Global Information Assurance Certification Paper

SANS Security Essentials GSEC Practical Assignment
Version 1.2f (amended August 13, 2001)

Password Synchronization: Friend or Foe to the Security Professional?

By Robin Gwynne
February 12, 2002

Introduction

As a key component of an enterprise security architecture, the reliable authentication and authorization of users is of the utmost importance. For many organizations passwords are the only method of restricting access to corporate systems. While passwords are a mainstay of the overall security architecture, exclusive dependence on passwords does not provide much defence in depth, or diversity of defence. Therefore every attempt should be made to bring the organization into 100% compliance with a strong password policy. In a recent article in the online publication, Information Security, Peter Tippett writes, *"When it comes to strong passwords, anything less than 100 percent compliance is weak."*[1] While this may seem extreme, the old adage that a chain is only as strong as its weakest link is as applicable here, as it is throughout the rest of the enterprise security architecture.

As our IT organizations become increasingly complex, so too do our authentication and authorization systems. In the past six years alone at my previous employer, a typical employee has gone from having two or three ID's and passwords to six or more. In the case of some front line staff, they have to keep track of as many as ten. The natural end-user response to this proliferation of Id's and passwords is to simplify... The end-user quickly identifies the least common denominator of permitted passwords and manually synchronizes all their ID's to an easily remembered password that will be accepted by all systems. It only takes a couple of seldom-used systems to wreak havoc in this kind of environment, as they get missed in the manual password synchronization process. Often, this results in the user being unable to remember what their password was a few iterations earlier and/or locking the account out through unsuccessful attempts. Both of these situations generate calls to the help-desk.

One needs look no further than help-desk call statistics to see the ramifications of the growing number of ID's and passwords. In the previously mentioned organization, no fewer than 30 percent of calls made to the help-desk are password related. This is a huge operational impact on any medium to large sized corporation. A help-desk that is overloaded with requests for password resets and account unlocks is a prime target for social engineering attacks, as they struggle to meet response time expectations.

[1] Tippett, Peter

Management and system administrators do not sit idly by, they look for "quick win" solutions to help users cope with the plethora of authentications. Any number of approaches to ID/password simplification are available, including:

- Common ID names across systems and applications to make ID's easier to remember.
- Turn password expiry off on many systems so that ID's are not expiring on a daily basis.
- Reducing password strength to allow for easier to remember passwords.

All of these approaches to alleviating the burden of multiple ID's and passwords result in one thing: decreased security.

We have identified a number of security problems that can be directly or indirectly attributed to the multitude of ID's and passwords in today's heterogeneous environments:

- Weak password rules
- Social engineering of support staff
- Passwords that do not expire
- Common ID's and passwords for multiple systems
- Weak enforcement and low adoption rate of strong passwords

Scenario

A fictitious company, BigCorp, has a typical heterogeneous IT environment consisting of Microsoft Windows NT 4.0, SUN Solaris, and IBM OS/390. The various server platforms have the following security for ID's and passwords:

Microsoft Windows NT	ID's = same as RACF on OS/390 Minimum password length = 4 Passwords expire = Never Accounts not locked out for invalid login attempts
SUN Solaris	ID's = same as RACF on OS/390 Minimum password length = 6 Passwords expire = 45 days Accounts not locked out for invalid login attempts, but attempts are logged.
IBM OS/390	ID's = RACF, four alpha numeric characters Minimum password length = 4 Passwords expire = 45 days Accounts locked out after 3 consecutive invalid login attempts Last 7 passwords cannot be used.

In BigCorp we have a reasonably secure mainframe. The password strength rules are weak, but it is unlikely that anyone will successfully guess or brute force a password in three attempts. The RACF ID's are expired every 45 days and the last seven passwords cannot be used. The Unix environment has a slightly stronger password length and forces password changes every 45 days, but accounts are not locked for invalid attempts and is thus susceptible to password cracking attacks. With any luck, an administrator will notice this activity in the logs and take steps to eliminate the threat. Finally, we have the "Achilles Heal" of the corporation. As the platform for file and print sharing, the emphasis on Windows NT 4.0 has been convenience rather than security. Microsoft Windows NT 4.0 is notoriously weak in the area of passwords and BigCorp has made the situation worse by disabling password expiry and allowing four character passwords. The lack of any account locking makes this platform a prime candidate for a password cracking attack.

BigCorp feels relatively secure in the knowledge that their mission critical applications and financial systems are on the mainframe that is considered "secure". Unfortunately, the common ID's across platforms and high degree of manual synchronization means that a successful password crack under the NT environment will very likely result in successful access to the OS/390 and Solaris systems.

As the above scenario illustrates, it is easy to identify that there is a problem. Coming up with a solution that satisfies the requirements of diverse groups like Data Security, Enterprise Server Support, Desktop Support, Help Desk, and of course End Users, is no small task. One solution that is often proposed is to implement a password synchronization system. This paper discusses the main driving forces behind password synchronization as well as the benefits and risks of implementing password synchronization in a typical, heterogeneous environment.

Driving Forces

The main driving forces behind password synchronization are typically two-fold:

1. Reduce the number of password related Helpdesk calls
2. Increase security through increased password strength

As I mentioned above, in corporations with large numbers of ID's and passwords required by each user, the support staff deal with an ever-increasing number of password related problem calls. These problems are usually trivial, but in large numbers are costly to the corporation if it wants to maintain service levels. A reduction in problem calls, and a corresponding reduction in support staff are usually the main benefits used to cost justify any kind of password synchronization initiative.

Increased security is another key leveraging element of password synchronization. Users that are faced with too many ID's and passwords to remember, in an effort to be productive, will choose weak passwords so that they can remember them, or worse, write them down and stick them to the monitor or keyboard. To further simplify their authentication process, users will manually synchronize their passwords between systems so that they do not have to remember more than one. And finally, the help desk being overwhelmed by the number of password reset calls, is highly susceptible to "social engineering" attacks.

Password synchronization to the rescue?

As you will recall from our scenario, BigCorp has fairly widespread password synchronization already happening through a manual process, triggered by one of the key authentication credentials expiring. So, what does a full-featured password synchronization product provide? Quite a bit actually, the key features being:

- Enforcement of password strength rules consistently across systems.
- Real-time synchronization of passwords across participating systems.
- Self-Serve password recovery and/or reset.

There are essentially two ways that Password Synchronization products provide this functionality:

1. Require password changes be done through a standard interface that will control the synchronization process. This is a much less invasive approach but is susceptible to passwords getting out of synch through native password change utilities.
2. Redirect host password change processes to the password synchronization product for password rules enforcement, and management of synchronization process. This method is a more robust synchronization process, but requires changes on the participating systems.

The following benefits and concerns are typical of Password Synchronization products:

Benefits

- Consistent enforcement of password strength rules.
- Real-time synchronization of passwords across participating systems.
- Self-Serve password recovery and/or reset.

Concerns

- Security of all participating systems is only as good as the least secure system.
- Password change utilities vulnerable to attack.
- Passwords can still be cracked unless hosts identify and lock out intruders, and administrators ensure that authentication databases are secure.

Regardless which method is used, the goal is to reduce the number of passwords that need to be remembered so that password strength can be increased without an increase in support issues.

"Houston, we still have a problem"

Let's assume that we have selected our password synchronization system at BigCorp and implementation is proceeding. We have the product up and running on all of our major computing platforms and passwords are being kept in synch. We have set up a set of rules that enforce strong passwords for the host systems. Unfortunately, we are still only as secure as our least secure system. If an ID/PW pair is cracked on any system participating in password synchronization, the password can be reset, thereby gaining access to other systems in the environment. Given time and opportunity even the strongest passwords can be cracked through brute force.

Single Sign-On: An Improvement over Password Synchronization

Single Sign-On products are rapidly maturing to the point where they are a viable mechanism for securing multiple systems and applications with a single login. The technology usually implements a secure store that holds the ID's and passwords of the various systems that require authentication. This allows the SSO product to maintain its independence from the back-end systems and eliminates the need to synchronize passwords from system to system. Typical SSO products provide additional functionality, but with higher licensing and implementation costs. The following benefits and concerns are typical of SSO products:

Benefits

- Single Sign On for enterprise access
- ID's and passwords do not need to be synchronized across systems
- Maximum password strength can be enforced since users do not need to remember the majority of their passwords.
- Having a single ID and Password for central login allows increased SSO password strength.
- Passwords can be expired and reset much more frequently on the back-end systems than would normally be tolerated by end-users.
- SSO systems can be extended to include additional authentication types to deliver two-factor authentication.

Concerns

- SSO authentication must be the primary authentication mechanism.
- Security is only as strong as the SSO id and password pair.
- Implementation and licensing costs
- Interface support

Conclusion

In general, a good password synchronization product can be an excellent tool for moving a corporation towards strong password compliance, while driving down the support costs normally associated with strong passwords. There are however, drawbacks that are of particular concern to the IT security professional. Specifically, the overall security of the environment is reduced to that of the least secure system in the environment. As the security expert in an organization looking at deploying some form of password synchronization, you must be aware that, while password synchronization products are a useful tool for facilitating increased password strength, they do not increase the security of the native authentication service being synchronized.

To mitigate new risks resulting from enterprise wide password synchronization, any plan to provide password synchronization should also address these areas:

Policy: Password synchronization has security impacts across the enterprise and security policies should be reviewed to determine if those policies are still appropriate. If your organization does not have security policies in place, this would be as good a time as any to start creating them. At a minimum, the intentions of the password synchronization initiative will need to be documented.

Host and System Security: Any deficiencies in the authentication subsystems of each platform being synchronized must be identified and corrected if possible. In addition, each individual system or computing platform should provide some form of intrusion detection to prevent brute force password attacks. Locking accounts on repeated failed attempts is an easy way to combat this problem.

Finally, before deciding to move ahead with password synchronization, investigate the near term objectives and requirements that your organization may have. For corporations that are planning to implement some form of two-factor authentication, one of the more feature rich SSO products may provide a more appropriate solution.

References:

Tippett, Peter. "Stronger Passwords Aren't" Security for the CXO. June 2001. URL: http://www.infosecuritymag.com/articles/june01/columns_executive_view.shtml (10 Feb. 2002)

"P-Sync Whitepaper" M-Tech Mercury Information Technology, Inc. 2001. URL: http://www.psynch.com/docs/white.html (10 Feb. 2002)

Peltier, Thomas R. "Single Sign-On: Myth or Reality" Netigy. August 2000. URL: http://csrc.nist.gov/nissc/2000/proceedings/papers/303slide.pdf (10 Feb. 2002)

"Novell Single Sign-on takes the worry out of password administration" InTouch. December 1999. URL: http://www.softwarespectrum.com/intouch/edition11/novell.htm (10 Feb. 2002)

"The New Face of Single Sign-On" Network Computing. March 22, 1999. URL: http://www.networkcomputing.com/1006/1006f12.html (10 Feb. 2002)

Allen, Julia H. The CERT Guide to System and Network Security Practices. CERT, 2001.