Global Information Assurance Certification Paper

Password Protection: Is This the Best We Can Do?

Jason Mortensen
August 2001

GSEC Practical Assignment Version 1.2e
Resubmission

© SANS Institute 2000 - 2005 Author retains full rights.

More often than not, the last barrier between the "outside world" and most computer systems is some kind of password authentication. While passwords are practically ubiquitous in modern computer systems, numerous deficiencies associated with passwords present a critical challenge to network security professionals. If an attacker is able to determine a valid username and password to a computer system, they will be able to impersonate the valid user and access the system. Since valid credentials are presented, these intrusions often go unnoticed.

There are numerous problems that can make password authentication a poor line of defense, including weak passwords, improper password storage, and passwords that are captured by eavesdropping on network traffic. These problems can lead to unauthorized access of computer systems and potentially the compromise of important data.

Weak Passwords

Weak passwords are a classic way in which computers are compromised. Attackers often attempt to exploit weak passwords as a first step towards obtaining unauthorized access to a system.

Weak passwords come in many forms. These include null passwords, passwords that are guessable, and default passwords.

Null passwords. Null passwords, also known as empty passwords, represent the ultimate in user convenience. Null passwords occur when users choose not to use a password by simply hitting "enter" when setting their password. While null passwords are extremely convenient for users, they represent a critical security problem.

Guessable passwords. Many users do not want to remember, or they have difficulty remembering complex passwords. As a result, many users choose simple passwords that are easy to remember, such as the word "password" or "secret", or the name of a family member. An attacker, however, may easily guess these simple passwords.

One recent study conducted by CentralNic [1] suggested that users choose passwords based on one of four criteria:

- "Family" – Names and nicknames of family members or pets. Almost half of the people that were polled in the study fell into this category.
- "Fan" – Sports stars, cartoon characters, or pop stars. This group makes up about a third of the people polled.
- "Self obsessed" – About 11 percent of the people polled used passwords like "sexy," "stud," or "goddess."
- "Cryptics" – About 9 percent used complex passwords, specifically chosen to enhance security.

Other guessable passwords include using the login name as the password (sometimes reversed, just to be "tricky"), or the company name in corporate environments.

Below is a list of "guessable" passwords that are among the most commonly used:

password	sesame	changeme
secret	sex	qwerty
money	pass	abc123
private	admin	123456
god	hello	111111

Password-based computer systems are usually susceptible to dictionary attacks, where automated login attempts are made to guess the password by using words from a dictionary. This is particularly the case when users are allowed to choose weak, guessable passwords. Electronic dictionaries exist for a variety of languages, including English, Spanish, French, German, Chinese, Japanese, Russian, and even Klingon! Dictionaries also exist containing words from TV shows, movies, music, works of literature, sports, and numerous hobbies. Password guessing programs such as the NetBIOS Auditing Tool (NAT) [2] and Brutus [3] can use these password dictionaries in attempt to determine weak passwords. NAT can be used to automate password guessing of Windows NetBIOS passwords, while Brutus can be used to guess passwords for numerous other protocols such as HTTP, Telnet, FTP, and POP3.

Default Passwords. Default passwords are another easy way that attackers can access systems. Many software packages and network devices are installed with "default" passwords that are often left unchanged when the system is configured. For example, some 3Com networking devices use the default password of "manager" and these passwords often remain unchanged by network administrators. Lists of default username/password combinations can be found on the Internet, and knowledge of these accounts can provide attackers an easy way to access systems. The popular web site Slashdot (http://www.slashdot.org/) had a problem in September 2000 where a default password led to a compromise of the system [4].

Below is a sample list of "default" passwords that are used on networking devices. This list is based on information from the book Hacking Exposed [5].

Device	Username	Password
Bay Router	User	<null>
	Manager	<null>
Bay 350T Switch	NetICs	NA
Bay Superstack II	security	security
3Com	admin	synnet
	read	synnet
	write	synnet

	debug	synnet
	tech	tech
	monitor	monitor
	manager	manager
	security	security
Cisco	(telnet)	c (Cisco 2600s)
	(telnet)	cisco
	enable	cisco
	(telnet)	cisco routers
Shiva	root	<null>
	Guest	<null>
Webramp	wramp	trancell
Motorola CableRouter	cablecom	router

Improper Password Storage

Some users have a hard time remembering their passwords, and as a result, they tend to write their passwords down. This can be especially problematic when users are required to choose complex passwords. Some users even resort to writing down their passwords on small notes that they attach to their monitors!

Aside from users writing down their passwords, some software programs store passwords in a manner that the passwords can be easily retrieved. This is especially the case with software programs that offer to "remember" user passwords so that users don't need to remember them. These passwords are sometimes stored in files in an encoded or encrypted format, while other times they are stored in clear-text. Experience has shown that the methods used to encode or encrypt passwords are often weak and can lead to an easy compromise of passwords. Earlier versions of Netscape, for example, stored email passwords in the prefs.js file (the preferences.js file under Unix flavors) using a simple Base64 encoding. Although the passwords were obscured, an attacker could easily obtain the clear-text value from the file [6]. Other programs, such as the Caesar FTP server, store passwords in clear-text in files [7].

Operating systems themselves may implement a weak password storage scheme as well. For example, older versions of Unix stored user passwords in an encrypted format in the world-readable text file called /etc/passwd. Using a program like Crack by Alec Muffett [8], password guesses can be made to determine the clear-text values of user passwords. Crack uses a technique called "password cracking", where password guesses are encrypted using the algorithm that Unix uses to store passwords, then the output is compared to the password strings that are stored in the /etc/passwd file. If the encrypted password strings match, then the password guess must be correct. Newer versions of Unix solve this problem using a technique called "password shadowing", where the encrypted password strings for each user are stored in a separate "shadow" file that is readable only by the "root" (super-user) account.

Windows NT is also vulnerable to a similar password cracking attack. Under Windows

NT, passwords are stored in a database called the SAM database. Quite often, a backup copy of this database can be found in the c:\winnt\system32\repair directory, with read permissions given to the Everyone group (anyone with access to the system). An attacker can utilize a program such as L0phtCrack by the L0pht Heavy Industries group [9] to crack Windows passwords in much the same way that Crack works under Unix. However, Windows suffers from two major design flaws that makes password cracking easier than with Unix. First, Windows separates passwords into two parts, each part having seven characters. The two parts of the password are then run through a hash (encrypting) function to hide the clear-text value. For an attacker, this means that instead of cracking one long password, at the most they only have to crack two separate seven-character password strings, which is much easier. The second design flaw relates to the fact that Windows does not use a "salt" value to make each hash value random and different. Any users that may be using the same password will also have the same encrypted password strings stored by the system. To prevent Windows NT password cracking, Windows users can utilize a program called syskey to prevent attackers from obtaining the encrypted password strings. Syskey works by providing strong encryption of the SAM database itself [10].

Clear-text "sniffable" passwords

Many of the protocols in use on the Internet were not designed with security in mind. Some protocols, such as Telnet, FTP, and POP3 transmit passwords across the network in clear text. Other protocols, like HTTP, employ only a simple Base64 encoding before transmitting passwords. Attackers often "listen" to network traffic, eavesdropping on the traffic hoping to "sniff" passwords that are transmitted in clear-text.

A tool that makes the process of password sniffing easy is called dsniff, written by Dug Song [11]. The dsniff tool can listen to network traffic and automatically pick out usernames and passwords. The result is a convenient report that displays username and password pairs that are captured from network traffic, as shown below.

```
-----------------
08/11/01 08:59:24 tcp 192.168.22.158.1072 -> 192.168.10.150.21
(ftp)
USER fjones
PASS n0tepad

-----------------
08/11/01 09:01:30 tcp 192.168.22.158.1074 -> 192.168.10.150.23
(telnet)
john
secret

-----------------
08/11/01 09:05:33 udp 192.168.22.103.1028 -> 192.168.230.193.161
(snmp)
[version 1]
```

```
public

-----------------
08/11/01 09:12:37 tcp 192.168.22.21.1195 -> 192.168.3.48.80 (http)
GET /admin/ HTTP/1.1
Host: www.example.com:80
Authorization: Basic YWRtaW46bHVtYjNyakBjaw== [admin:lumb3rj@ck]

-----------------
08/11/01 09:16:20 tcp 192.168.22.48.61583 -> 192.168.6.153.110
(pop)
USER markham
PASS letmein
```

In addition to "sniffable" passwords, some systems may be susceptible to a password replay attack. Windows 95 and 98 are susceptible to such an attack. In this particular scenario, an attacker can "sniff" the challenge-response sequence of a valid login between Windows NT and Windows 95/98 and can replay this string to gain access to the services on the Windows 95/98 computer. Knowledge of the clear-text password is not needed for this attack to work [5].

Solutions to Bad Password Systems

There are several things that can be done to protect computer systems from problems related to passwords. Some of these solutions rely on better password policies while other solutions require mechanisms that enhance or completely replace password systems.

User Education. Many users are not aware of the security problems associated with passwords. Through educational programs, users can be taught the importance of selecting strong, secure passwords. Many corporations have education programs that are designed to inform users of the importance of choosing strong passwords. Users can be reminded through newsletters, email, posters, and flyers of the importance of password security. Users should be taught how to select strong passwords, and why it is important to keep their passwords secret.

Enforceable Password Policies. Some computer systems offer integrated mechanisms that enforce the use of strong passwords. Windows NT, for example, includes a file (in service pack 2 and later) called passfilt.dll [12]. This file can be installed and used to enforce rules on allowable passwords. Unix also offers utilities that enforce strong password policies when setting passwords. Passwords are more secure when users are required to have a password six characters or longer, use numbers and special characters, and aren't able to reuse passwords. Security is also enhanced when users are required to change their passwords on a regular basis.

Encrypted Network Traffic. Even if users select strong passwords, there is always the

possibility that passwords can be "sniffed" from the network. Encrypting network traffic can prevent this type of attack. Several technologies exist for encrypting network traffic, such as Secure Sockets Layer (SSL). Using SSL, an encrypted "tunnel" can be established to transport other protocols, such as HTTP, POP3, and LDAP. Other protocols, such as Telnet, can be replaced with encrypted protocols such as SSH, the Secure Shell.

One Time Passwords. In a one-time password system, it doesn't matter if a password is captured from the network, since the password is valid for one login only. A popular system for implementing one-time passwords is the SecurID system from RSA Security [13]. Users are issued a SecurID "authenticator" that generates a one-time passcode every sixty seconds. SecurID authenticators use a patented RSA algorithm that combines a seed value with Universal Coordinated Time (UCT) to generate a psuedo-random number. When logging into a system, a user enters a PIN that they have chosen, along with the psuedo-random numeric value that is displayed on the authenticator at that moment. An authentication server on the back end called an "ACE Server" serves to verify the value entered by the user. The ACE Server knows the seed value, UTC, and PIN for each user, and should be able to calculate the same psuedo-random number that the user presents. When using the SecurID system, the combination of something you know (your PIN) and something you have (the number on your authenticator) provides a very strong two-factor authentication into computer systems [14]. Below is an example of a SecurID hardware authenticator.

Public-Key Infrastructure. Using a Public Key Infrastructure system, users are issued a "digital certificate" that is used to represent identity. PKI technology is based on public-key cryptography, where each user possesses a "key pair" made up of two related but different cryptographic keys. One of these keys is designated as a private key, and the other key is designated as a public key. A certificate consists of the users' public key, information about the user, and a digital signature from a certificate authority (CA) that has verified the authenticity of the individual in question. PKI systems are designed to establish a strong level of trust, and thus they are an excellent choice for authenticating users to computer systems. Using the cryptographic technology that PKI is built on, users can present their digital certificate as authentication to a PKI-enabled system and establish a strong authentication to that system [15].

Smart card technology can be used to support PKI by storing and protecting digital certificates that are used by end-users. Smart cards look much like common credit cards, except that they have a small microchip embedded in them. Smart cards allow PKI users to be mobile, since certificates are stored on smart cards that can easily be carried around.

Smart cards are also ideal for storage of private keys, since information stored on the cards cannot be copied [16].

Biometric Systems. Biometric systems employ technology to identify some physical characteristic as the basis of authentication. Popular biometric systems include fingerprint scanning, retinal scanning, voice recognition, and facial recognition technology. Since biometric systems rely on specific physical characteristics of an individual, they can offer a high level of assurance that individuals are who they claim to be. Biometric systems are often used in connection with another authentication method, such as PKI [17, 18].

Although biometric systems are designed to strongly authenticate individuals, there are several factors that prevent biometrics from wide acceptance. Cost is one of the major disadvantages of biometric systems, since biometric devices tend to be rather expensive. Interoperability is another problem, because biometric products are often based on proprietary standards [18].

Conclusion

There are numerous problems associated with the use of passwords on computer systems, including the use of weak passwords, improper password storage, and passwords that are vulnerable to eavesdropping. In spite of these problems, password systems can either be enhanced or replaced to provide stronger authentication into computer systems. Through a combination of user education, strict password policies, encrypted network traffic, one-time passwords, Public-Key Infrastructure systems, and the use of biometrics, authentication into computer systems can be made quite strong.

References

[1] McAuliffe, Wendy. "Computer Passwords Reveal Workers' Secrets."
URL: http://www.zdnet.com/zdnn/stories/news/0,4586,2781327,00.html (June 29, 2001).

[2] NetBIOS Auditing Tool.
URL: http://packetstormsecurity.org/UNIX/utilities/nat10_tar.gz (August 2001)

[3] Brutus web site.
URL: http://www.hoobie.net/brutus/ (August 2001)

[4] Malda, Rob (CmdrTaco). "Yup, Somebody Cracked Slashdot."
URL: http://slashdot.org/articles/00/09/29/1245218.shtml (September 29, 2000)

[5] Scambray, Mclure, and Kurtz. "Hacking Exposed, Second Edition." © 2001 Osborn/McGraw-Hill, Berkeley, CA.

[6] Netscape Password Storage (email thread from Bugtraq).
URL: http://www.packetstormsecurity.com/new-exploits/ns4.5-mail-passwd.txt (November 1998)

[7] "CaesarFTP Plaintext Password Storage Vulnerability."
URL: http://www.securityfocus.com/bid/2785 (May 27, 2001)

[8] Crack (Password Guessing Program for Unix).
URL: ftp://ftp.cerias.purdue.edu/pub/tools/unix/pwdutils/crack/ (June 2000)

[9] L0phtCrack web site.
URL: http://www.atstake.com/research/lc3/index.html (August 2001)

[10] Edwards, Mark Joseph and LeBlanc, David. "Where NT Stores Passwords."
URL: http://www.win2000mag.com/Articles/Index.cfm?ArticleID=5705 (August 1999)

[11] Song, Dug. Dsniff web site.
URL: http://www.monkey.org/~dugsong/dsniff/ (August 2001)

[12] Microsoft Corporation. PassFilt.dll web site.
URL: http://msdn.microsoft.com/library/en-us/security/hh/logauth/pswd_about_9x7w.asp

[13] RSA Security. RSA SecurID web site.
URL: http://www.rsa.com/products/securid/ (August 2001)

[14] RSA Security. "Two Factor Authentication for an e-Business World."
URL: http://www.rsa.com/products/securid/brochures/SID_BR_0999.pdf (1999)

[15] Adams, Lloyd, Kent. "Understanding Public-Key Infrastructure." © 1999. Macmillian Technical Publishing, Indianapolis, IN.

[16] GEMPLUS – All about Smart Cards web site.
URL: http://www.gemplus.com/basics/what.htm (August 2001)
URL: http://www.gemplus.com/basics/why.htm (August 2001)

[17] Ashbourn, Julian. "The Biometric Whitepaper."
URL: http://homepage.ntlworld.com/avanti/whitepaper.htm (1999)

[18] Harreld, Heather. "Biometrics Points to Greater Security."
URL: http://www.fcw.com/fcw/articles/1999/FCW_071999_799.asp (July 19, 1999)