute changes, it typically is changed once at most. On the other hand, the password attribute is changed multiple times. And even in organizations that don't require periodic password changes (yes, there are still some of those around), there are still cases where sign-on failures occur and passwords need to be reset.

The real purpose of SSO is to significantly reduce the high cost of managing passwords across the organization.

THE MULTIPLIER EFFECT

In the average organization, three different constituencies are affected by the cost to manage user IDs and passwords: end users, administrators, and help desk personnel. End users, obviously, need to remember (or get help remembering) and manage passwords. Administrators and help desk personnel are responsible for creating, changing, and helping people manage their user IDs and passwords, as well as for helping people log in when they lock themselves out. Managing passwords isn't limited to helping users set or reset passwords. It also involves helping users debug sign-on failures and then fixing the problem. Of course, most problems associated with being unable to access computing resources are due to password-related problems.

In effect, everyone in your organization is affected by the cost of managing passwords. While I worked at IBM as a lead security architect, I had roughly seven passwords that needed changing every three months and needed to be the same on all systems. I was the lead security architect, right? I knew the password composition rules for each system. I learned the order in which I should change passwords to ensure I had a password that worked everywhere. Still it seemed that more often than not I either had issues changing passwords or using the new passwords after I changed them. At one point, I started tracking the time I spent changing passwords and "recovering" from those changes. I was very surprised to learn that instead of the 10–15 minutes I thought I was spending, it really was taking closer to 35–40 minutes! And I was just one of about 300,000 employees! Assuming 30 minutes on average across all employees, four times a year—that equates to 600,000 hours of time! If the average hourly rate per employee is only $20, that's $1.2 million dollars! And that's just for end users!

Across all organizations, the overwhelming majority of the cost of managing employee access to computing resources is tied up in the cost of managing passwords. Most people are shocked by the magnitude of these costs. When you add up the time spent managing passwords by all end users, administrators, and help desk personnel in an organization, plus the time waiting on the phone for a solution, and the time it takes every employee to change all their passwords four or more times a year, these costs are surprisingly high.

When you understand the real cost of managing passwords, evaluating SSO solutions becomes so much easier. The only valid business reason for embarking on an SSO project is to significantly reduce the cost of managing passwords. Conversely, this means SSO solutions should not be judged by how many sign-on prompts they eliminate. In short, SSO projects should be driven by business considerations; not technology.

This insight has several implications:

1. Before looking for solutions, you need to accurately estimate what you currently spend on managing passwords.
2. For each solution you consider, you need to accurately estimate how much it will cost to implement and manage your solution over time.
3. For each solution you consider, you need to determine how well that solution's approach to SSO actually reduces the cost password management in your organization.
4. When considering potential solutions, you need to compare the projected return on investment (ROI) between them in order to select the best one for your organization.

When you understand the real cost of managing passwords, evaluating SSO solutions becomes so much easier.

MORE PROBLEMS CAUSED BY MULTIPLE PASSWORDS

While reducing the obvious costs of managing multiple passwords is the primary goal, SSO also helps reduce costs that are more implicit. Specifically, the costs associated with the increased risk accepted by organizations due to mechanisms employed by users to help them remember and manage passwords. These mechanisms often impart very high risk onto the organization.

Some examples of these mechanisms are:

- Easily guessed passwords
- Written lists of passwords located under keyboards, desk drawers, overhead bookshelves, on top of the desk, or written on whiteboards
- Lists of passwords stored in files on workstations or network drives
- Shared user IDs/passwords

A physical audit is almost certain to find most of these. True, there are ways to "better" use some of these mechanisms (e.g. encrypted stored files), but they only serve to reduce the amount of additional risk the organization accepts.

In addition to severely impacting productivity, managing multiple passwords tends to reduce employee satisfaction and morale, which can lead to other behaviors that may be detrimental for the organization.

The costs associated with these problems require risk management and analysis techniques and thus are much more complex to measure. However, the easily measurable costs are large enough that that there is usually no need to account for these additional costs.

Technical Approaches to SSO

SSO is one of those terms that, when you say it to someone else, you both assume that you are talking about the same technology. This is very often not the case. Different SSO solutions may take entirely different approaches. Even within a given approach to SSO, there are different implementations.

Most SSO solutions fall into one of these categories:

1. Sign-in prompt elimination
2. Password synchronization
3. Utilizing a distributed authentication protocol that doesn't require passwords
4. User ID and password unification

Some solutions combine two or more of these approaches.

The most important thing to remember is that successfully signing into a system or application entails very little cost. It is only when a sign-on failure occurs that productivity is impacted and organizational cost climbs.

ELIMINATING PASSWORDS

The best possible solution to the high cost of managing passwords is one that eliminates the need to have more than one password. Eliminating a password eliminates the entire cost of managing it. As long as passwords continue to be stored, there will be a continuing cost associated with managing them. Distributed authentication protocols—of which Kerberos is the most highly used and available—don't require passwords to securely identify and authenticate trusted users to network resources. If a password is not needed to authenticate a person to a server system or application, then the password associated with the user ID for that person in that system or application isn't needed and can often be removed from the user ID, thus eliminating the password.

The downside of this approach to SSO is that, while most operating systems support Kerberos, most applications do not. Therefore, client/server applications that perform authentication either must be changed to support Kerberos, or you have to continue to manage the password for those applications.

Another drawback is that Kerberos only works within the intranet. There are some Kerberos derivatives (e.g. CAS) that work over the internet, but they are not widely supported. This means that this approach will not support external website applications.

PASSWORD SYNCHRONIZATION

The next best solution is to minimize the cost of passwords that you cannot eliminate. SSO solutions that take this approach usually include password synchronization. By this we mean making sure that all of a person's passwords are the same for each system and application. When this task is automated, users only need to change their password once and the solution ensures that all of the person's user IDs get changed to the same thing. The idea here is twofold. One, if you can automatically synchronize passwords, then end users spend much less time changing them. Two, even though users continue to have multiple passwords, if they are guaranteed to be the same no matter where a user signs on, end users will enter the correct information much more often and thus significantly reduce the number of calls to the help desk.

In addition to not eliminating the entire cost of password management, there are a couple of other negatives for this approach. First, it can be difficult to formulate a password that is acceptable to all systems and applications. Second, not all systems or applications lend themselves to automated password management. So, SSO solutions using the password synchronization approach may not be able to accommodate all the passwords a person needs to manage.

You should also note that any solution that automates password management, rather than eliminating passwords, will not eliminate the entire cost of managing passwords. Because these solutions require third-party software, there will be license and maintenance fees, plus your costs to administer the product. In effect, these solutions reduce the cost of managing passwords and transfer some of the costs from end users to the IT department. The best solutions utilizing this approach will require the least administrative time and the lowest maintenance fees, thus minimizing the costs transferred to the IT organization.

USER ID AND PASSWORD UNIFICATION

The user ID and password unification approach involves using LDAP to store a network user ID and password in one location, the LDAP repository, and then have all systems and applications use this one, stored instance of the user ID and password for authentication. This is a very appealing solution. It eliminates passwords and user IDs. However, the reality is that this approach is really only practical for web- and web-server-based applications and for some UNIX and Linux implementations. It can also be relatively expensive to switch to this SSO approach. This approach often makes sense for organizations that are already using LDAP authentication for one or more systems or applications in their networks.

ELIMINATING SIGN-ON PROMPTS

The least effective solution is one that focuses only on eliminating sign-on prompts and which does nothing to reduce the cost of managing passwords. This is because, as we argued above, that successful sign-on attempts are not very costly to the organization. Password management is the real cost culprit and is the reason for failed sign-on attempts. The good news is that most SSO solutions that attempt to eliminate sign-on prompts also include some sort of automated password management capabilities.

Most SSO solutions using the prompt elimination approach include some form of process to "learn" each person's user ID and password for each system and application. This involves detecting when a login operation occurs and capturing the information provided. The next time that user signs in to the same system or application, the solution "replays" the information it learned in the initial sign on. Solutions using this approach often include hardware appliance components as well as software components as part of their product.

Prompt elimination approaches store the credentials for each person somewhere in the network. This means that each user ID and password is stored in two places: wherever each system or application stores them and wherever the SSO solution stores them. Depending on where and how the SSO solution stores this information, this approach can create a fairly large security vulnerability.

In addition, rather than eliminating passwords, this approach stores another copy of every user ID and password. So, when passwords need to be changed, each one must be changed in two places. Often, these solutions provide some sort of mechanism to automate password management, including the copy created by the solution itself. If, however, a solution using this approach doesn't provide a nearly foolproof way to automate this task, this approach can create additional problems and costs you hadn't faced before.

Very few solutions are able to eliminate all sign-on prompts. Some only work for system sign on and not legacy applications. Some don't handle web applications. Some only handle web-based applications. This doesn't necessarily mean these solutions aren't potentially valuable. It means that you need to understand what they do, how much they cost to implement and manage, and how much, if at all, they will reduce the current costs of managing your passwords. It also means that combining approaches or solutions may provide the best overall ROI.

Architecting the Best Solution for Your Organization

No single SSO approach can eliminate all sign-on prompts in any network environment except, perhaps, the most basic. Fortunately, we have also shown that eliminating prompts should not be goal of any SSO project. The objective of any SSO project should be to **significantly reduce the high cost of managing passwords for the entire organization.**

As stated earlier, SSO is a solution to the business requirement to reduce the cost of providing secure access to computing resources. As such, return on investment is the best metric to use to select the best solution for your organization. Before describing how to calculate ROI for SSO projects, we'll discuss the process of identifying the potential solution, or combination of solutions.

THE PROCESS

The first step is to determine how much your organization currently spends on user ID and password management. We describe this process briefly in the "How to Calculate ROI for SSO Projects" section below.

The next step is to understand your network environment. This involves answering the following questions:

- How many user IDs, on average, does each end user manage today?
- How many different types of systems, legacy applications, client/server applications, and web-based applications does the average user access?
- How many external web applications does each user access?
- Since you typically don't manage user IDs and passwords for these applications, do you want to include these in your solution?
- What specific types of systems does each user access? What specific applications does the average user access?
- Do your users initially authenticate to a Windows domain before accessing other services?
- Do any of your users access your internal network over the internet?
- If so, how do you facilitate this capability? For example, do they use a VPN? If so, what specific VPN solution do you use? Does the VPN solution require a separate user ID and password? Does it use the person's Windows domain user ID and password?

Your specific answers to these questions will begin pointing you toward the best combination of approaches to use in your environment.

The next step is to determine which, if any, particular SSO approach seems to address the password management issue for the greatest number of people and/or the most types of user IDs. Then identify which approach affects the next greatest number of people or user IDs.

TIP: For most companies, eliminating passwords with Kerberos and Enterprise Identity Mapping (EIM) ends up being the best starting point for SSO, even if additional approaches or solutions are needed as well. So, always be sure to include this dynamic duo in your evaluation.

Next, use the results from the previous steps to begin identifying specific potential solutions. Google (or Yahoo, or Bing, etc.) search is likely to be the most efficient way to accomplish this task.

Now, you need to investigate each potential solution to understand how, at a high level, they provide the capabilities they claim. This step will weed out some of the options.

For each remaining option, calculate the estimated ROI for that solution. Technical folks will roll their eyes at this statement. But this calculation is not only necessary, it is also not that difficult.

Many companies will find that a single solution reduces password management costs by 80 percent or more. Assuming the cost to acquire, implement, and manage that solution is relatively low, many companies will choose not to address the remaining 20 percent of the cost. In other words, if the ROI for a single solution is large enough and the ROI to address the remaining costs are too low, it might not make sense to attempt to address the remainder of the costs.

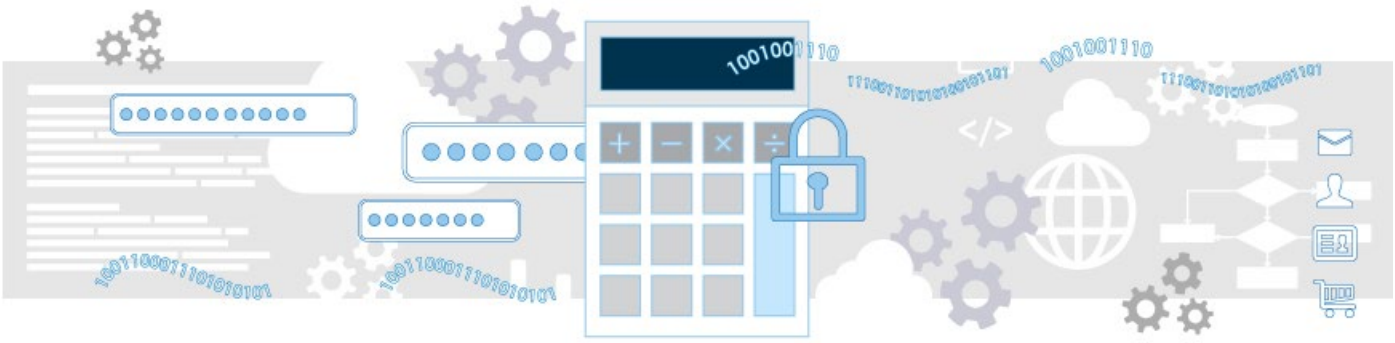
HOW TO CALCULATE ROI FOR SSO SOLUTIONS

Calculating return on investment (ROI) for SSO projects involves calculating your current costs for managing passwords, calculating the cost of a potential solution, and estimating how much of your current costs the estimated solution will eliminate. Next, calculate net savings by subtracting the solution costs from the estimated cost savings due to the solution. Finally, divide net savings by the solution costs.

Calculating the ROI for SSO projects requires an estimate of the current cost of managing user IDs and passwords, the total cost of the password solution(s) you are considering implementing, and an estimate of how the solution will reduce the overall cost of managing passwords.

You can estimate the current cost of managing passwords by adding up the time, on average, that end users spend changing passwords and getting help from the help desk and or administrators and applying an average burden rate (or salary) for the entire company to that time.

Added to this is the cost of the time that administrators and/or help desk personnel spend on password related issues. The burden rate applied to administrator/help desk time is often higher than the average burden rate across the whole organization. When calculating the cost of a solution, you need to include the cost to acquire, the cost to implement, and the cost to manage as well as any yearly maintenance charges.



Train, Hire, or Outsource?

Another important decision affecting ROI and the speed of implementation is whether to train in-house resources to evaluate and implement your SSO solution, hire the expertise as a direct employee, or outsource the project to a service bureau.

Single sign on represents one of the most difficult security tasks to handle with in-house resources. The technology is complex, and it requires very specialized knowledge that you use once when you implement, and then not again until a new release of Windows or an application introduces a new variable that conflicts with your SSO structure. When that happens, your organization needs an immediate solution.

When beginning your SSO project, look for a project leader with in-depth knowledge and experience with:

- Security protocols on all the platforms within your network
- Complex authentication protocols
- Related technologies such as EIM
- SSO technologies and ISVs

This expertise is not usually readily available in most IT organizations.

This leaves organizations with the options of growing, hiring, or contracting the required expertise. Growing the expertise will be time consuming and costly. Hiring someone with the specialized skills will also be costly. Contracting the expertise can be the cheapest solution; however, if you don't hire an independent contractor you'll end up with the solution the contractor's company sells. This may or may not be the right solution for you.

Managed SSO Services

"Managed" SSO services often provide the greatest ROI for SSO projects; assuming the service provider doesn't sell or receive any kind of monetary remuneration for selected solutions. Vendors providing these services already have the technical and business expertise required to help you though the process of identifying, implementing, and supporting SSO solutions. When provided at a reasonable cost, managed SSO services will provide the best solution at a fraction of what it would cost the average organization to develop in-house expertise.In addition, the ongoing support of your solution gives you piece of mind that if any problems do crop up, you have the expertise needed to resolve it readily available.

Of course, just as with any SSO solution, you need to do an ROI calculation for any managed SSO service you consider. Then you need to compare the ROI for managed services against any other solutions you may identify. Only if the ROI for managed services is better should you choose this solution.

AN EXAMPLE: Managed SSO

Following is just one example of how a managed service by an experienced SSO consultant can expedite the path to cost savings and positive ROI:

STEP 1: Evaluation and Project Plan

A typical SSO engagement with HelpSystems starts with a complimentary evaluation call—usually 30 to 45 minutes—to discuss the customer's environment and requirements. Usually the environment includes a Windows domain and one or more non-Windows systems, such as UNIX (AIX), Linux, or IBM i. Some environments are far more complex, others are more straight-forward.

We may use an ROI calculator to understand the customer's current password management costs. This becomes increasingly important in very complex SSO environments or when management desires tangible metrics on ROI.

Based on your input during the evaluation call, we work up a proposed statement of work outlining all aspects of the project, including price. If your environment includes applications that require specialized attention, that will be called out.

Contrast the speed of this process to the elapsed time necessary to create a project plan using internal resources, even with a project leader who already has some SSO experience.

STEP 2: Internal Preparation

The most important thing you can do to prepare for SSO implementation is to ensure that your host name resolution for your server systems is consistent between the Windows domain and the server system itself.

Typically, the best way to do this is to configure your server to use the Windows domain DNS server as the primary host name resolution service. You should maintain a host table file on your server only if there are server applications that use non-standard host names to access one or more remote systems.

You can also generate a CSV file containing Windows user IDs, first and last names, middle names and/or initials (if any), and if your company uses them, the employee numbers. This information can be quickly and easily dumped from Active Directory.

STEP 3: Implementation

For the remainder of this example, we'll assume the environment includes one or more IBM i systems; although the example would be much the same if the servers were UNIX, AIX, Linux, Mac, or any other OS.

SSO setup is typically done via a single interactive, online session with your Windows (or Active Directory) administrator and the systems administrator of the non-Windows server(s). We give the administrators presenter status, so we can see their desktops while they do the typing under our direction. This protects your systems and helps cement the knowledge transfer.

Together, we work through the following set-up process, making configuration choices appropriate to your unique environment and troubleshooting anomalies as they occur.

1. Network Setup
2. Kerberos Configuration
3. EIM Configuration
4. Add Servers to SSO Network
5. Test
6. User Identity Mapping
7. Enable SSO for All or a Subset of Users

This entire implementation stage typically takes less than a day.

For many customers, Kerberos and EIM enablement is all they want and need. In this case, their end users no longer need to have a password even on the non-Windows server(s) configured for SSO. This eliminates all the cost associated with managing passwords on those servers.

STEP 4: Ongoing Support

Although SSO technology tends to be "set-it-and-forget-it," the environment it manages is not. Periodically, OS and application changes conflict with Kerberos and EIM settings. When this happens, it's critical to have quick access to a resource experienced in troubleshooting SSO. This ongoing support is an important part of a managed SSO service.

In addition, as you add new applications, you and your users will want them included in the SSO network. And some customers want to extend SSO to web applications, Lotus Notes, SAP, or other critical applications. In most cases, the cost of managing passwords in these other environments can be significantly reduced, but this requires additional configuration and sometimes additional software utilities. The consulting time required to analyze the application(s) and determine what can be done is covered in the basic managed SSO support contract. If the customer decides to proceed with our recommendations, the additional work is covered under a separate contract.

Finally, ongoing support comes in handy during audits and when you add or change servers, host names, or IP addresses.

Summary

Historically, single sign on (SSO) has been perceived as a complex technical project requiring a comprehensive solution that encompasses all of a user's applications. This all-or-nothing approach to SSO typically leads to solutions that are costly and time-consuming to implement, with long payback periods.

However, when organizations take a straightforward, sensible business approach to SSO—utilizing return on investment (ROI) analysis—most find that they are quickly able to mitigate anywhere from 50 percent to nearly 100 percent of their password management burden. By utilizing technology they already own and use today, they often recoup their investment in 2–6 months, even when the solution serves just a few hundred end users. This fresh ROI approach to SSO brings the benefits of SSO within easy reach of nearly any organization, particularly when implemented using a managed SSO service provider.

How Much Can SSO Save You?

To find out how much SSO Managed Services can save your organization, talk to an SSO pro today.

About HelpSystems

Organizations around the world rely on HelpSystems to make IT lives easier and keep business running smoothly. Our software and services monitor and automate processes, encrypt and secure data, and provide easy access to the information people need.