

The
Broadcasters' Desktop Resource
www.theBDR.net ... edited by Barry Mishkind – the Eclectic Engineer

Bits and Bytes

And the Secret Password is...



By Curt "Cowboy" Flick

[July 2010] The first two words in computers must be "user" and "password." Back in the early days of computing, the main use was to make sure you got your email and not someone else's. Today, however, your password may be protecting anything from your web preferences to your employment records to your bank account. With so many hackers, crackers, and other attempts to find and exploit your password, it is worthwhile to ask: "is my password safe?" Curt Flick explains.

Is your data safe? Can you be sure no one else can get into your accounts? These are important questions to answer today, when computer theft of personal information and/or money is rampant around the world – and once the thieves gain access to your life, it can be changed forever.

With anything from your bank account to your medical history at risk, it makes sense to ask "What makes a good password? Or are any old passwords "good enough?" It may be helpful for you to know that I am considered one of the more paranoid system administrators out there.

You may be surprised - some passwords are really quite common - and really stink! At the same time, seemingly simple passwords can be really secure. Let us take a look and offer some ideas on policies that will work for you.

PASSWORD POLICIES AND PEOPLE

No policy that is ignored is going to be very useful. Unless you intend to check on everyone every day, it is best to go for the "middle of the road" and build policies that are not too loose – and not so tight that people will circumvent them at every opportunity.

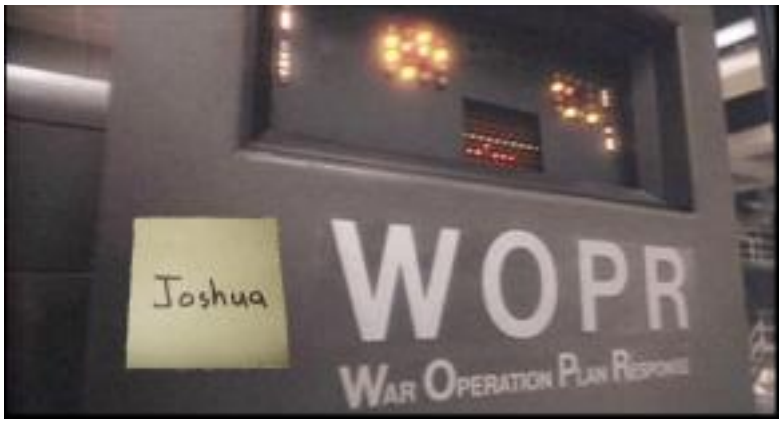
On the BROADCAST list (at www.radiolists.net), an on-line mailing list for broadcasters, each time a discussion starts on password practices – someone shares experiences that are often, frankly, amazing. Study after study has concluded that far too many users routinely set simple passwords that are so easy to guess that it is stunning. Passwords such as “password,” “123456,” “love,” or even common profanities are the first ones hackers try. Standard dictionary attacks will often allow bad guys to break in and steal everything you have before you know it.

One fellow on hotmail this past month lost his account to hackers – they took ten years of contacts and saved messages. While the guy eventually got his account back, all of his email and contact info was lost to him – and was already being circulated in spams and scams. Right now, all his contacts are getting an “emergency” message from him that he lost his passport and credit cards, needs help to get home. Any callers will get someone posing as a hotel operator, and be pleased to take money and credit card numbers from would-be helpers – and rip them off soundly.

BAD PASSWORD IDEAS

Likely you now understand why "default" or "123go" are bad as passwords. Many more examples are possible, such as anything a bank or credit card company has ever used. These would include your mother's maiden name, *any* digits out of a social security number or credit card numbers, and addresses.

On the other hand, some company employees get locked out of their accounts regularly because the policy is to use long, complex lines of characters for login and passwords. I have seen things like **TrE$%^978KjHg?/** used as supposedly secure passwords. However, they are forgotten quickly or, much worse, they get written down. It is often easy for visitors to indentify such companies - the staff has taped them to their desktop or monitors. This is exactly why lengthy strings of random characters are a really, *really*, bad idea.



Passwords that require being written down are not secure

The exception would be machines that auto-logon and do not have number of character restrictions, like the individual's company workstation, or roving profile. There, a 245-character "random" string that is entered once – and never again seen by human eyes – is entirely workable. This is similar to the machine “trust account” that Microsoft servers have used for years. Most likely, the machine will be replaced long before a password generator would have had time to crack it.

Posted passwords are, in effect, no password at all. They are worse, actually, since someone using no password at all does not usually advertise that fact anywhere with a Post-It. However, do not let this give you the idea that not having a password set is safe. A common first effort for hackers who might try to access your computer is to simply try to press the “enter” key.

Furthermore, overly complex passwords are unlikely to change very often. Unchanging passwords in a world of three-gigahertz-plus speed password-crunching attack computers is not a great idea *if someone wants to break in.*

PASSWORDS FOR PEOPLE

Moving targets, like changing passwords, are much harder to hit. This is why some outfits require all passwords to expire periodically. Even if the employees use "simple" changes, like password1 then password2, and so on, at least the change adds a small but definite step in security (a very small step to be sure). Some will alternate passwords back and forth whenever one expires. Not good, but still ever so slightly better than no changes at all.

For human use, *pass-phrases* are much better, provided they are not too obvious. Something like *The Quick Brown Cat Jumped Over The Lazy Fox Back* - slightly changed – is long, but easy for human memory. That human users can *easily* remember them is crucial.

Also good are *apparently* random strings derived from a pass-phrase, something like *MBWbiJ4*, derived from "My beloved wife's birthday is June 4" (This has the double-advantage of reminding one to get out and buy that birthday present).

Of course, using the actual, slightly altered, pass-phrase as the password is better, if the OS will allow it.

INTERNET CONSIDERATIONS

The use of actual phrases is definitely a good idea for any machines that are exposed to the Internet. The reason is that while *any* combination of "random" characters can be broken by a brute-force password generator, it is unlikely such a generator could re-create your phrase (within reason, assuming you take any care at all).

In many cases, a fair "generic" password would be something like the first and last letter of each word in the station slogan, in order, *and including at least one real word.* These things are easy to remember, yet difficult to decipher if one is not familiar with station operations or the details of an individual's life.

On the other hand, someone who is trying to crack your account might know some details of your life, so your child's name is a bad idea. Remember "Joshua" from *War Games?* Likewise, names of persons you admire, spouses, parents, etc. are bad choices. And anything taken from something that might eventually end up in a trash can – like non-shredded credit card statements or old program logs – is a bad idea.

DEVELOPING STRONG PASSWORDS

It is also a good idea to change passwords anytime an employee leaves, for whatever reason.

Good are seemingly random combinations of words; constructs like *spatula&motorCar*. They have enough characters to be secure – with no apparent relationship between the words – and a random character separator. It is my opinion that these make some of the best passwords (and pass muster with Microsoft's latest "strong password" requirements).

Want to be even more secure? Use three words, such as *ThE%paSsword*was* with random letters capitalized. The more words, the more non-standard characters, the stronger your password will be.

Hopefully, you are getting the idea.

HOPELESS USERS

Of course, for the individual whose password is her very own surname - and she still manages to forget it at least once a week - there just is no hope!

Since this unfortunately is not an uncommon problem, I recommend on-air and other such mission-critical machines should be set up to auto-log on to whatever account has been set up as necessary to maintain normal operations - *not* the same as the account used for administration.

For a mission critical system, a password forgotten because it was a long string of unguessable random characters chosen for "high security" can be much, much worse than no password at all when that system goes down. Even a password on a hard to find Post-It note will cause unnecessary delays in getting those machines back on-line and making money.

Of course, none of these machines should be exposed to the Internet without a very secure firewall between (and things like web browsing severely restricted), but that kind of system security would be the topic of another article.

STOPPING PASSWORD CRACKERS

As mentioned earlier, most crackers will try a dictionary attack first or simply try the more obvious permutations in common use, like *password, PaSsWoRd, passWORD, Pa**word, letmein, root, toor,* etc.

Although it was a common limit for Microsoft for many years, it is not safe to assume that 14 characters are enough. Some Linux systems especially recommend that eight characters is sufficient. SuSE did this for a while, and would not allow any more than eight characters. This is just silly, in my opinion. A determined cracker with access and time can use a brute force automated generation of many random characters until they hit one that works.

An automated, random character generator is different from a dictionary attack. These are used because is well known that the common "random" password web sites do not generate real words. Thus most current brute force generators filter and skip actual words and common dictionary permutations. They know that a dictionary attack will be far more efficient with real word passwords, so they devote their efforts to random single character combinations.

I have often wondered just how secure it would be to use a combination of common permutations, separated by a few random characters, something like *webmaster$#@root.* The words will be tried by a dictionary attack, but the random characters will not. Likewise the random characters would be tried in a brute-force attack, while the real words would (likely) not be used.

By the way – that typical bit on TV where the password generator displays single characters one by one as they are "found" to be part of the password? That is pure fiction. I know of no system that signals pass or fail until the entire password is entered, rightly or wrongly.

SOME USEFUL IDEAS

Where a user is not known for good password retention, I would recommend using something like the first and last letter of their names, both first and last names, or some permutation of that personalities

own name. (If air talent, use their real names, not their air names, and include at least one real word; it seems in many situations that is about as good as it gets.)

If you find that they have written this down, it is past time to reconsider your options.

Another idea is something like the old authenticator list we used to get for EBS, where the password du jour changed daily. If the passwords on the list follow the above rules for combinations of words, this will be hard to beat, but has the disadvantage that no one can remember them, so they will be written down.

This still is not necessarily a bad thing, though. You can publish the passwords for the week, and know that even if the list is stolen, it expires this weekend. (Also, if you publish the list in-house, there is far less reason for the staff to use post-its as well.)

REALLY STRONG PASSWORDS

If one wishes to get really secure and can assume some degree of intelligence on part of the authorized people, passwords that change with time of day and/or date are almost impossible to crack, unless it becomes too obvious. Even then, a cracker has to know that this happens in that specific IT center.

A solid tactic would be use of some easily manipulated algorithm that uses the date, and hour of the day, in conjunction with a word list. As an example, something like the words apple and lawnmower taken from a daily word list, then using the date and time - two time zones removed – can produce a password like *apple724161420lawnmower* for a station in Massachusetts when it is actually 4:20 PM on July 24 2016. Again, one would need an operating system capable of handling such high security.

Here is a something out of the ordinary that is will put your security in high gear: some UNIX systems will accept the back space as a valid character. I am pretty sure Microsoft systems will not even transmit the back space character, so a large portion of the “script kiddies” we worry about have been easily blocked by this “simple” trick. You can imagine my glee when I discovered that!

Tricks like that virtually insure that accessing some of the machines I have set up is not going to happen by random chance, nor by using Internet password finders. Learning to think outside the box can have advantages.

PASSWORD POLICIES TO AVOID

Unfortunately, many on-line banking systems *will* signal that the user name was correct, but the password failed. What are they thinking? Do *not* follow their example.

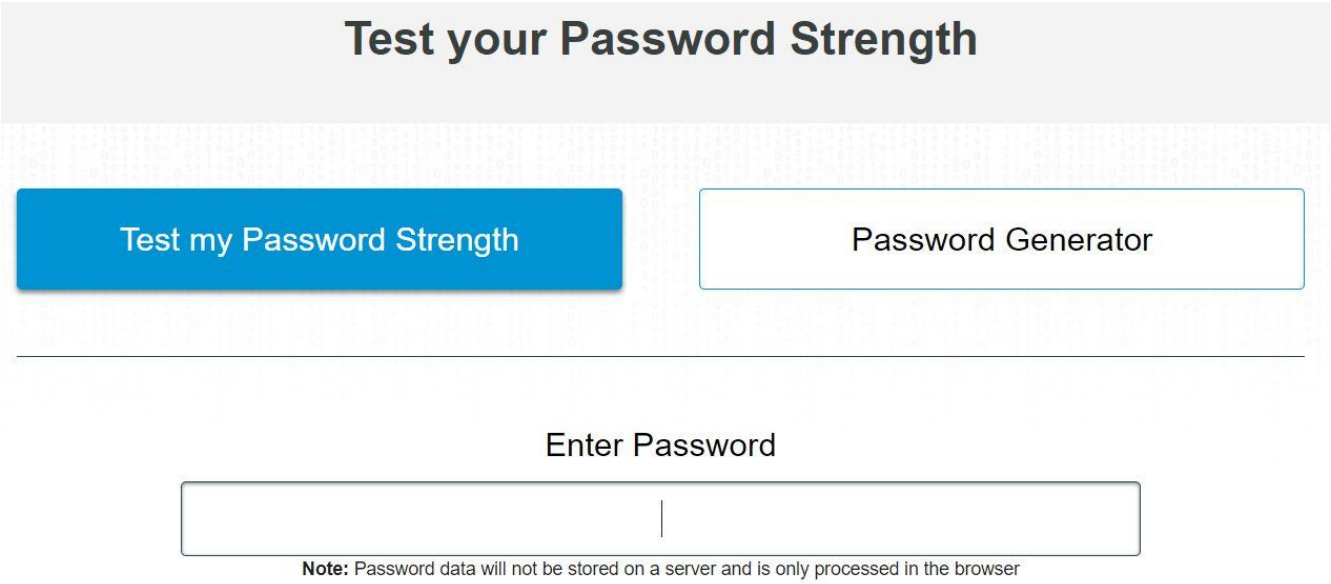
Older Microsoft systems are known to work in reverse. Give them a valid password, and they do not even check the user name. If you are using any NT4 systems, be aware that they and Windows 3.x, as well as Win9x systems, do work this way - at least, in part.

SOME USEFUL IDEAS

Pass-phrases and common words in uncommon combinations are far more secure than long strings of "random" characters. They will not be in the dictionaries and it will take a brute force attack years to hit that "random" combination - *if* it's been reprogrammed to include real words. If you have a cracker after you that is that determined, you have worse problems than simple net scans and weekend visitors.

There are a number of sites on the Internet that either will generate a password for you or will check your password and suggest whether it is a strong one or not. Depending upon whether you trust that site not to use the password generated or entered, this can be useful.

For example, passwordmeter.com is one site that gives you a clue, according to their algorithm, of whether your password is secure. Comparitech offers a password checker and generator, located here. They also offer some suggestions.



You will notice the checker is on what advertises itself as an https (secure http) site. Again, it is important for you to consider if you trust the sites.

Another option is secure "VPN" type applications that generate passwords for each session. The applications agree on a temporary password for the moment, encrypt and then use that secure channel to negotiate the password and encryption that will be used for that session.

BOTTOM LINE – BE SAFE

We can wax poetic, but the most secure passwords likely are those that *you* invent with the method(s) or combinations that you feel comfortable in using – and introducing a bit of randomness (you being you) into the methodology, and *not* locking your organization into some corporate wide "policy" but instead using corporate guidelines.

In the end, just keep in mind that simple, yet really secure, passwords are possible, but they are not going to come from random character generators, banks, corporate or university "policy," nor the hallowed halls of any sufficiently large organization.

- - -

Curt Flick is a contract engineer based in Akron, OH. Known to many as "Cowboy," he can be seen almost anywhere in the country as he works on a variety of broadcast station projects. Contact him at curt@spam-o-matic.com

Return to The BDR Menu