**Written by: Adam Berent**

# How safe is your security software?

**Version 1.3**

1.0 Bursting the bubble...
1.1 Choosing the right security software
2.0 Possible Attacks on Security Software
2.1 The Brute Force Attack
2.2 Dictionary Attack
2.3 Trojan Horse and Virus attacks
2.4 Public Key Algorithms and Attacks
2.41 Man in the middle attack
2.5 Security Bugs Attacks
2.6 Physical Attacks
2.7 Other Strange Attacks.
3.0 Conclusion

**How safe is your security software?**

This subject is somewhat a favorite of mine, since I am the developer of the popular encryption software ABI- CODER (www.abisoft.net). In this column, I will discuss the potential security threats against security software, and specifically, file encryption software.

**1.0 Bursting the bubble...**

OK, so you downloaded some security/encryption software, encrypted all your files, the sky has opened, the light has shined and you are safe. Well, not quite. There are still many possible attacks against you that are fairly easy to execute. No security software can protect you 100%.

There are also many types of security/encryption software available to you.
Are you using the correct one for the right task? Is your software written by professionals using well known encryption algorithms? Are you using your software correctly? Are you creating security holes? Unlike cars, planes and food, the quality of any software is not regulated by your government. Unfortunately, as a security software consumer the responsibility of quality control is placed on yourself. It's simply not enough that we cross our fingers and hope for the best. As a software developer, it is important that you protect your source code. As a network administrator, the security of the entire company depends on your choice.

**1.1 Choosing the right security software**

There are hundreds of security applications available out there. How do you choose the right one?

First, you eliminate the scams. Anything that promises "guaranteed security" etc. is obviously written by someone who knows nothing about it. There is no such thing as 100% security. You will understand why this is later in this introduction. Also, software that uses "Military strength" security

should be considered as suspicious. There is no such thing.

Make sure that the software uses a well know encryption algorithm that has been tested again and again for a long period of time. Let me stress this point: there is actually no good way to test for security of an encryption algorithm. It can't be done in a lab or through testing over a short period of time. Just because no one has ever bothered to break the algorithm does not mean that it is secure. This may take years of testing by educated professionals. Some of the encryption algorithms that are currently considered secure are: Triple DES, AES, RSA, IDEA, Twofish and Blowfish. Even these do not guarantee security. A hacker does not have to break to encryption algorithm to break the security software.

So we eliminated all of the obviously insecure products and we have a list of programs that use strong encryption algorithms and are possibly secure. Now which do we choose?

As I mentioned before, you can't test for security. A developer can create a software give it to 100 people to test. Does this mean that the software is secure? Of course not, chances are this test did not find even one security vulnerability. Mostly, because 100 test users will not have the combined knowledge that one serious hacker is capable of. Finding security bugs takes years of evaluation by serious professionals. This is why I would choose popular software that has been around for a long period of time. Still, this does not guarantee security; it just gives you a better chance of security.

**2.0 Possible Attacks on Security Software**

So, we choose a security software now we can sleep at night. Well not yet. Even the best security software is still vulnerable to different attacks. Here is an explanation of some of the more popular ones.

**2.1 The Brute Force Attack**

This kind of attack basically tries every possible key on an encrypted file until it finds the one that you used. This attack is quite easy to set up and can work on every single encryption algorithm no matter how complicated the math. The only way to defend against a brute force attack is to use keys at least 128 bits in length. That way there are just too many possibilities of your key to successfully retrieve it. To do this we also need an encryption algorithm that can use keys at least 128 bits in length. There are many such encryption algorithms. Some of the more popular ones are: Triple DES, Blowfish, Twofish, RSA. You should be wary of any software that uses an algorithm that has a maximum key length of anything less than 112 bits. DES supporting only 64 is one such an algorithm.

For public key algorithms such as RSA you must use keys that are even longer. Some public key software allows keys that are 2048 bits in length. PGP is one such software. (I will discuss this in further detail about public key algorithm later in this introduction)

If used properly a 128 bit encryption algorithm should protect you from a brute force attack. However you have to make sure that the key you enter is at least 16 characters long (128 bits). There is no point of choosing a strong encryption algorithm if you end up using a small and weak key.

**2.2 Dictionary Attack**

A dictionary attack is similar to the brute force attack. However, instead of searching all key possibilities it only attempts to use words out of the dictionary. This type of attack method can be very successful and will work unless you use keys that are not only 128 bits in length, but it may also

contain words that are not in the dictionary. Some of the more sophisticated dictionary attacks try not only English words but also combinations of words numbers and characters.
Recently, I tested software called L0pt Crack, which allows you to sniff your NT network for secure login packets. The software then records the one way hash that was send over the network and allows you to crack it. Now, remember that in theory a one way hash cannot be reversed to show its original value, which in this case is the users password. This is why the passwords Hash value is freely sent over the network. However, L0pt Crack computes the hash of an English word in its dictionary file and compares it to the intercepted one. If the two hash values match then the original values are the same and the password was found. After testing this software on one of my job sites, I surprised the network administrator when I handed him the login password for the CEO's computer. This technology is real and can easily be used against you.

To protect you from the dictionary attack most encryption software adds salt to their keys (add random bits) before encryption. This does not prevent dictionary attacks but it does make them too slow to execute in a real world situation. Encryption software that does not add salt to keys should be avoided.

To really get the most protection against dictionary attacks use stronger keys/passwords. For example "my name is my password" is a very bad key/password. "m3y-JKnAme()isUmy^&76pa8ssword" is much better. However, an unsalted key like this can still be easily cracked using the Dictionary attack, it just takes a longer period of time in doing so. Only keys that don't use any English words should be considered secure.

This makes encryption software very difficult to use securely. Chances are you will want to use passwords that are easy to remember. To make your life easier you can use a Password/Key Manager. This means storing all your keys in an encrypted file, so that the only key you have to remember is the one for the password/key manager. However, keep in mind that if that key was ever compromised, all your keys and passwords that were protected by it, and all your encrypted data would be up for grabs. The advantage is huge, because by using a password/key manager you only have to memorize one complicated password.

**2.3 Trojan Horse and Virus attacks**

Ok, so lets say we protected ourselves with a strong encryption algorithm, and we are using random looking keys, aren't we secure? Well, not really. There are still many ways to steal your data. One very good method is to use a Trojan Horse. A Trojan Horse is nothing more than a malicious piece of software hidden in another sometimes-useful program.

For example, Joe sends us an email to Bob with an attached file, in this case a game. The game allows you to punch a geeky-looking employee who, in turn responds in a funny voice to all your actions. Bob plays the game not realizing that the game is (meanwhile) installing a Trojan Horse on his computer. Bob gets bored and closes the game and then runs his trusty encryption software to send Joe an encrypted email. Meanwhile the Trojan Horse is running on Bob's computer broadcasting all his actions to Leo (who is a hacker and the creator of the Trojan Horse).

Those of you who is reading this and thinking "Wait a minute... I received something similar to that in an attachment" are not mistaken, and just had the security awakening of their lifetime. The rest of you can believe me - this Trojan Horse and 100s more exist and are capable of stealing all you keys and passwords, while you are typing them. Furthermore, some Trojan Horses can do much more, they can give the control of your computer over to the hacker who in turn can do even more with it than its original owner. Imagine the possibilities.

One of the most popular Trojan/Backdoor servers is Back Orifice. It is available for free download at http://www.cultdeadcow.com/tools/. To gain almost total control of the victim's computer all you have to do is send them an executable space.exe (can be changed to any name ex: sex.exe). Upon running the software BO will add an entry to

HKEY_LOCAL_MACHINE\Software\Microsoft\ Windows\CurrentVersion\RunServices

so that it is started every time the computer boots.

Viruses different from Trojan Horses because they infect or change an existing software to act differently from its original purpose. For example one can be a victim of a virus designed to change your encryption software so that it uses the same key (possibly known by the virus' designer) to encrypt all your files. The possibilities are endless.

There is no easy way to stop the Trojan Horse and Virus attacks, even I often download email attachments sent to myself by my friends. Be wary that your friends may not know that they are sending you a Trojan Horse or a Virus. They will most likely think that they are sending you a cool game. Their software might be fooled to send an automatic email to everyone in their phonebook. The solution is the right mix of paranoia a personal fire wall and an updated virus scanner. Avoiding using Microsoft Outlook will also minimize your vulnerability to most Viruses and Trojan Horses, specifically written for Microsoft Outlook and Microsoft Outlook Express.

**2.4 Public Key Algorithms and Attacks**

Public key algorithms are used primarily for communication between two parties. They are fundamentally different from secret key algorithms because they allow the two parties to communicate securely without a secure exchange of keys. Each party has 2 keys 1 that is public and 1 that is private. A message encrypted using the public key can only be decrypted using the private key. Since only my private key can decrypt the message encrypted using my public key, the public key can be known to anyone and the message will still be secure. Because everyone knows my public key, anyone can send me an encrypted email without ever meeting me.

2.41 Man in the middle attack

A simple attack on public key algorithm is to simply falsify your identity. For example; Leo wants to read all the messages between Joe and Bob. He sends two encrypted emails one to Bob pretending he is Joe and one to Joe pretending he is Bob. Leo can then just simply receive all messages going between Bob and Joe. To avoid detection he encrypts the emails again and sends them to their intended receiver. This way Bob and Joe will never miss a message and never know that their email is intercepted by Leo.

**2.5 Security Bugs Attacks**

Estimates from Carnegie Mellon University show that there is on average 5 to 15 bugs per 1000 lines of code (after testing). This fact is scary when you consider that the Windows OS has Millions of lines of code. Sooner or later, a hacker or researcher will eventually find a security hole(s) in most software. Chances are, that soon after that, your vendor will release a security patch to fix the security bug. This is where the problem begins. Hackers often attack software with known security bugs and hope that you are one of the people that did not apply the patch. Over 90% of successful attacks on the Internet can be prevented if the correct security patches are applied. Whatever software you are using, make sure that you keep it updated. In my opinion, you should visit the software site at least once every week to month. Some vendors even provide services that

automatically inform you when a new patch is available.

### 2.6 Physical Attacks

Well there is not much an encryption software can do about these, but these attacks exist and are quite possible. A great idea is to install a camera overlooking the users keyboard to record you typing in keys and passwords. There are many more like changing your keyboard to record your keystrokes. There are hundreds of variations which usually work well, because people don't think about protecting their workspace. If they are worried about anything at all, it is usually digital attacks.

### 2.7 Other Strange Attacks.

How about capturing the radiation of your computer or monitor?
This can be done up to a block away by your favorite government agency. Military computers are shielded to prevent this; most important, military computers are located in shielded rooms. However you don't have to worry about this if you are not trying to hide from your government. This is just a note that will hopefully get people to rethink using encryption software to do criminal deeds. If they want you -they'll get you.

### 3.0 Conclusion

This outline was created to outline the most basic attacks against your security software. Furthermore, each of these attacks can easily be individually discussed in their own book. Do not misinterpret this outline as your guide to security. This is meant to be used like a mini-guide or warning. If you wish to further your understanding of this subject, I suggest you read Bruce Schneier's Secrets and Lies: Digital Security in a Networked World. This book has a much broader explanation of network security described in a language that even the tech-savvy can understand.

 If you would like to contact me feel free to do so at aberent@abisoft.net or visit www.abisoft.net.