Global Information Assurance Certification Paper

SANS/GIAC Practical Assignment
For GSEC Certification
Version 1.2e

Password cracking with L0phtCrack 3.0

By Patrick Boismenu

June 2001

TABLE OF CONTENTS

INTRODUCTION

This paper was designed to describe how most password crackers operate. In today's world of security, password security is one of the priorities for all authentication-based protected systems.

There are many types of security that can be introduced in a system and one could not possibly describe them all at once but the authentication process is based on one or a combination of these three facts criteria:

Something you know.
Something you are.
Something you have.

For the sake of this paper, only the first fact will be touched. There are many other places on the Internet that describes in great detail the two others here is a few links where you can start regarding those two:

http://www.retina-scan.com/

http://www.sans.org/infosecFAQ/authentic/fingerprint.htm

http://www.hillschmidt.de/gbr/twotoken-0008.html

Now let's start cracking the material…

What exactly is a password cracker?

A password cracker is virtually any program that can decrypt passwords or can disable password protection. Most password cracker use a technique referred to as comparative analysis in order to crack the encrypted passwords. This technique that will be described in details later rely on one big factor, human laziness. Users tend to ignore the need for strong passwords. However, the blame is not entirely pointing to the users:

> Users are rarely, if ever, educated as to what are wise choices for passwords. If a password is in the dictionary, it is extremely vulnerable to being cracked, and users are simply not coached as to "safe" choices for passwords. Of those users who are so educated, many think that simply because their passwords is not in /usr/dict/words, it is safe from detection. Many users also say that because they do not have private files online, they are not concerned with the security of their account, little realizing that by providing an entry point to the system they allow damage to be wrought on their entire system by a malicious cracker.
>
> Daniel V. Klein, *A survey of, and improvements to, Password Security,* Software Engineering Institute, Carnegie Mellon University, Pennsylvania.

This problem often shows the weakest link theory within an organization. Password Security education would usually require minimal resources but even thought this is a critical security issue it is simply overlooked.

> …exploiting ill-chosen and poorly protected passwords is one of the most common attacks on system security used by crackers. Almost every multi-user system users passwords to protect against unauthorized logons, but comparatively few installations use them properly. The problem is universal in nature, not system-specific; and the solutions are simple, inexpensive, and applicable to any computer, regardless of operating system or hardware. They can be understood by anyone, and it doesn't take an administrator or a systems programmer to implement them.
>
> K. Coady. *Understanding Password Security for Users on and offline.* New England Telecommuting Newsletter, 1991.

The weak password phenomenon isn't a myth or something that disappeared ten years ago; it is ever present in the wide majority of systems and is currently viewed as one of the most critical threats to Internet Security:

> 8. User Ids, especially root/administrator with no passwords or weak passwords.
>
> Some systems come with "demo" or "guest" accounts with no passwords or with widely known default passwords. Service workers often leave maintenance

accounts with no passwords, and some database management systems install administration accounts with default passwords. In addition, busy system administrators often select system passwords that are easily guessable ("love," "money," "wizard" are common) or just use a blank password. Default passwords provide effortless access for attackers. Many attackers try default passwords and then try to guess passwords before resorting to more sophisticated methods. Compromised user accounts get the attackers inside the firewall and inside the target machine. Once inside, most attackers can use widely accessible exploits to gain root or administrator access.

Systems Affected:
All systems.

SANS Institute Resources,
How to Eliminate the Ten Most Critical Internet Security Threats.
The Experts Consensus v1.32 January 18,2001

Is it well protected by the operating system?

First you have a password generator, which will create an encrypted form of the password you have entered. Most password generators will use some form of cryptography.

There is a multitude of great sites on the net that describes in great details what cryptography is, here is a simple definition that sums it all up for us:

> Cryptography is defined as "the science and study of secret writing", concerns the ways in which communications and data can be encoded to prevent disclosure of their contents through eavesdropping or message interception, using codes, ciphers, and other methods, so that only certain people can see the real message.
>
> Yaman Akdeniz, *Cryptography & Encryption* August 1996,
> Cyber-Rights & Cyber-Liberties (UK)
> (Criminal Justice Studies of the Law Faculty of University of Leeds, Leeds LS2 9JT)

There are two kinds of cryptosystems: *symmetric* and *asymmetric*. Symmetric cryptosystems use the same key (the secret key) to encrypt and decrypt a message (Windows authentication), and asymmetric cryptosystems use one key (the public key) to encrypt a message and a different key (the private key) to decrypt it. Asymmetric cryptosystems are also called *public key* cryptosystems (PGP).

Let us take the example of the Data Encryption Standard (DES) algorithm and see how it works:

Your password is taken in plain text first. Sadly, for our example we will use one of the most popular password out there: ***password***

The password is then used as the key to encrypt a series of zeros (64 in all), the result, which is encoded, is then referred to as cyphertext, which is the encrypted version of the plain text password. On our test Windows 2000 computer, the plain text password mentioned above will become 8846F7EAEE8FB117AD06BDD830B7586C using NTLM authentication.

Basically it encodes it one-way which makes it part of the symmetric cryptosystem. What is interesting regarding this example is that while this operation seems simple by itself it is computationally complex and resource consuming to decode this form of encryption. Here is a few numbers that will help you understand the basis of this cryptosystem:

> The cryptographic algorithm [DES] transforms a 64-bit binary value into a unique 64-bit binary value based on a 56-bit variable. If the complete 64-bit input is used (i.e., none of the input bits should be predetermined from block to block) and if

> the 56-bit variable is randomly chosen, no technique other than trying all possible keys using known input and output for the DES will guarantee finding the chosen key. As there are over 70,000,000,000,000,000 (seventy quadrillion) possible keys of 56-bits, the feasibility of deriving a particular key in this way is extremely unlikely in typical threat environments.
>
> NIST, December 30, 1993. "Data Encryption Standard (DES)," *Federal Information Processing Standards Publication* 46-2.

Now, seventy quadrillion keys may look like a big number and it seems impossible to crack. This statement is partly right; although it is virtually impossible to crack, nothing stops us from comparing similar values.

How does the password cracker actually cracks?

The comparative analysis referred to earlier is a technique that solves almost all our problems when it comes to cracking passwords. Since the key is one-way encoded, the fastest and easiest way to crack down this key is by encoding the same word and comparing the hash referred to as 8846F7EAEE8FB117AD06BDD830B7586C in our previous example.

Here is how it all comes down to:

Get a dictionary file, which contains a huge list of words, from a site that has them:

ftp://ftp.cerias.purdue.edu/pub/dict/

These words will be fed through the program used to crack a specific password type.

The resulting hash will be compared with the one being attacked. If they match you have 90% chance of success. If they don't the next word is fed through the program and it starts again until it comes to the end of the wordlist or it cracked all passwords.

L0phtCrack 3.0: Crack 'em up!

This program is one of those that uses a comparative analysis attack on a pre-identified hash value. There were many versions of LC, the latest and most impressive is v3.01. This version brings us many enhancements over the previous versions, here is a brief description of those new tools as described at @Stake's website:

http://www.atstake.com/research/lc3/whatsnew.html

> **Support for Windows 2000**
>
> LC3 now runs cleanly on Windows 2000. It can extract unencrypted password hashes from systems that use Microsoft's SYSKEY protection, and it uses an updated packet sniffer that supports most Windows 2000 systems.
>
> **Distributed Cracking**
>
> LC3 lets an administrator speed up a time-consuming password audit by breaking it into parts that can be run simultaneously on multiple machines.
>
> **Hide Cracked Passwords**
>
> LC3 gives administrators the option to know whether or not a password was cracked without knowing the password itself.
>
> **Audit Time**
>
> Password auditors get a quantitative comparison of password strength from LC3's report on the time required to crack each password.
>
> **Wizard**
>
> LC3 offers a Wizard to help new password auditors configure and run their first audits quickly and easily.
>
> **Export**
>
> It's easier than ever to manipulate the results of a password audit by exporting results to a tab-delimited file.

Improved Product Support

Registered LC3 users get email support with one business day response time.

New Dictionary

LC3 now includes an optional 250,000 word English dictionary for comprehensive English dictionary audits.

Improved Password Management

LC3 lets users import passwords from multiple machines, and easily delete those they don't want to audit, directly in the LC3 window.

Let's demonstrate the power of this tool. We will simulate a situation where a password is cracked without the knowledge of the user or the administrators of the system.

First let us download the tool which is widely spread on the net, I would recommended their own website to avoid altered versions of the software (a PGP Signature of the software is also available at the under mentioned location):

http://www.atstake.com/research/lc3/download.html

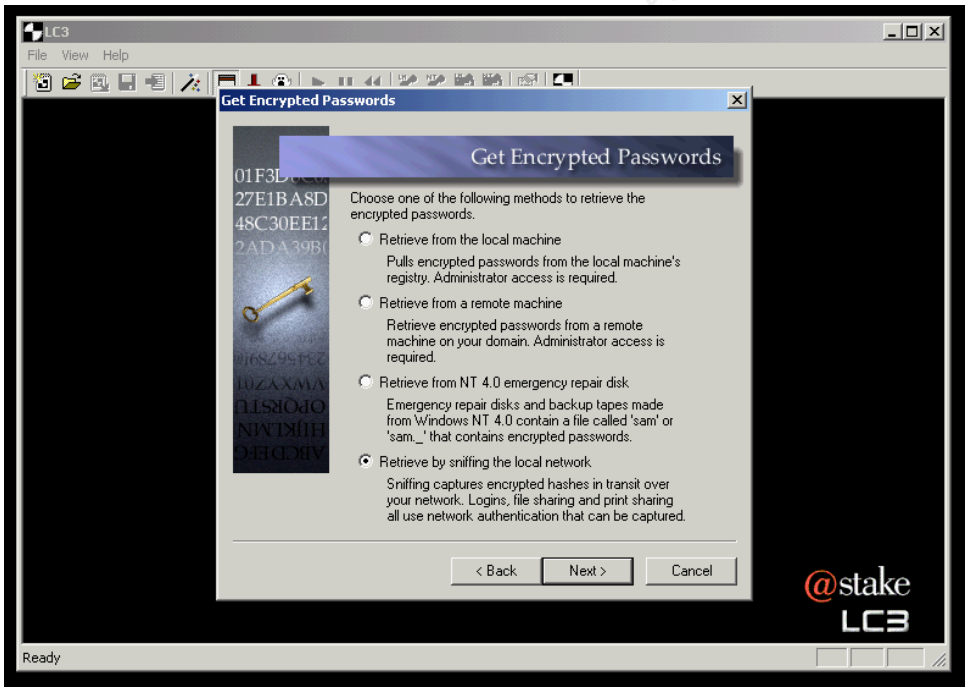
Once the software is downloaded and installed let's execute it.

This software is a trial version and can be registered directly with @Stake. After reading the following pages most of you will be running to register a copy.

Once inside the program, a wizard will guide you through the different functions that can be accomplished using this program. The menu enables us to retrieve passwords from the following locations:

1- Local Machine stored passwords. Admin Access Required.
2- Remote Machine stored passwords. Admin Access Required.
3- Emergency Repair Disk from NT4.0. (These disks are more often than not stored in a place where security isn't a priority, therefore leaving anyone to grab it and crack the file)
4- Sniffing the Local Network.

For our example we will be sniffing the local network, which will enable us to catch password hashes as there are authenticated between machines and all this without Administrator Access. In fact authentication is not even required, the only requirement is that you have Admin right on the local computer you are running LC3 from. This type of sniffing can be accomplished easily with the use of a Hub. If the network is using a switch, then other programs can be used to trick the switch (Cache Poisoning), which will enable the user to catch the passwords that are transited through it.



After that comes the type of attack that will be used against the hashed password if we are successful in catching one. There are three types and a custom option mentioned here.

The first type is a quick password audit, which will basically check for words, which are stored in a dictionary file. As seen previously, LC3 comes with a dictionary of 250,000 words. But there are dictionaries out there which have a much more impressive number attached to them.

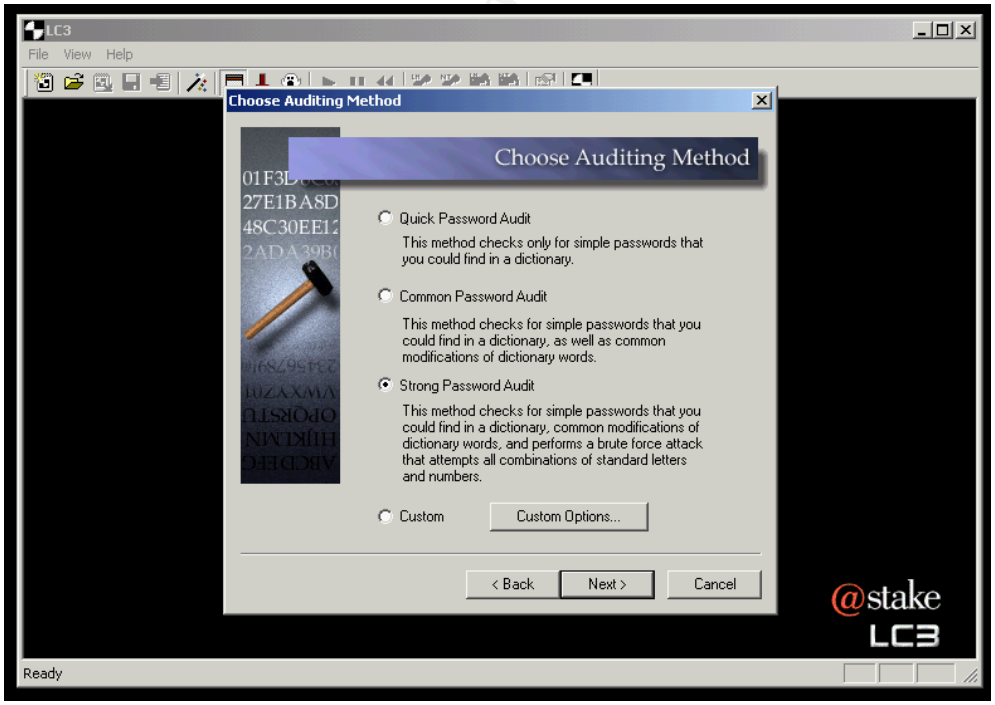
The second type is a common password audit, which is also referred to as Hybrid attack. It will check for common dictionary words used in passwords but will also add modifications to the words. Like **Internet** may become **internet99** or **9internet9**.

The third type is the strong password audit. This type will virtually test the two previous types and if they fail it will start a Brute Force attack. This type of attack is using all combinations of a set of characters. In our alphabet we have 26 letters and 10 digits. **A to Z** and **0 to 9**. There are also the special characters **!@#$% ^&*()_-+={}[]|\~`<>,.?/:;**

The brute force attack will use all the possible combination for those characters, first starting with 1 characters password and then when it's all done going for 2 characters passwords. This type of attack is the ultimate attack and it may take a long time to complete depending on the complexity of the password. But as we have seen earlier in this paper, complexity is often overlooked. Ultimately if you let the Brute Force attack go on until the end, it should result in the password being cracked.

Finally, the custom options lets you specify different cracking attributes. Like different dictionary, or different character set.

Of course all those types of passwords attacks are all done through comparative analysis. For this example the Strong Password Audit will be chosen.



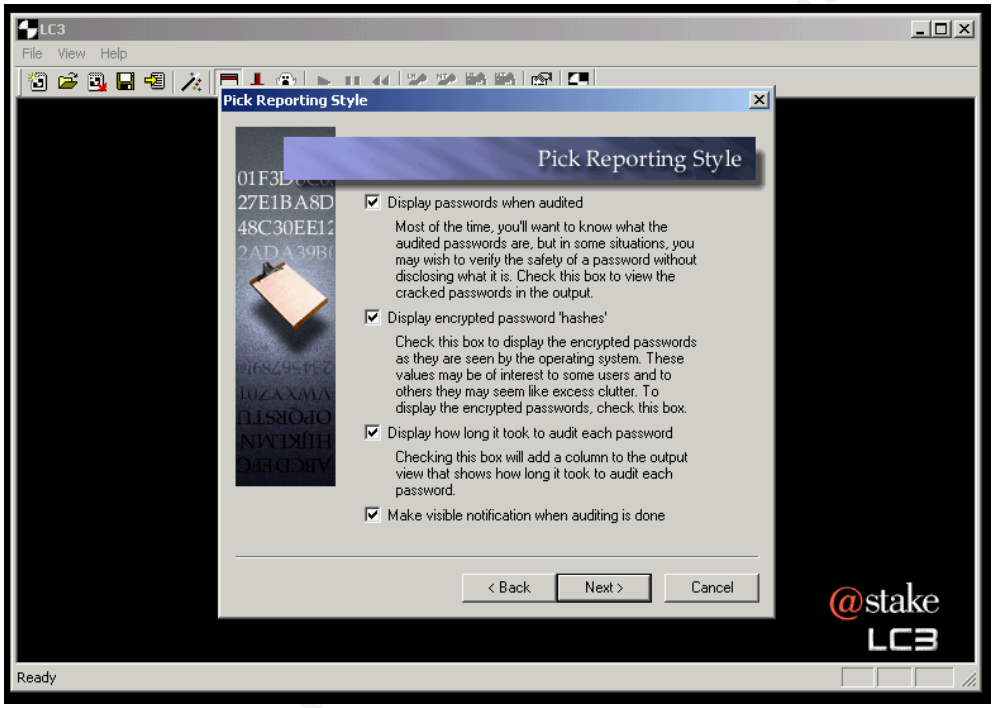
The next step will let you pick how LC3 should report to you what it will be doing.

You can specify whether or not you want the actual password to be listed to you or not. This option is great for Administrators that want to audit their user's password complexity without revealing the password itself.

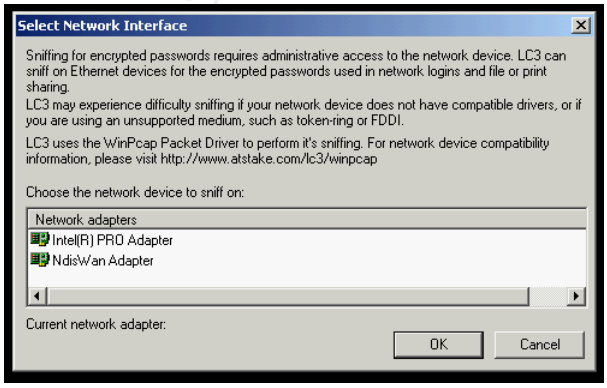
You can also display or not the password hash.

You can have LC3 tell you how long it took to crack it. Very good for Administrators again if they want to show their user that it took them 2 minutes to crack their **uncrackable** password.

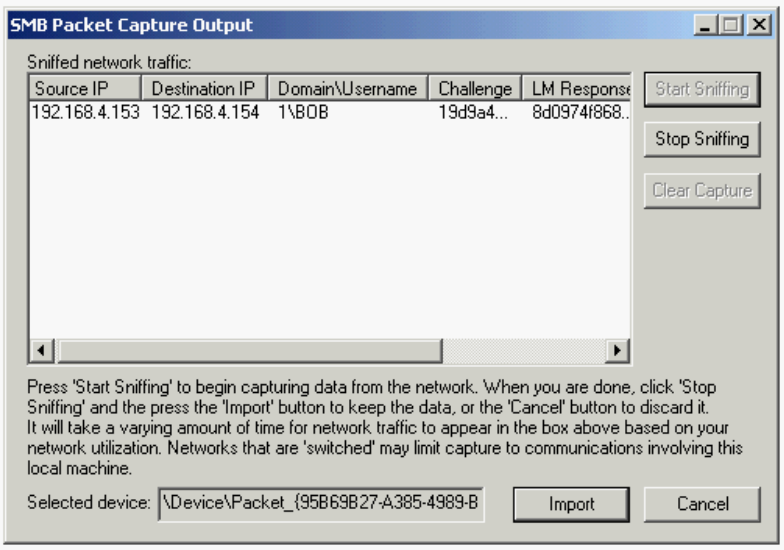
And you can request a visible notification when the auditing is finished.



The configuration of the audit is now completed; the next step is to choose the network adapter that will be used to sniff the network.



Once that step is completed. Just click on START SNIFFING and wait for someone to authenticate on the network. In heavy traffic network you should see someone come up fairly fast.



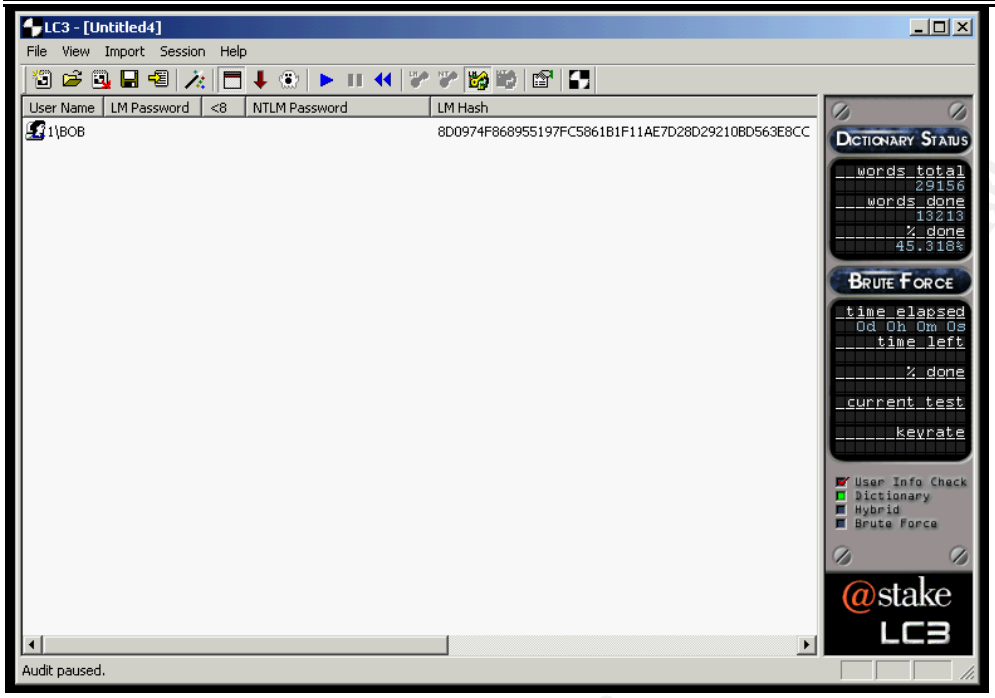
We now see a user called BOB incoming from IP address 192.168.4.153 trying to connect to a computer at IP 192.168.4.154, which for our example will be referred as EVE's box.

BOB may have tried to access something he was not supposed to, or maybe he was, in any scenario it does not matter, windows automatically passed on the credentials in which BOB is logged on to. In case those credentials would fail, EVE would probably tell BOB to specify a different username and password or just drop the connection. Either way, the password was caught traveling along the wire and it can now be attacked. You can sniff the network for as long as you want and catch as many passwords as you want. If BOB would authenticate to EVE's box again, another line would add itself with the same BOB credentials.

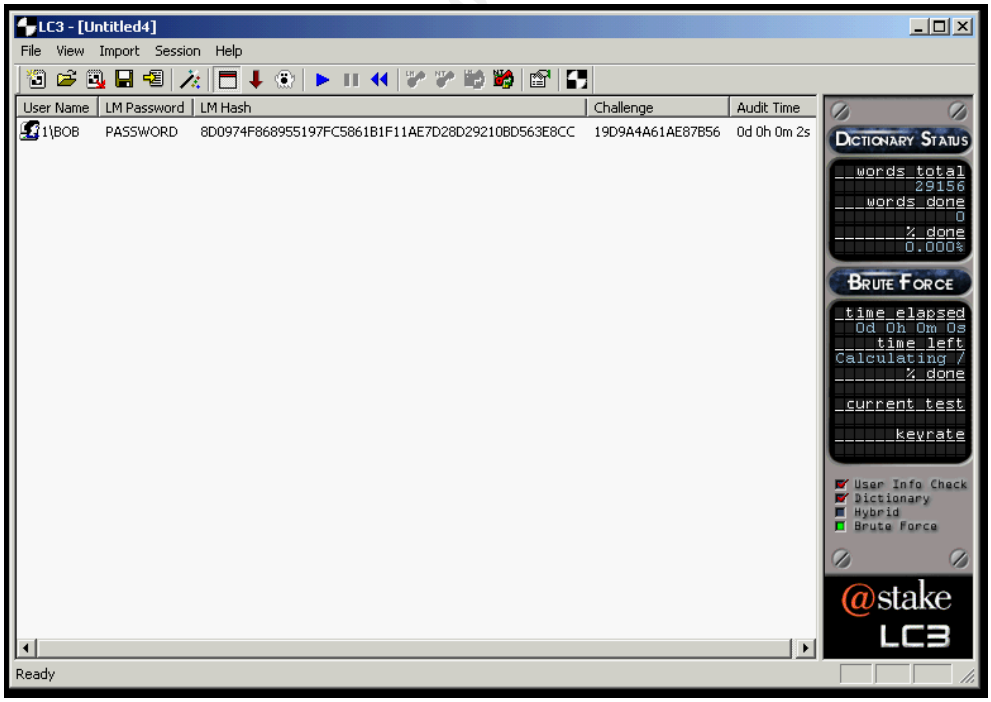
Once we captured the password we wanted to audit, we will click on STOP SNIFFING and then we can click on IMPORT, once that is done the password cracking should automatically start.

Note that the brute force attack is only available in the registered version of this software.

Let us see how long BOB's password can last with this program.



The password is currently being attacked with the dictionary that comes with LC3. Like seen earlier, it will go through every word and encrypt it using LM Hash encryption then will compare that value with the one seen above using the same challenge that was issued by EVE's box when authenticating to it.



Yes, it took a long 2 seconds before LC3 cracked the very secure password of BOB, which is shown in the previous slide under LM Password: **PASSWORD**

Now, the door is slightly opened and you have your foot inside the door. What if it was an administrator account?

You can easily see the power of this tool and how easy it is to comprehend and put in to action.

CONCLUSION

Should I just unplug all network equipment and go home?

No, fortunately there are defenses against password crackers. Here are a few of those defenses, which are enumerated by SANS in their GCIH track.

Establish a good password policy. Very important as discussed before, this is the first line of defense. Make it a very good one.

Guard the password file. Don't let that emergency repair disk lying around. Would you buy a steel door in front of your house and let your window wide opened besides it?

Disable LAN Manager Authentication. Please use NTLMv2, LANMAN authentication is very weak and easy to break, NTLMv2 is much, much stronger.

Use other forms of authentication. Remember the introduction when I mentioned there were 3 types of authentication, what you are, what you know and what you have. Implement at least two of those. It greatly enhances your security.

Finally, **educate your users**. Make them understand that any network is as secured as its weakest link.

Password cracking programs are growing in numbers and there are many discussions pertaining to their legality. Password cracker can be used as a valuable resource for any system administrator in order to alert them of weak passwords within the organization. The problem is not their existence; it is the lack of usage from system administrators.

Hope fully this paper will have enlightened the fact that you should enhance the security within your organizations. Especially regarding password security, the **FIRST** and, unfortunately sometimes, the **LAST** line of defense.

Resources

Sams.net - ***Maximum Security***, Second Edition, Chapter 10
Macmillan Computer Publishing USA
VIACOM

Daniel V. Klein, ***A survey of, and improvements to, Password Security***,
Software Engineering Institute, Carnegie Mellon University, Pennsylvania.

K. Coady. ***Understanding Password Security for Users on and offline***.
New England Telecommuting Newsletter, 1991.

SANS Institute Resources,
How to Eliminate the Ten Most Critical Internet Security Threats.
The Experts Consensus v1.32 January 18,2001

http://www.sans.org/topten.htm

Yaman Akdeniz, ***Cryptography & Encryption*** August 1996,
Cyber-Rights & Cyber-Liberties
(Criminal Justice Studies of the Law Faculty of University of Leeds, Leeds LS2 9JT)

NIST, December 30, 1993. "Data Encryption Standard (DES),"
Federal Information Processing Standards Publication 46-2.

@Stake L0phtcrack v3.01 Web Site and Documentation

http://www.atstake.com

CNET News.com
A new Windows password cracker
By Ben Heskett
Staff Writer, CNET News.com

http://news.cnet.com/news/0-1003-200-326537.html