

# Ten Windows Password Myths

Mark Burnett 2002-03-07

## Ten Windows Password Myths

by Mark Burnett

last updated March 7, 2002

---

With all of our advances in security technology, one aspect remains constant: passwords still play a central role in system security. The difficulty with passwords is that all too often they are the easiest security mechanism to defeat. Although we can use technology and policy to make passwords stronger, we are still fighting the weakest point in any system: the human element.

Ultimately the goal is to get users to choose better passwords. However, it is not always clear how to achieve that goal. The problem is that as creative as humans are, we are way too predictable. If I asked you to make a list of totally random words, inevitably some sort of pattern will emerge in your list. Selecting good passwords requires education. System administrators need to be educated and that education needs to be passed on to end users. This article is meant to bring you closer to understanding passwords in Windows 2000 and XP by addressing common password myths.

### **Myth #1: My Password Hashes Are Safe When Using NTLMv2**

Many readers will be familiar with the weaknesses in LanManager (LM) password hashes that made L0phtcrack so popular. NTLM made hashes somewhat stronger by using a longer hash and allowing both upper and lower-case letters. NTLMv2 made even more advances by computing a 128-bit key space and using separate keys for message integrity and confidentiality. It also uses the HMAC-MD5 algorithm for further message integrity. However, Windows 2000 still often sends LM or NTLM hashes over the network and NTLMv2 is also vulnerable to in-transit (also known as replay) attacks. And since LM and NTLM password hashes are still stored in the registry, you will still be vulnerable to attacks against the SAM.

It will still be some time until we are completely free from the grips of LanManager. Until then, do not assume that your password hashes are safe.

### **Myth #2. Dj#wP3M\$c is a Great Password**

A common myth is that totally random passwords spit out by password generators are the best passwords. This is not true. While they may in fact be strong passwords, they are usually difficult to remember, slow to type, and sometimes vulnerable to attacks against the password generating algorithm. It is easy to create passwords that are just as strong but much easier to remember by using a few simple techniques. For example, consider the password "Makeit20@password.com". This password utilizes upper and lower-case letters, two numbers, and two symbols. The password is 20 characters long and can be memorized with very little effort; perhaps even by the time you finish this article. Moreover, this password can be typed very fast. The portion "Makeit20" alternates between left and right-handed keys on the keyboard, improving speed, decreasing typos, and decreasing the chances of someone being able to discover your password by watching you (for a list of nearly eight thousand English words that alternate between left and right-handed keys, see <http://www.xato.net/downloads/lrwords.txt>.)

The best technique for creating complex passwords that are easier to remember is to use data structures that we are accustomed to remembering. Such structures also make it easy to include punctuation characters in the password, as in the e-mail address example used above. Other data structures that are easy to remember are phone numbers, addresses, names, file paths, etc. Consider also that certain elements make things more memorable for us. For example, patterns, repetition, rhymes, humor, and even offensive words all make passwords that we will never forget.

### **Myth #3. 14 Characters is the Optimal Password Length**

With LM, password hashes were split into two separate 7-character hashes. This actually made passwords more vulnerable because a brute-force attack could be performed on each half of the password at the same time. So passwords that were 9 characters long were broken into one 7-character hash and one 2-character hash. Obviously, cracking a 2-character hash did not take long, and the 7-character portion could usually be cracked within hours. Often, the smaller portion could actually be used to assist in the cracking of the longer portion. Because of this, many security professionals determined that optimal password lengths were 7 or 14 characters, corresponding to the two 7-character hashes.

NTLM improved the situation some by using all 14 characters to store the password hash. While this did make things better, NT dialog boxes still limited passwords to a maximum of 14 characters; thus the determination that passwords of exactly 14 characters are the optimal

length for the best security.

But things are different with newer versions of Windows. Windows 2000 and XP passwords can now be up to 127 characters in length and so 14 characters is no longer a limit. Furthermore, one little known fact discovered by Urrity of [SecurityFriday.com](http://SecurityFriday.com) is that if a password is fifteen characters or longer, Windows does not even store the LanMan hash correctly. This actually protects you from brute-force attacks against the weak algorithm used in those hashes. If your password is 15 characters or longer, Windows stores the constant AAD3B435B51404EEAAD3B435B51404EE as your LM hash, which is equivalent to a null password. And since your password is obviously not null, attempts to crack that hash will fail.

With this in mind, going longer than 14 characters may be good advice. But if you want to enforce very long passwords using group policy or security templates, don't bother - neither will allow you to set a minimum password length greater than 14 characters.

#### **Myth #4. J0hn99 is a Good Password**

While it does pass Windows 2000 complexity requirements, "J0hn99" is not as strong a password as it appears. Most password crackers have rules that can try millions of word variants per second. Replacing the letter "o" with the number "0" and adding a couple numbers is no big deal to a password cracker. Some password crackers have rulesets that can create password combinations well beyond the average user's creativity or patience.

A better approach is to be less predictable. Rather than replacing "o" with "0", try replacing "o" with two characters such as "()" as in "j()hn". And of course, making your password longer will make it even stronger.

#### **Myth #5. Eventually Any Password Can Be Cracked**

Although a password may eventually be discovered through some means (such as through a keylogger or through social engineering), it is possible to create a password that cannot be cracked in any reasonable time. If a password is long enough, it will take so long or require so much processing power to crack it that it is essentially the same as being unbreakable (at least for most hackers). So yes, eventually any password can be cracked, but eventually may not fall in your lifetime. So unless you have the Government hacking away at your passwords, chances are you are pretty safe. Of course, advances in computing power may some day make this

myth a reality.

### **Myth #6. Passwords Should be Changed Every 30 Days**

Although this may be good advice for some high-risk passwords, it is not the best policy for the average user. Requiring frequent password changes often causes users to develop predictable patterns in their passwords or use other means that will actually decrease the effectiveness of their passwords. It is frustrating to a user to have to constantly think of and remember new passwords every 30 days. Rather than limiting password age, it may be better to focus on stronger passwords and better user awareness. A more realistic time for the average user may be 90-120 days. If you give users more time, you may find it easier to convince them to use better passwords.

### **Myth #7. You Should Never Write Down Your Password**

Although this is often good advice, sometimes it is necessary to write down passwords. Users feel more comfortable creating complex passwords if they are able to write them down somewhere in case they forget. However, it is important to educate users on how to properly write down passwords. A sticky note on the monitor is not a good policy, but storing passwords in a safe or even a locked cabinet may be sufficient. And don't neglect security when it comes time to throw those passwords away, many passwords have been compromised after hitting the garbage dumpsters.

You may want to consider allowing users to save passwords in software-based password storage utilities. These utilities allow a user to store many account passwords in one central location, locked with a master password. If you know the master password, you gain access to your entire list of passwords. But before allowing users to save passwords in such tools, consider the risks: first, it is software-based and therefore can itself become a target of attack, and, second, since it is all based on a single master password, that password becomes a single point of failure for all the user's passwords. The best technique is to combine technology, physical security, and company policy.

Sometimes passwords need to be documented. It's not uncommon to see a company in a panic because their admin just quit, and he's the only one who knows the server password. You should discourage writing down passwords in many situations, but if writing them down helps or is necessary, be smart about it.

## **Myth #8: Passwords Cannot Include Spaces**

Although most users do not realize it, both Windows 2000 and Windows XP allow spaces in passwords. In fact, if you can view a character in Windows, you can use that character in a password. Therefore, spaces are perfectly valid password characters. However, due to how some applications trim spaces, it is often best not to begin or end your password with a space.

Spaces can actually make it easier for users to come up with more complex passwords. A space is used between words therefore using spaces may encourage users to use more than one word in their passwords.

An interesting fact I recently discovered in my research is that spaces do not fall into any of the categories for Windows password complexity requirements. It is not a number or letter yet does not count as a symbol either. So while it will make your password more complex, it does nothing to help you pass Windows complexity requirements.

And finally, one drawback with spaces is that the spacebar makes a unique noise when tapped. It is not hard to hear when someone uses a space in their password. So use spaces, but don't overuse spaces.

## **Myth #9: Always Use Passfilt.dll**

Passfilt.dll is a component that will enforce strong user passwords. In Windows 2000 and XP it is implemented through the "Passwords must meet complexity requirements" policy. While it is often a good policy to enforce, some users may find it frustrating when their passwords are rejected because they are not complex enough. Even experienced administrators have likely had to enter multiple passwords before finally getting one that does pass complexity requirements. Frustrated users certainly are not going to be giving you or your password policy much support.

If you find users are frustrated with the complexity requirements, perhaps a better solution is to not enforce that policy but instead require long passwords. If you do the math you will see that a nine-character lower-case password is roughly as complex as a seven-character password that uses upper and lower-case letters and numbers. The only difference is how the password cracking software handles different character subsets; some brute-force password crackers may attempt all lower-case letters before trying numbers

Another alternative is to take the Platform SDK sample in the `\samples\winbase\Security\WinNT\PwdFilt\` directory and modify it to be a little more forgiving with password selection.

Educating users on what makes a password complex and giving them some ideas for strong passwords will also help.

### **Myth #10: Use ALT+255 for the Strongest Possible Password**

It common to see recommendations to use high-ASCII characters as the ultimate password tip. High-ASCII characters are those that cannot normally be typed on a keyboard but are entered by holding down the ALT key and typing the character's ASCII value on the numeric keypad. For example, the sequence ALT-0255 creates the character `<ÿ>`.

Although they are useful in some situations, you should also consider the disadvantages. First of all, holding down the ALT key and typing on the numeric keypad is something that can easily be observed by others. Second, creating such a character requires five keystrokes that must be memorized and later typed every time the password is entered. Perhaps a more effective technique would be to make your password five characters longer, which would actually make your password much stronger for the same number of keystrokes.

For example, a five-character password made up of high-ASCII characters will require 25 keystrokes to complete. With 255 possible codes for each character and five characters, the total possible combinations are  $255^5$  (or 1,078,203,909,375). However, a 25-character password made up of only lower-case letters has  $26^{25}$  (or 236,773,830,007,968,000,000,000,000,000,000,000,000) possible combinations. Clearly, you are better off just making longer passwords.

Another thing to consider is that some laptop keyboards make numeric keypad input difficult and some command-line tools may not accept high-ASCII characters. For example, you can use the character ALT+0127 in Windows, but you cannot type that character at a command prompt. Conversely, I have found that some character codes such as Tabs (ALT+0009), LineFeeds (ALT+0010), and ESC (ALT+0027) can be used when setting your password from a command prompt but cannot be used in any Windows dialog boxes (which may actually be a desirable side-effect in some rare cases).

Nevertheless there are times where it is good advice to use extended characters codes. If you have sensitive service or local admin accounts that are rarely used, sometimes the extended character set will be worth the extra keystrokes. Since few password crackers are set up to handle extended characters, that may be enough to make your password very difficult to crack. But in that case, don't stop with high-ASCII, one little-known fact is that you can actually make use of the full Unicode character set which has 65,535 possible characters. Still, a character such as ALT+65206 is not as strong as the equivalent number of keystrokes using regular characters.

One final note on extended characters is the use of the non-breaking space (ALT+0160). This character appears as a space and can often fool those who are somehow able to view the password. Say, for example, that an attacker is able to install a keylogger on your system. If you use a non-breaking space in your password, it will look like a regular space in the keylogger's logfile. But if the attacker is not aware of the non-breaking space, and without seeing the actual ASCII code, the password they think they have will fail. And many people simply are not aware that this character exists, although perhaps they do now.

## **Conclusion**

Some may disagree with individual points I have presented here, but that is the whole purpose. A myth is a half-truth. Many of the myths that I have attacked here were once good advice or they still are good advice but only in specific scenarios. But to many this advice has become a set of solid rules that are generally applied to all scenarios. Password advice, including my own, is nothing more than advice. You must determine which rules work for you and which do not. Perhaps the biggest myth of all is that there are fixed rules when it comes to password security.

Sometimes John99 is a good password and sometimes passwords must be changed in less than 30 days. Some passwords, such as administrator passwords, need more protection while others, such as user passwords, need less. You must take what you know as well as what I presented here to form a password policy that protects you best.

A good password is more than just a complex password. A good password is one that is not easily guessed but still easy to remember. It should be long and should consist of letters, number, and symbols, but still easy to type quickly with few errors. It should have elements of randomness that only a computer can provide while still having familiarity that only a human can provide.

But the best password of all is the one that the user chooses based on an educated understanding of passwords - a password that is hard to crack, but never forgotten. And the best password policy is one that helps users in creating these passwords.

[Privacy Statement](#)

Copyright 2006, SecurityFocus