An Analysis of Local Security Authority Subsystem & Extracting Password Using Packet Analyzer

Bhanu Sharma
Computer Science & Engineering Department
BBSBEC, Fatehgarh sahib(Punjab) India

Sandeep Kaur Dhanda
Computer Science & Engineering Department
BBSBEC, Fatehgarh sahib(Punjab) India

Abstract: User authentication in computer systems has been the major objective of computer security for decades. Local Security authority Subsystem Service (LSASS) is one of the most important concerns when it comes to a computer's security with the responsibility of processing authentication requests & allowing or denying access to user's requests, generates access tokens, manages the local security policy & handles other windows security mechanisms. Now, one of the most important challenges to be faced here is what if the SAM file is attacked? As it is the heart of a fundamental security infrastructure, with the foremost priority of restricting unauthorized users to access the system. The major objective in this paper is to make a computer system more secure but we are simultaneously analyzing how to break the security of the system . The main goal is to create a protection scheme that can't be bypassed by any software. In today's time, there are various security software available in the market and out of them there are some which cracks the login password. But ours is to provide the best solution considering the parameters likes time dependency, SAM database, password length, design strategies etc. Further we will use local security authority subsystem extract password using packet analyzer tool. Then we will compare the present result with the previous one.

KEYWORDS - Security, Malware, authenticates, directory, SAM, LSA, LSASS, Syskey, Time dependency, CPU.

I. INTRODUCTION

Local Security Authority Subsystem Service (LSASS) provides an interface for managing local security, domain authentication, and Active Directory processes. It handles authentication for the client and for the server. Local security authority subsystem service process is called upon to store the hashes in memory. Hashes Means

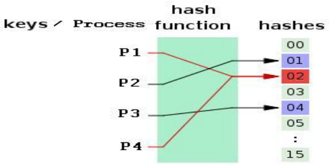


Figure No.1 hash function

If a password is used that consists of 14 or less characters both hashes will be created and stored in the local security account manager file or in active directory. The time to crack a password is related to bit strength which is a measure of the password's information entropy. Most methods of password cracking require the computer to produce many candidate passwords, each of which is checked. One example is brute-force cracking, in which a computer tries every possible key or password until it succeeds. Attempting to guess the correct secret by trying all, or a chosen subset of all, possible options. Brute Force attack is method of breaking a cipher by trying every possible key. Cipher Means is an algorithm for performing encryption or decryption. Feasibility of brute force attack depends on the key length of the cipher, and on the amount of computational power available to the attacker.

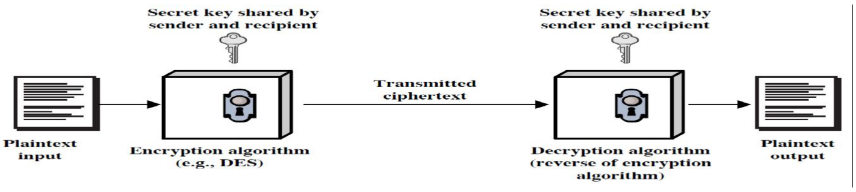

Secret key shared by sender and recipient
Secret key shared by sender and recipient
Transmitted ciphertext
Plaintext input
Encryption algorithm (e.g., DES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

Figure No.2 Transmitted Cipher Text

- The key space of all possible combination of passwords to try is calculated using the following formula:

  KS = L^(m) + L^(m+1) + L^(m+2) + ... + L^ (M)
- Try every key to decipher the ciphertext.
- On average, need to try half of all possible keys
- Time needed proportional to size of key space

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/µs	Time required at 10^6 decryptions/µs
32	2^{32} = 4.3 x 10^{9}	2^{31} µs = 35.8 minutes	2.15 milliseconds
56	2^{56} = 7.2 x 10^{16}	2^{55} µs = 1142 years	10.01 hours
128	2^{128} = 3.4 x 10^{38}	2^{127} µs = 5.4 x10^{24} years	5.4 x 10^{18} years
168	2^{168} = 3.7 x 10^{50}	2^{167} µs = 5.9 x 10^{36} years	5.9 x 10^{30} years
26 characters (permutation)	26! = 4 x 10^{26}	2 x 10^{26} µs = 6.4 x10^{12} years	6.4 x 10^{6} years

Table No. 1 Key Size Details

The Main objectives are Understand authentication and authorization, to get familiar with how passwords are used, to study why password policies are essential and classify various password recovery techniques.

II. PROBLEM FORMULATION

Our goal the login window authentication is not sufficient which is very easily breakable so we apply more passwords before login window, which will not be breakable, and nobody can break that password. The OS designer is to create a protection scheme that cannot be bypassed by any software that might be created in the future. Not really possible, if a problem is solvable with software, then programmers can eventually write a program to do it. We are also working on password extracting technique any user can easily bypass the login window by using some software tools but they can not able to fetch the exact password. So we are trying to extract the exact password of the login window by using brute force technique with using packet analyzer tool.

III. EXISTING TOOLS AND TECHNIQUES

The following is a detailed analysis of common tools and techniques used to capture Windows credentials. The most basic method is keylogging. User level keylogging is usually ineffective at capturing Windows credentials due to the privilege level of the logon processes. However, keyloggers running at the kernel level are capable of capturing Windows logins. Account credentials can also be compromised by replacing the Microsoft Graphical Identification and Authentication Dynamic Link Library (MSGINA). Passwords can also be obtained by cracking their equivalent password hash. The different types of password hashes on Windows systems, LM and NTLM, will be briefly discussed. There are various methods to obtain password hashes. The most straightforward way is to extract them from the local registry. However, the local registry will only contain login information for local accounts and not domain accounts. Hashes can also be extracted from the lsass process. Pwdump is one tool capable of extracting hashes from lsass. Once hashes are obtained, they can either be cracked or utilized directly in pass-the-hash tools. Passwords can be cracked using a variety of methods (Fischer, 2007). These methods, including password lists, GPU cracking, and rainbow tables will be explained in detail. Pass-the-hash tools will also be examined. These are effective even when very strong passwords are used and the hashes cannot be broken. A less commonly known technique for compromising credentials, hooking Windows authentication functions, will be examined. The majority of these methods require local administrator access. It will be assumed that the attacker has already achieved local administrator credentials before utilizing these tools. As stated in the introduction, there are many methods to obtain local administrator access. The different password resetting tools are presented.

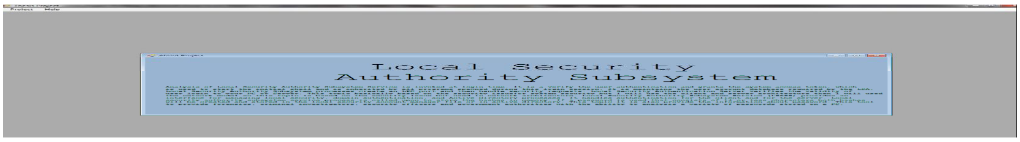
Tool	Short description	Operating System	Objective	Prerequisites
Tool 1	PCLoginNow 2.0.1	Windows XP, Windows Vista, Windows Server 2008, Windows 7	Password Resetting	Empty Writable CD
Tool 2	Offline NT Password & Registry Editor	Windows XP, Windows Vista, Windows Server 2008, Windows 7 RC	Password Resetting	Empty Writable CD
Tool 3.0	BartPE	All Windows systems	System access	Empty Writable CD

Tool 3.1	Password Renew	Windows XP, Windows Vista, Windows 7RC	Password Resetting	Included on for example the BartPE CD
Tool 3.2	WindowsGate	Windows XP, Windows Vista, Windows Server 2008, Windows 7 RC	Password Resetting	Included on for example the BartPE CD
Tool 4	Kon-Boot	Windows XP, Windows Vista, Windows Server 2008, Ubuntu Server 8.10	Bypass password authentication	Empty Writable CD
Tool 5	DreamPackPL	Windows XP	Password Resetting	Empty Writable CD, Windows XP Installation CD
Tool 5	DreamPackPL	Windows XP	Password Resetting	Empty Writable CD, Windows XP Installation CD
Tool 7	John the Ripper 1.7.2	Windows XP, Linux systems with hash values Generated with crypt (only some algorithms)	Password Recovery	Empty Writable CD
Tool 8	Cain & Abel	Windows XP, Windows Vista, Windows Server 2008, Windows 7 RC	Password Recovery	Password and System file, Rainbow tables
Tool 9	LCP 5.04	Windows XP, Windows Vista, Windows 2008 server, Windows 7 RC	Password Recovery	Password file

Table No. 2 Resetting tools

IV. IMPLEMENTATION

We have developed small tool according to the requirement. We have provided the more security of computer system so we have developed one tool for providing the security of the computer system. This tool is made in Vb.net programming language. This is the experimental concept of the security. There are many tools are available in the market which are already mention above but we have used in this thesis we have to design and implementing the security concepts according to the requirement. This tool based on the Syskey concept. We have already discussed in above about Syskey.This Screen Shots are show the experimental concept and Discussion about project.



Local Security Authority Subsystem

Figure No. 3 Show About Project

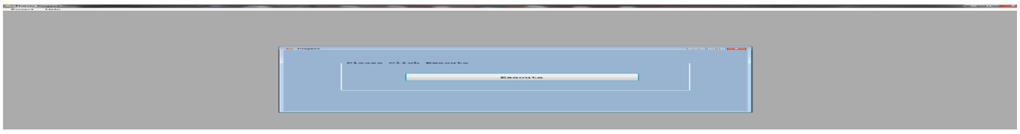


Figure No. 4 Show How to Run Project

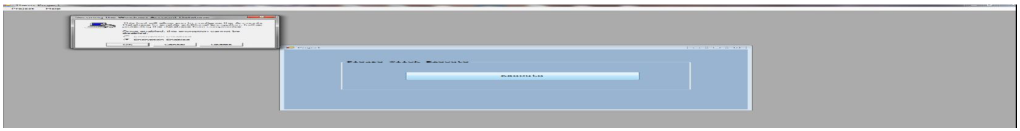


Figure No. 5 Show the Steps

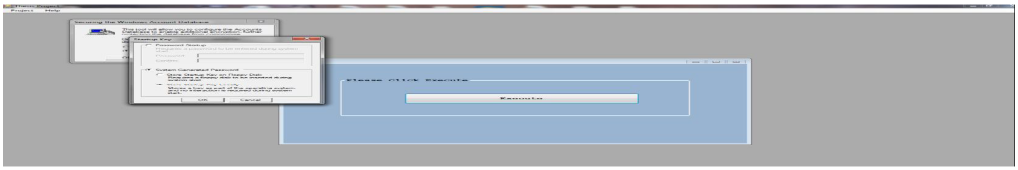

Project
Help
Securing the Windows Account Database
Startup Key
Password Startup
Password
Confirm
System Generated Password
Store Startup Key on Floppy Disk
OK
Cancel
Please Click Execute
Execute

Figure No. 6 Show Set the Password

V. TEST CASES AND RESULTS

We divided the experiment into two different phases. These two phases are called as phase 1 and phase 2. Phase 1 is basically like a timing phase. Here we can have an idea that it is beneficial that when we run rainbow table to crack the password, before this process we have to run a dictionary attack. The second one phase that is phase 2 is the main phase for cracking process. In this phase the main cracking process was happened. The both phases get completed with the three different types of hashed chosen for this experiment. The three hashes are LM hash , NT hash and MDs hash. We analyze that in which technique the process will work, it may be dictionary attack and / or cryptanalysis attack. The above technique can be used for both phases phase 1 and phase 2 for their hash type. We used a wordlist, which is in build in Cain and Abel tool for all the dictionary attacks. When john the ripper was used for the dictionary attack, at the same time the same wordlist was imported. It we talk about rainbow tables, they are only used for LM & NT hashes. Here we used two types of different rainbow tables. The rainbow table consists of 64 sub tables and they have 64 GB size. They are specially designed to crack the LM hashes. In our experiment, we added an extra hard disk to store these rainbow tables. We imported the LM hashes into the Cain & Abel tool. To understand which characters these rainbows table's support we had to include a character set in Cain and Abel tool. The main goal of our experiment is to check the password of different strength. So we followed a scenario. We created three different password groups. After that we added 10 passwords in each group. There are two main and famous groups that are mnemonic password & random password. We took 10 passwords from Bruce Schneier and added to the most popular group. We tool mnemonic password from mnemonic generator. Out of them we create two. We used random password generator to generate the random password, which can be easily found in the Internet some random password are considered as very secure password. The reason behind it that they are generated randomly in the combination of lower case character, upper case disadvantage of random password that they are very tough to remember they have some tough sequence of character password the mnemonic password are easy to remember because they are made by user by selecting some phases that are memorable and can be easily recall. These passwords seem like random generated password but they are very easy to remember because they are a part of phrase. We have already discussed that we will create 2 of the mnemonic password. Which is in 9 or 10 character in length. This is required to test the password having different characteristics. The phase 1 is basically a timing phase. Here we analyzed how much time dictionary attack and crypto analysis attack will take to crack the selected password of the LM hash & NT hash, another thing was analyzed in this phase that was the time of the dictionary attack on the MD5 hash.

Technique	Time
Dictionary attack	1 minute 10 seconds (70 seconds)
Cryptanalysis attack	1 hour 1 minute 10 seconds (3662.36 seconds)
Total	1 hour 2 minutes 20 seconds (3750 seconds)

Table No. 3 Time measurements

Phase 2 is the main phase where the password is cracked. We created 30 users with different passwords and we wanted to examine that how many password out of 30 can be cracked in a working day. This process will be dome for all 3-hash process. A full image of the selected password can be found either in the table no 6.3 or from the result which are shown below. When the phase 2 will be completed then we will be able to take a comparison between the phase 1 and phase 2 to see how long time the cracking process would take to crack all 30 passwords, after cracking of one password. This is the main difference between the both processes. There is another difference between both phase that in phase 2 we used full set of rainbow tables for both the LM hash and NT hash. The results presented above show that 8 passwords were revealed during the dictionary attack and 10 passwords were revealed during the cryptanalysis attack, meaning that 18 of the 30 selected passwords were revealed in total. Lists all the 30 passwords that were revealed. The user name, length of the password and to which category the password belongs is also included in the table. The 18 passwords that were found are listed with Yes in the Revealed? Column. Note that each row in the table containing a password revealed during the dictionary attack has italic fonts. This means that the passwords revealed during the cryptanalysis attack is marked with a Yes in the Revealed? Column but do not have their font in italic. 12 of the 30 passwords were not revealed. The 27% of the 30 selected passwords were revealed during the dictionary attack while 33% of the passwords were revealed during the cryptanalysis attack. This means that 40% of the NT hash values were not cracked.

Technique	Passwords Revealed
Dictionary attack	8
Cryptanalysis attack	10
Total	18

Table No. 4 Revealed passwords

VI. CONCLUSIONS AND FUTURE WORK

There is advantage and disadvantage in every Security Concepts .The fundamental concept is based on local security design strategy. Comparison of the two-concept of security design strategy is done on the basis of Syskey concept and lssas.exe concept. After the study of all various design strategy we find that different concept and different requirement and have advantages and disadvantages depending on the different parameters and different technique. In this thesis is to help authorized users who have lost their password get access to the current operating system. All of the procedures and tools described in this thesis require physical access to the computer containing the operating system where the password is lost. Some may say that when the security of a system is defined, the first to be considered is the physical security of your computer. In other words, this means that if an unauthorized person acquires physical access to your system, the files on the system are characterized as lost. These statements should not be an excuse to have a poor login security. The first part of this thesis had the purpose of giving a basic understanding of the login mechanism and password handling of the selected operating systems. This highlights existing weaknesses of the login mechanism, and may motivate the operating system vendors to improve the password handling security in their systems. The second part gives a comprehensive step-by-step description of various existing procedures and tools. These can be used by a forgetful person to reset or recover the lost Administrator password. The last part presents an empirical password study that was carried out to test the strength of 30 different passwords. For Further research can be done to choose the best concept out of all design strategies for a particular problem considering all the parameters. In all the problems it is seen that many design strategies are tested and design are made using various techniques. If some guidelines are available regarding the suitability of a particular technique to a problem, then a lot of time can be named and designed may be developed only in that technique method.

REFERENCES

[1] Siti Rahayu S., Robiah Y., Shahrin S., Mohd Zaki M., Faizal M.A., and Zaheera Z.A, “Advanced Trace Pattern For Computer Intrusion Discovery”, Journal of Computing, Volume 2, Issue 6, June 2010, Issn 2151-9617.

[2] Michael Muckin, Lockheed Martin:, “Window security Internal “,Corporate Information Security Engineering, 2009.

[3] US-CERT Technical Cyber Security Alert TA04-104A Multiple Vulnerabilities in Microsoft Products, US-CERT April 14, 2004.

[4] Microsoft Security Bulletin MS04-011: Security Update for Microsoft Windows (835732), Microsoft Corporation April 13, 2004.

[5] S. Swanson, "Way too many passwords, not enough protection," in Chicago Tribune, online edition ed. Chicago, 2003, pp. 1.

[6] M. Zviran, Haga, William, "Password Security: An Empirical Study," Journal of Management Information Systems, vol. 15, pp. 161- 185, 1999.

[7] T. Jones, "Too many secrets? Password proliferation leads to user fatigue," in Columbia News Service - Columbia University Graduate School of Journalism. New York, 2002.

[8] S. Swanson, "Way too many passwords, not enough protection," in Chicago Tribune, online edition ed. Chicago, 2003, pp. 1.

[9] I. Sommerville, Software Engineering, 6th ed. Essex, England: Pearson Educational Limited, 2001.

[10] J. J. Whitmore, "A method for designing secure solutions" IBM Systems Journal, vol. 40, pp.747-768, 2001.

[11] Mudassar Raza, Muhammad Iqbal, Muhammad Sharif and Waqas Haider ,” A Survey of Password Attacks and Comparative Analysis on Methods for Secure Authentication, World Applied Sciences Journal 19 (4): 439-444, 2012 ISSN 1818-4952;

[12] X. Wang, Y. L. Yin, and H. Yu, “Finding collisions in the full SHA-1,” Lecture Notes in Computer Science, vol. 3621, pp. 17–36, 2005, Proceedings of Crypto 2005.

[13] R. Morris and K. Thompson, “Password security: a case history,” Commun. ACM, vol. 22, no. 11, pp. 594–597, 1979.

[14] M. Luby and C. Rackoff, “A study of password security,” J. Cryptol., vol. 1, no. 3, pp. 151–158, 1989.

[15] Secure Hash Standard, National Institute of Science and Technology Std. Federal Information Processing Standard (FIPS) 180-1.

[16] X. Wang, Y. L. Yin, and H. Yu, “Finding collisions in the full SHA-1,” Lecture Notes in Computer Science, vol. 3621, pp. 17–36, 2005, Proceedings of Crypto 2005.

[17] P. Oechslin, “Making a Faster Cryptanalytic Time-Memory Trade-Off,” in The 23rd Annual International Cryptology Conference, CRYPTO ’03, ser. Lecture Notes in Computer Science, vol. 2729, 2003, pp. 617–630.

[18] D. V. Klein, “Foiling the cracker: A survey of, and improvements to password security, (revised paper with new data),” in 14th DoE Computer Security Group, May 1991.

[19] M. Wilkes, Time - Sharing Computer Systems. New York, NY, USA: American Elsevier, 1968.

[20] Russinovich, Mark and Solomon, David, Windows XP: Kernel Improvements Create a More Robust, Powerful, and Scalable OS, 2001.

[21] Russinovich, Mark and Solomon, David, Windows XP: Kernel Improvements Create a More Robust, Powerful, and Scalable OS, 2001.

[22] Ahmed, A.A.E. and I. Traore, 2005. Anomaly Intrusion Detection Based on Biometrics, Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop, IAW '05.

[23] Klein DV. Foiling the Cracker: A Survey of, and Improvements to Password Security, (revised paper with new data). In: 14th DoE Computer Security Group; May 1991.