Passwords Can Still Be Effective

Scott Magruder

The University of Southern Mississippi
118 College Dr. Box 5072, Hattiesburg, MS, 39406-0001
Phone: (601) 266-5099
james.magruder@usm.edu

ABSTRACT

This paper describes a method for making passwords more secure. Many experts indicate that the "time for passwords" has passed and other (more expensive) techniques should be used to authenticate users. However, as this paper shows, passwords can still be used effectively. Encrypting the combination of the password typed by the user with other information proves very effective. This requires a little more CPU time for logins but it is well worth the cost. Potential hackers who might try and break the password (without knowledge of the required "extra information" needed in the password), will never be able to break the password. In addition, the paper will describe using more than one encrypting method used for password protection. The concept of not letting the user name be known publicly will also be discussed.

INTRODUCTION

"Passwords are Useless, Outdated and a Security Risk" according to Cem Paya (Dignan). In 2000, "The Gartner Group suggests that the problem of lost passwords cost $1 million per year in an organization with 2500 desktop computers." (Gartner Group). If this is true, then why do passwords continue to be used to protect accounts and corporate data? The infrastructure for the password system is already available and is cheap to implement. In addition, the author of this paper contends that passwords work and continue to work. The software used in the login process may have to be modified, but the rewards for this modification will be better passwords.

A login and password pair are designed to protect something—an account; access to a network; access to a database, access to a website, etc. This protection is even more important as the Internet continues its global advance and more and more businesses have a web footprint for their organization. Hackers (crackers and script kiddies) try to get around the current security measures for an organization and system administrators try to prevent them from accessing areas they should not. The login/password pair allows employees access to organizational resources that they need to perform their job activities. The login/password pair is also meant to keep unauthorized access limited. This limitation of access can also apply to employees who do not need to access certain organizational resources. The following shows the potential access possibilities:

(1) Employees access resources for their job work
(2) Employees access resources outside of their required job needs
(3) Hackers (outside of the organization) attempt to access organizational resources

(1) and (3) of the above will be addressed by making passwords harder to break. (2) will be addressed briefly--the inside hacker already has an account and is closer to the organizational resources than the outside hacker.

Login/password pairs

To login to an organizational asset, a login name and associated password must be used. One of the original assumptions about logging into an organizational asset was that the login name may be (or is probably) public knowledge. This system was designed in a different era than what we are seeing today. Making login names non-public increases the effort needed to break into these resources.

Any protocol (such as telnet) that sends the login name and password in plain text over the network should not be used. It only takes one time of use, when the network is being sniffed, for the login name and password to be grabbed. This paper assumes an encrypted protocol (such as SSL) is used when an employee attempts to login to a network or other organizational resource. That leaves the local computer (where a keystroke logger may reside) and the server (where the login/password pair is checked for authenticity) where hacking attacks may occur.

Passwords should be between 8 and 16 characters (or more). The larger the password (all other things the same) the harder the password is to crack. The passwords should be a random assortment of characters (upper and lower case), punctuation and some special characters. These passwords are “harder” to break. The following continuum shows the possibilities:

|----------|
| |
Dictionary words Random characters

Totally random characters are hard to remember and so are often written down. This is ok as long as the written password is secure. Dictionary words are easy to remember and easy to break. This is where passwords fail—users choose easy to remember passwords. These passwords are also easy to crack.

In addition, password cracking software has increased significantly in its ability to crack "bad" passwords. Cain & Able will be used in the examples in this paper. "Cain & Abel is a password recovery tool for Microsoft Operating Systems." (Cain & Able). Cain & Able has several additional tools to try and break passwords (sniffing the network, etc.) There are password crackers for other operating systems as well.

One approach to making passwords harder to break is to use the same encryption process multiple times. Magruder, Lewis and Burks describe this process. If a hacker gains access to the password file (or physical access to the computer), the passwords cannot be cracked as the hacker is not using the correct number of encryption iterations (Magruder, Lewis and Burks). This approach presented here is similar, except different encryption techniques are used rather than the same one in a loop.

LOGIN PROCESS

The actual passwords are not stored on the server that a user might want to log onto. Rather, an encryption technique is used to encrypt the password. The encrypted password is stored in the password file along with an associated login name. The login name is saved as plain text. This is a problem. If a hacker gains access to the password file, then they will know what password is associated with a given login name. Thus, one approach to making passwords harder to crack is to encrypt the login names also. This will require more CPU time on the server, but makes the accounts more secure.

The current login process can be stated as the following process:

(1) Read the user's login name;
(2) Read the user's password;
(3) Using the defined encryption technique on the server, encrypt the user given password;
(4) Compare the encrypted password with the encrypted password stored in the password file;
(5) If the encrypted passwords match and the user name is correct, the user is allowed onto the system;

The third step is where the major change presented in this paper appears. Instead of "encrypt the user password;" the new process (replacement step) is:

Encrypt the user given password; encrypt the encrypted password with a different encryption technique;

When the user logs in, this “replacement step” would be used in place of the original step 3 above.

BREAKING PASSWORDS

To break passwords, a hacker must get access to the password file and know what encryption technique(s) is used on the given server. The hacker must somehow download the password file and try to crack the passwords off the server as the administrator would notice the intensive CPU time used on the server cracking the passwords.

Assume we have a password “dictionary”. Using the hash library of python, this encrypts (crypt for Linux) to “rtOJd5gYZj5qw”. The first two characters are the “salt” which is used to add two characters to the encrypted password, making it more secure. Because this is a dictionary word, it is easily cracked. However, if we now encrypt this encrypted password as an MD5 hash, the following is produced:
9d4545912fc6c94d5bfabe6456416653
Cain & Able was unable to break this MD5 hash. However, even if it or another password decrypting software were able to break this password, the process would have to be performed again to break the first encryption. This is what makes this approach better. Even if the passwords are eventually broken, the administrator has more time to discover the intrusion and correct the situation. The logs for the server play an important part in discovering the intrusion.

OTHER REPLACEMENT STEPS

In the original login process, step (3) was replaced with two encryption techniques. Other options will be discussed next. The “salt” added by a Linux server has already been discussed. Two additional characters are added to the password to make it longer and thus, harder to break. One problem with this is that the plain text salt is stored with the encrypted password. A better solution would be to save the salt in a separate file. The Shadow Suite breaks the password file up into two files, making it harder for the hacker to get the password information.

Assuming the login name is not public, creating an encrypted password of the login name and password that the user submits is a good alternative. Again, for all these suggestions to work, the system has to set up the user account in this manner i.e., with two encrypt techniques, etc. When the user logs on the server, their credentials must be compared with the result (saved in the password file) which uses the same process.

Appending the salt or other information (that the user nor hacker knows is being done) saved in a separate file from the password file will make the passwords harder to break. Adding this information makes good passwords which are already hard to break, better. It will also make bad passwords (easily cracked) much harder to crack.

OTHER WAYS FOR HACKERS TO GET PASSWORDS

There are other ways that hackers might get access to the user's credentials: tricking the user to supply the information via phishing techniques or fake login screens, etc. and downloading a keystroke logger onto the user's computer. Especially in a business environment, education is the best solution to the first problem. Setting up good firewalls and individual computer anti-virus systems may help the second problem.

INSIDE HACKERS

Employees have the first thing a hacker wants to get—an account on the system. Once the hacker gets this, they want to elevate their privileges on the system. Employees may try to elevate their privileges or go to areas in the system that they do not need access to in performing their jobs. To allow employees to sensitive areas on the organization's site, a separate login could be required. As all logins are recorded in the server logs, these could be periodically checked to see if any employees are accessing areas of the system that they should not. Short of this, sensitive area access could be recorded in the server logs and reviewed periodically.

Disgruntled workers also present a problem to the organization. These are workers who are causing problems for the organization or know they are about to be fired. If these employees are known to the organization, then they need to be kept away from the organization's system. Their accounts should be disabled. The employee may have multiple accounts. All of their accounts must be disabled—the passwords can be changed to ensure they cannot get back into the system.

CONCLUSIONS

The primary objective of this paper has been to show that passwords can still be used effectively. The software used to login a user may have to be modified. However, the results will be better, harder to break passwords. Passwords are made more effective by using more than one encryption technique to encrypt the passwords. In addition, adding additional characters to the encryption process makes the passwords more secure. These additional characters should be saved in a file separate from the password file. Encrypting the login name and password was also discussed. An important aspect of this process is to not make the login name public. This makes it harder for the hacker to break into the user's account.

Employees with accounts on the system already are also a problem for the organization. If necessary, monitoring of certain employees by the organization may be required.

REFERENCES

Cain & Able, http://www.oxid.it/cain.html.

Dignan, Larry, (2009). Passwords are Useless, Outdated and a Security Risk – Cem Paya, Delfigo Security. http://www.delfigosecurity.com/iamblog/password/Passwords-Are_Useless-Outdated-and-a Security-Risk-Cem-Paya, downloaded December 14, 2009.

Gartner Group, (2000). Article: Outdated, insecure passwords are losing money for Internet businesses. (Gartner Group report)(Industry Trend or Event). Article from: Communications News, February 1, 2000. COPYRIGHT 2000 Nelson Publishing. This material is published under license from the publisher through the Gale Group.

Magruder, Scott, Stanley X. Lewis, Jr., and Eddy J. Burks, More Secure Passwords, Journal of International Technology and Information Management, Volume 16, Number 1, 2007, p.87-96.