Single Sign-on
Finding the Best Fit for Your Business
Is Single Sign-on Right for Your Business?

Single sign-on has long been touted as the ultimate solution for a growing number of login and password-management challenges. From system server logins to Web-based Internet banking and e-mail programs, users are often bombarded with a vast number of different system logins each day—most of which require different usernames and passwords. With single sign-on, users log in one time, and the computer then takes over to perform any subsequent logins automatically on behalf of the user. This can reduce or even eliminate the burden of remembering many different passwords, which can dramatically improve convenience for users, lower password-related helpdesk calls and strengthen security.

Although single sign-on has been around in various forms for years, adoption of this technology was relatively slow until 2005, when adoption started to increase rapidly. One reason is technology maturity; another is the extensive research published by Gartner, Forrester, Bloor and other analyst groups stating that traditional passwords have reached the breaking point of usefulness.

This paper looks at the current standing of single sign-on authentication technology and explores the benefits, security concerns, costs and return on investment (ROI) that can come from the right kind of single sign-on solution. The scope of this paper is limited to enterprise single sign-on (ESSO) and does not address Web single sign-on (Web SSO). ESSO is the more complex version of single sign-on technology because it manages user interactions with a myriad of different legacy and production applications. Web SSO is limited to managing single sign-on for pure Web-based applications, which are fewer in number.

Chapter 1: An Overview of Single Sign-on

Single sign-on is not a new concept. As the name suggests, single sign-on is designed to take the many logins a user typically performs among various systems and reduce the logins to a single operation. In an ideal single sign-on situation, users should be able to log in one time—and have further authentication requests be serviced automatically by the software rather than by the user. This means that data required during a login—such as username and password—are presented to an application without users having to remember these values themselves.

Research studies by analysts, including Gartner, show that large corporations typically have more than 70 applications or systems that require a person to log in and present some kind of credential, such as a password. This basic fact of corporate life clearly illuminates the need for an effective single sign-on solution.

In this white paper, we will refer to single sign-on as something that presents credentials on a user’s behalf to perform a login. In this context, single sign-on does not include systems that synchronize multiple user passwords to the same value.
Figure 1: The overwhelming number of passwords and logins users face in a typical organization

How Does Single Sign-on Work?

To understand how single sign-on works, it’s important to understand the different single sign-on methods—and to know how the technology performs the single sign-on process.

A typical user performs an average of five to 10 different logins on a regular basis. The single sign-on (SSO) product must therefore remember and store those login details so that it can present the proper credentials to the required application. Although the concept remains largely the same across most single sign-on products, there are important differences in how user credentials are stored and accessed.
Local Storage

At first glance, a local storage single sign-on solution may seem sufficient for most basic requirements. This is true to some degree, since theoretically no other user should require another person’s credentials.

But this architecture has some serious limitations in a network environment. Because a user’s login credentials are stored on a single workstation, users can’t take advantage of single sign-on capabilities if they use more than one machine to perform their duties. Of course, you can choose to duplicate credentials across multiple computers, but as the number of workstations increases, this becomes an unacceptable administrative burden.
This type of solution also has significant administrative and management drawbacks. Effective, centralized management and control becomes virtually impossible; synchronizing data between workstations is difficult; and overall systems management is slow and inefficient.

Although some more advanced local storage systems allow for limited forms of centralized administration, such as auditing, there is a definite cut-off point where local storage-based systems become unworkable for most medium and large businesses.

**Directory Storage**

Within larger network systems, users need the ability to roam and log in from any workstation to perform their duties. Server or directory-based single sign-on systems make it possible for user credentials to be administered, stored and retrieved from a central server repository.

*Figure 3: Directory-based single sign-on*

Not only does this enable more powerful and effective centralized management capabilities, but it can actually reduce the effort of managing each user’s access portfolio. Most products based on this type of system allow detailed configuration of user access, auditing of user activity and enforcement of security policies to ensure the use of strong passwords.
Having a single server to hold all user data is convenient, but this design can fall prey to the high risk of a network breakdown in the event of a single server failure. This is the main reason most enterprise systems tend to offer high availability by distributing the user credentials over more than one server.

**Combined Systems**

In large organizations with large numbers of remote and mobile users, it is not always possible to guarantee a connection to the server. As a result, an ideal single sign-on system should combine both the local and directory storage methods discussed above. This makes it possible for users to access their single sign-on credentials, whether or not they’re connected to the network.

**Chapter 2: Current Products and Technology**

Single sign-on vendors use a number of different approaches to reduce the number of logins for users. Unfortunately, there is no standard for tackling the issues of single sign-on because today’s modern production environments tend to be extremely diverse in their requirements. While explaining the varied technologies capable of providing single sign-on is beyond the scope of this paper, the following section will focus on the major approaches.

**Ticket-based Systems**

Although not strictly labeled as single sign-on, ticket-based systems represent some of the first attempts to provide single sign-on capabilities. (Kerberos in the early 1990s is one notable example.) These networks operate by providing an authentication service, which is used to issue a valid ticket for an authenticated user. The user then presents that ticket to gain access whenever he or she is challenged for authentication. This process generally happens seamlessly without the user ever being aware of the need to authenticate, thereby giving the impression of single sign-on.

![Figure 4: Overview of ticket-based access](image-url)
Cookies
This ticket-based concept can still be found in many of today’s business and enterprise systems. On the Internet, cookies are one good example of a ticket-based approach. When a Web site obtains a user’s credentials, an authentication ticket (cookie) is placed on the user’s machine. Whenever the user accesses the site, the site checks for the cookie, and if present, the user is not prompted again for a login. Unfortunately, cookies are easily copied or deleted, making them an unreliable source for credential storage.

PKI and Digital Certificates
A similar idea starting to take hold is the digital certificate, which forms part of a public key infrastructure (PKI). A PKI is designed to establish a basis for trust between digital identities. Within a PKI, a certificate authority (CA) will check a user’s credentials and issue a digital certificate. With other systems, someone may steal a user’s ticket or cookie in order to gain access to a system. With a PKI, this is much more difficult due to the underlying public key cryptography. This enhanced security is one of the main benefits that sets this system apart.

PKI also introduces the concept of no repudiation. Vitally important in today’s electronic business transactions, nonrepudiation enables the system to legally vouch for the user’s identity. As we will discuss in more detail in Chapter 3, a user’s identity and the associated security aspects are important considerations for any single sign-on implementation.

Integration
The underlying concepts behind PKI systems are valid, but they introduce a number of difficult integration challenges. To be successful, an infrastructure needs to be designed specifically for a PKI-based single sign-on system. This may necessitate some rather radical changes to existing architectures, and it may narrow the scope of potentially compatible applications, since applications generally have to be specifically engineered to utilize the technology. These interoperability issues—along with the high costs of integration—are seen as the major stumbling blocks that have held this technology back for so many years.

However, the development of industry standards such as RFC2109 for cookies, X.509 for digital certificates and the public key cryptography standards (PKCS), have helped address these issues. Consequently, the adoption of these types of single sign-on solutions is on the rise.
Automated Login Systems

Emerging as the leading single sign-on technology today, automated login systems are designed to provide user credentials directly to the application, without the need to alter the underlying infrastructure or the application itself.

Figure 5: Overview of a typical automated login

These automated login systems are easy to integrate into existing environments because applications aren’t even aware that single sign-on software—not a person—is presenting the credentials. As a result, almost any type of application can be single sign-on-enabled. Not all automated login systems are the same in this regard. It’s important to understand how your single sign-on software knows what credentials are being requested—and how those credentials need to be presented. Single sign-on solutions use a number of different methods to enable different applications for single sign-on.
Figure 6: Example of single sign-on scripting

**Scripting**
Many single sign-on products use scripts to integrate single sign-on capabilities into popular packaged applications—such as enterprise resource planning (ERP) or customer relationship management (CRM) applications. These scripts map the appropriate fields from the application to the user’s credentials.

As you consider different single sign-on products and approaches, make sure there are pre-existing scripts for your applications. Furthermore, it’s important to carefully consider the effort you will have to expend to make all of your required applications part of your single sign-on solution.

A good single sign-on implementation will not only have existing scripts to save time on implementation but will also offer wizards or tools that detect the appropriate login fields and automatically create scripts based on that input. Make sure your single sign-on solution includes an easy-to-use graphical user interface (GUI) for performing these functions.

**Multifactor Authentication**
Historically, single sign-on has always been perceived as something that weakens system security—mainly because it reduces authentication to a single validation event. Because of this, it’s important to make the initial
login very secure, so you can stop malicious parties from gaining unauthorized access to multiple systems using a single bogus login.

Advancements in secure user login technology make it possible to alleviate these concerns using strong authentication methods. For example, multifactor authentication uses two or more methods to identify the user. This generally includes something the user owns (a smart card or token), something the user knows (a password or PIN) and something only the user can present (a biometric ID, such as from a fingerprint or retina scan).

Any good single sign-on implementation should offer a wide range of advanced authentication options to boost the security of the user’s single login event.

Integration Environments
An important consideration in any single sign-on implementation is how well that product fits into the existing IT infrastructure. It is doubtful that any off-the-shelf product exists that covers every possible IT infrastructure, but it’s important to evaluate how well various single sign-on solutions support different industry standards, hardware and software.

Lightweight Directory Access Protocol (LDAP)
The Lightweight Directory Access Protocol (LDAP) was designed for updating and searching directories running over TCP/IP networks. Large vendors like as Microsoft with Active Directory, Novell with Novell eDirectory and Netscape with Netscape Directory Server have all adopted LDAP as a standard for directory services.

Due to the widespread use of LDAP directories, a single sign-on product should provide built-in support for LDAP so it can work effectively with today’s modern infrastructures.

Terminal Emulators
Since most access to mainframe infrastructures is performed via a terminal session, it’s more difficult for a single sign-on solution to present mainframe and other text-based applications with the appropriate credentials.

In earlier sections, we identified that a single sign-on solution recognizes when to supply the user’s credentials—and then places those details automatically into the required login window. A problem with terminals—or any other text-based interaction—is that there are no screen objects such as login boxes or OK buttons. Enabling single sign-on for these types of environments generally requires the use of custom-engineered software that integrates well with the terminal emulator used to access the mainframe or text-based application. The terminal emulator typically has the support for single sign-on built into the application, so it recognizes when and how to supply the user’s credentials as part of a terminal session.

There are literally hundreds of different terminal applications, so it’s important to carefully consider what types of terminal emulators your single sign-on solution can support. Support for HLLAPI mainframe terminal emulators is simply not enough. There are literally dozens of terminal emulators out there, and you need to make sure a single sign-on solution can support all of your mainframe and other text-based applications before you make a decision.
**Self-service Password Reset**

Lost or forgotten passwords is a notoriously common problem for every IT helpdesk. Single sign-on addresses the issue by reducing the number of passwords people have to remember, but users can still forget their primary passwords—even when they only have one to remember.

Self-service password reset can help eliminate this problem. It allows individuals to reset their own passwords, generally by answering a set of questions that only they would know. To provide general access, most of these types of systems can operate via a Web interface. More obscure methods include voice-recognition systems that allow password reset to be performed over the phone. However, since these more advanced systems can be very expensive, most businesses prefer simple Web-based password self-help solutions that don’t require any special hardware or infrastructure additions.

Single sign-on and self-service password reset complement each other, although it’s important to consider the compatibility between any single sign-on and password-reset application. Businesses looking to integrate these types of solutions should look for vendors that offer both types of technologies in order to provide maximum compatibility, reduced complexity and lower costs.

**Single sign-on basics:**

- Single sign-on is designed to reduce the number of logins users have to remember and create a more efficient and streamlined work environment.
- Modern single sign-on products achieve this by presenting login credentials automatically.
- Single sign-on for individual systems typically stores user credentials locally on the workstation.
- Single sign-on within network environments typically stores credentials on a central server or within a directory.
- Directory-based systems should provide high availability by replicating credential storage.

For enhanced flexibility, a single sign-on application should be able to provide both central server and local methods of credential storage.
Chapter 3: Security

Security on networked workstations and servers has always been important, but as computer networks expand and our dependence on these systems increases, so does the requirement for better security.

The complexity of most modern, large-scale business networks can be frightening from a security perspective. With standard open connectivity to the Internet, most businesses face having to secure their systems at various levels. This often forces users to authenticate separately to each individual area.

Access control is a primary component of any IT security system, and it should provide modern and proven technologies for securing sensitive information from outside attack or access.

Authentication systems have largely remained unchanged, and the standard login of username and password/PIN is still one of the most commonly-used methods for ensuring that the right people can access the right information—and nothing else.

Eliminating Passwords: the Pros and Cons

One argument against single sign-on has always been that the technology partly eliminates the security barriers that others have worked vigorously to enforce. Why bother having six different layers of authentication, if single sign-on reduces them all back to one?

To answer this question, you need to consider whether those six logins are actually increasing system security—or simply preventing a more efficient workflow. Are users really storing passwords in their heads—or just writing them down somewhere? How secure are their chosen passwords if they select weak values such as “1111” or “abcd” in order to remember them?

A study by the Hurwitz Group indicated that users cannot remember more than three passwords but are generally required to remember six or more. Recently, an unemployed British computer administrator was caught as he hacked into 92 separate U.S. military and government networks, often getting past easy-to-guess passwords to download sensitive data.

Studies have found that the fewer password values users have to remember, the more secure their password choices become. Reducing the number of password values also reduces the temptation to write them down on paper. So in many ways, reducing the number of passwords people have to remember actually increases system security.

Security Policies

The requirements for different levels of security vary with different applications, and every business creates security policies to define and enforce security across all its systems and applications. This creates another important consideration for your single sign-on solution. You need a solution that is flexible enough to support and enhance all your corporate security policies. Your single sign-on implementation should allow for sophisticated password policies, including the ability to define and enforce password length, specify allowable characters and manage password expirations. This makes it possible for you to extend your security policies to applications that would otherwise not allow password configuration.
System Security
The type of information referenced by a single sign-on implementation is obviously going to be highly confidential with user login credentials stored somewhere on a server or local workstation. As a result, you need to carefully examine the security of your credential storage solution to ensure that confidential data cannot be accessed by malicious parties.

Credential Storage
Due to the inherent weaknesses found in most operating systems, encryption is the only viable and effective method for providing protection against unauthorized access to sensitive data.

Encryption is the process of mathematically altering data using a given algorithm and a key value. There is a large range of algorithms available for use, and the best choice depends on the type of encryption that needs to be performed. For the purposes of data protection, an implementation should use industry-standard algorithms such as the data encryption standard (DES) or the advanced encryption standard (AES).

Just as important as algorithm selection is the choice of key length for the encryption. When discussing key lengths, the value is given as a “bit” length, referring to how many digits are represented in the key value. As a general guideline, the higher the “bit” length, the longer the key length and the greater the security that the encryption provides.

It is not safe to assume that just because credentials have been encrypted, they will be safe from outside attack. You also need to consider the security of the key used in the encryption and decryption of data. Gaining knowledge of the key value will generally compromise all the credentials which are secured by it.

As a result, the security behind any single sign-on solution should be designed to protect and secure key information. The most secure methods include:

- Storing the key on another secure device such as a smart card.
- Not storing the key in any fashion, but rather deriving the key value each time users log in using their PIN or password.

Most importantly, the key value should never be stored on a workstation, where an attacker can easily find it.

Transmitting Credentials
Securing the storage of user credentials is important. However, it's equally important to secure the transmission of credential data between servers and workstations. The communication lines between different nodes in a network are generally not secured from attacks, such as wiretapping or traffic monitoring.

To preserve the security of a user’s credentials during transmission, the single sign-on application should maintain encryption of data when sending communications across a network.
Advanced Authentication

Every security system has a weak point. With single sign-on, the major weakness lies in the initial user authentication, because one weak password or PIN can compromise the rest of the system. Good single sign-on products have the ability to enforce a certain password strength, but even this may not be sufficient for high-security environments. To address this issue, most reputable single sign-on vendors offer the additional security benefit of integrating their product with an authentication token, a smart card or a biometric device.

Authentication Tokens

Tokens are one extremely popular alternative to passwords. Tokens are small devices that emit a one-time password on an LCD display. The one-time password is calculated by an internal mathematical algorithm that combines the time of day and the secret serial number of the token to produce an eight- or ten-digit password. The one-time password, together with the user ID, can then be used to login to the single sign-on solution. Since the one-time password is invalid after a short period—say one to two minutes—it is difficult for hackers to intercept the one-time password and reuse it later.

More advanced tokens also use a PIN entry and an internal event counter to calculate the one-time password. The addition of the PIN is popular with banks, which can then use the token for digitally signing online transactions. The event counter is a simple count of the number of times the one-time password has been sent to the authentication server. Event counter one-time passwords expire the moment they are used, eliminating the one- or two-minute period when the one-time password is valid. This approach is popular with banks, retailers and online securities trading companies. Fortunately, both time-based tokens and time-PIN-event tokens are relatively inexpensive to deploy.

Smart Cards

A number of vendors offer smart cards, and this technology is sufficiently mature for any enterprise to adopt. Smart cards have multiple uses, including one-time password generation, PKI credential management, e-mail encryption, transaction signing, digital signatures and as employee badges for building access.

One-time passwords or PKI credentials are accessed by inserting the card into a smart card reader and entering the user’s secret PIN. The one-time password or PKI credential can be used to login to the single sign-on product, replacing the user ID and password typically used. Since the PIN is useless without the smart card and
the card is useless without the PIN, the combination is typically called “two-factor authentication” or “strong authentication.”

Compared to single-factor credentials like passwords, it is much more difficult for two-factor authentication solutions to be stolen. Most hackers and cyber thieves operate completely on the network and are therefore thwarted by their inability to acquire the physical smart card. Moreover, the PIN is only exchanged between the smart card and highly secure PC software. The PIN never gets sent across the network. Additionally, smart card users are much more reluctant to loan their smart cards to others, since they cannot use the system while the card is loaned out.

Ideally, the single sign-on product you choose should be integrated with at least one smart card and one authentication token product.

**Biometrics**

The range of biometric authentication devices is growing, but you should take care in choosing a solution that effectively enhances system security without sending costs through the roof. In most cases, biometric methods are further enhanced with the addition of a swipe or smart card that provides multifactor authentication.

In the world of biometrics, it’s important to pay close attention to the accuracy of the device. In terms of biometric readers, performance is generally measured against the following criteria:

- The False acceptance rate (FAR). This specifies the likelihood that an impostor could be incorrectly accepted by the system.
- The False rejection rate (FRR). This specifies the likelihood that a genuine user could be rejected by the system.

Reputable manufacturers will publish their FAR and FRR figures. As you investigate different biometric offerings, look for the lowest possible FAR and FRR numbers.

**Fingerprint Scanning**

The concept of fingerprint scanning is quite old, but the technology has only recently evolved to the point where it can be reliably used and implemented. Due to the increasing number of providers that are now selling this kind of hardware, the fingerprint-scanning method has become the most cost-effective biometric authentication method.

The accuracy of a fingerprint is generally measured in the number of minutiae points that the fingerprint reader captures. The security and reliability associated with such scans is obviously highly dependent on the quality of the scanning hardware. When considering this type of approach, you should select quality hardware to ensure trouble-free, effective operation.

**Hand-geometry Scanning**

The scanning of a user’s hand is based on measuring differences in hand size as well as finger length, thickness, curves, the distances between and shape of the joints and the overall bone structure.

Although the accuracy is often disputed, manufacturers claim that the FAR and FRR rates are on average 0.1 percent, or one in one thousand. The costs of implementing hand geometry scanners are generally higher than
fingerprint scanners. Hand-geometry scanners are widely used by various government departments. The main advantage of this method is that scans are quick, simple and noninvasive.

Eye Scanning
The human eye is still considered to be one of the best sources for a biometric identification scan because it does not change throughout a person’s life. The technology can be divided into either iris or retina scanners. Both are video-based systems that take a photo of the user’s eye to measure various characteristics.

The advantages of this system are the extremely low error rate, which is why eye scanners are found in some of the highest-security installations. Given that the required hardware is still quite complex and only produced in limited quantities, iris or retina scanners are generally prohibitively expensive to implement across an entire enterprise.

Chapter 4: Cost of Ownership
When considering any type of change or addition to your infrastructure, it’s important to look carefully at the cost of ownership imposed by the change.

Integration Costs
Depending on the type of single sign-on solution you implement, you should carefully consider a number of important cost factors:

- Purchase and licensing
- Infrastructure change
- Application integration
- Usability and training

Purchase and Licensing
The initial investment for single sign-on will largely depend on the number of users the system will accommodate. Apart from enterprise and volume licensing discounts, points to consider are ongoing costs, such as technical support agreements. Good single sign-on vendors will generally provide help with product integration and 24-hour global technical support.

Infrastructure Change
Depending on the single sign-on implementation you choose, you may need to make significant infrastructure changes to accommodate the new system. This is often the case when you deal with systems that require connections to specialized authentication services, particularly with PKI solutions and biometric hardware.

Ideally, the solution to integrate single sign-on should not involve any significant changes to your existing applications or hardware. The best single sign-on products are flexible and configurable enough to integrate into most existing environments.
**Application Integration**

The size of your integration effort will depend on the number of applications that have to be single sign-on enabled. Unfortunately, as the size of an organization grows, so does its number of applications. In these situations, designing and implementing a complete, effective single sign-on solution may require a substantial effort.

For medium- to large-scale integrations, it is critical to check that distribution to client workstations can be automated and performed from a centralized server. Most server architectures support automated installation strategies and tools (e.g. Novell ZENworks, MSI, SMS, Tivoli and Unicenter). You should confirm compatibility with the tool you use so that you can minimize the time and cost of deploying the solution.

As Figure 8 indicates, manually integrating single sign-on to individual workstations may be more suitable for a small number of users, but automation will generally translate into direct cost savings as the number of users grows.

*Figure 8 : Costs for manual and automated application integration*

A key point discussed in Chapter 2 was the fact that single sign-on should provide built-in support for a wide range of popular applications. This is important from an integration perspective because it ensures that no re-engineering will have to take place to integrate legacy applications into the new environment.
**Usability and Training**

New technology always brings training challenges. Make sure your single sign-on solution has a minimal impact on your users and IT staff. Single sign-on should offer transparent operation and centralized administration. Use of the product should be intuitive enough to minimize—or even eliminate—expensive training.

**Return on Investment**

The ultimate test of any single sign-on solution is the ROI it can offer your business. Since single sign-on is designed to enhance productivity, so it should show solid ROI figures across the application spectrum. It’s important to remember that the wrong type of single sign-on solution can dramatically reduce the effective return.

As shown in Figure 9, selecting the right feature set for single sign-on depends greatly on the number of users the system will need to support.

![Return on Investment Graph](image)

**Figure 9 : Return on investment overview**

A qualitative ROI measurement must also include outside factors your single sign-on deployment may affect. Password-management challenges affect many areas of your business, and they can cost your enterprise a substantial amount of time and money each year.
<table>
<thead>
<tr>
<th>Number of Users</th>
<th>500</th>
<th>1,000</th>
<th>5,000</th>
<th>10,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average number of helpdesk calls per year</td>
<td>8,100</td>
<td>16,200</td>
<td>81,000</td>
<td>16,200</td>
</tr>
<tr>
<td>Number of password resets performed by helpdesk per year</td>
<td>2,025</td>
<td>4,050</td>
<td>20,250</td>
<td>40,500</td>
</tr>
<tr>
<td>Cost per year for password resets</td>
<td>$42,525</td>
<td>$85,050</td>
<td>$425,250</td>
<td>$850,500</td>
</tr>
<tr>
<td>Cost per year for password related calls using single sign-on</td>
<td>$8,505</td>
<td>$17,010</td>
<td>$85,050</td>
<td>$170,100</td>
</tr>
<tr>
<td><strong>Password reset savings</strong></td>
<td><strong>$34,020</strong></td>
<td><strong>$68,040</strong></td>
<td><strong>$340,200</strong></td>
<td><strong>$680,400</strong></td>
</tr>
<tr>
<td>Average cost for all users to log in each year</td>
<td>$14,375</td>
<td>$28,750</td>
<td>$143,750</td>
<td>$287,500</td>
</tr>
<tr>
<td>Average cost for all users to log in each year using single sign-on</td>
<td>$4,313</td>
<td>$8,625</td>
<td>$43,125</td>
<td>$86,250</td>
</tr>
<tr>
<td><strong>Approximate productivity savings per year</strong></td>
<td><strong>$10,063</strong></td>
<td><strong>$20,125</strong></td>
<td><strong>$100,625</strong></td>
<td><strong>$201,250</strong></td>
</tr>
<tr>
<td><strong>Total Savings per year</strong></td>
<td><strong>$44,083</strong></td>
<td><strong>$88,165</strong></td>
<td><strong>$440,825</strong></td>
<td><strong>$881,650</strong></td>
</tr>
<tr>
<td>Break-even time in months</td>
<td>13.8</td>
<td>11.1</td>
<td>10.2</td>
<td>10.1</td>
</tr>
</tbody>
</table>

*Table 1: Single sign-on cost savings and ROI summary*

**Chapter 5: Summary**

Single sign-on has demonstrated that it can benefit almost any environment. The technology has evolved to now offer easy integration into existing applications and legacy systems, as well as to provide enhanced administration and security.

Due to the many varied infrastructures to which single sign-on may be applied, it is almost impossible to provide universally applicable advice about which solutions would be best suited for an average IT system. Various single sign-on options should be carefully evaluated within the context of your unique environment. As a basic rule, a successful single sign-on system should do the following:

- **Meet the changing needs of large organizations.** New software must be easily installed and configured for single sign-on.

- **Employ industry standards and open architecture.** The single sign-on system must be compatible with most existing software.

- **Easily accommodate mobile users.** Remote and roaming users must be able to access their single sign-on credentials and update them if necessary.

- **Provide ease of management, rapid deployment and high availability.** The single sign-on system must run efficiently, be easy for users to operate and be easy for you to control and maintain.

- **Be seamless to the user.** The second time a user logs in to an application, it should look the same as the first time. Subsequent attempts to open the application should involve no user interaction for authentication.

- **Be secure.** The password storage and playback mechanism must not allow for stealing secrets.
usernames and passwords must be encrypted and stored in a secure database.

- **Be cost-effective.** The single sign-on system must save money and reduce the cost of ownership.

- **Extend to strong authentication.** The single sign-on system should easily allow the addition of smart cards, authentication tokens and biometrics.

Single sign-on has long been touted as the ultimate solution for the growing number of login and password-management challenges. With mature technology that provides ease of use, cost savings and enhanced password security, single sign-on is finally posed to fulfill the hype we have heard about for years.