

Why Kids Shouldn't Be Criminal Hackers: An Explanation for High School Students, Parents & Teachers

by M. E. Kabay, PhD, CISSP

Associate Professor of Information Assurance, Department of Computer Information Systems
Division of Business and Management, Norwich University, Northfield, VT

Eleventh Edition, October 2003

Available on the Web at

http://www2.norwich.edu/mkabay/ethics/kids_not_hack.pdf

You may freely reprint this document if it remains intact and if you make it available at no charge to the recipients. This document may NOT be reposted on any Web site without permission.

At some time, someone is going to tell you how much fun it is to hack into computer systems and networks. I'm here to tell you it's a bad idea. I'd like you to understand what happens on the other side of that Internet connection when someone uses a computer system without permission.

Let me tell you a little about myself first.

I began programming computers when I was 15. It was 1965: that's probably around the time some of your parents were born. Then, computers were bigger than a refrigerator. The equipment needed enormous power cables to supply all the electricity. The computer rooms depended on gigantic air-conditioning units the size of four refrigerators to carry all the heat away. And all of this expensive equipment could serve exactly one user at a time.

Some big computers had about the same power as a hand-held programmable calculator of today. We used to tell the computers what to do by writing instructions – *programs* – using punch cards (rectangular cards with holes in rows and columns).

Gaining unauthorized access to computers back then was hard because they locked them up. The only way you got to program the computer was to hand in your cards through a window. If the technicians knew you or if you could show them your identification, they would feed your cards into the computer.

Whenever you finished running your program, it would disappear from the computer – that is, from the random-access memory or *RAM* – and someone else's program would be loaded into memory so it could be run or *executed*.

Giving your name to the technician was a form of *identification*. Identification means telling who you are. The next step was *authentication*. Authentication means proving that you are who you say you are. Showing your ID card from your school or from your employer is a form of *authentication* because it represents something that only you are supposed to own. When you or your parents enter bank-cards into an automatic teller machine, that card is *identification* and you are *identifying* yourselves. When you punch in your personal identification numbers (PINs), you are *authenticating* yourselves – you are binding the identifier to your real-world identity.

Identification and authentication (known to security people as “I&A”) are very important in today’s computer systems. There are four ways to authenticate yourself:

- what you have,
- what you know,
- what you do, and
- what you are.

Today, instead of speaking to a technician, now you usually type in a *user-ID* and give a *password* that no one else is supposed to know. This is *what you know*.

Other computer systems read a special card (a *smart card*) that has a *microprocessor* in it. The microprocessor is the part of a computer that does computations and comparisons. This smart card belongs only to you: this is *what you have*. This method of authentication is similar to how we carry physical keys with us to let us into homes or cars. The microprocessor in a smart card makes it harder to make a fake smart card because it contains secret and unique information. That is, it is hard to *counterfeit* smart cards.

Sometimes modern computers use other forms of I&A; for instance, some computers look at your fingerprints, the shape of your hand, your eyes (retinas or irises), or your face. They identify and authenticate you by *what you are*. This kind of I&A is also known as passive biometrics.

Some computer systems recognize your voice or your handwriting to identify and authenticate you: they use *what you do*. This kind of I&A is also known as active biometrics.

Well anyway, back in the 1960s and 1970s, computers and communications continued to evolve. Computers began to allow several people – even hundreds or thousands of people – to use them all at the same time. It also became possible to use computers from much farther away than in the old days. The development of *modems* meant that you could reach a computer from many miles away by using the phone system. Computer *networks* (interconnections of lots of computers) became more important in the world of computing during the 1980s.

However, remote access made it easier to use computer systems without permission. For example, people could find out your user-ID and password and pretend to be you – after all, there was no technician to check to see if it really was you.

Many computers back then had tight limits on how many hours of use you had on the computer every month. Someone who used your user-ID was really stealing computer time from you. These people who used the computer in your name were among the earliest *criminal hackers*.

For a long time, people in the computer field used the word “hacker” to mean someone who enjoyed learning about technical matters such as amateur radios and computers. Unfortunately, the news media started using the word to refer to people who used computers and networks without permission and now

they often see the word as bad. I prefer to refer to *criminal hackers* to make it clear that being interested in computers does not have to mean breaking the law.

So let's go back to criminal hacking.

Some people used passwords that were easy to guess. They used their own name, or their dog's name, or their favorite sports team. Finding that kind of password is easy because there are only a few words to try. Trying out all the words in a list of likely possibilities is known as a *dictionary attack*.

But even if the password were not in a dictionary, it was still possible to try all possible passwords. Trying all possible passwords is called the *brute-force* method. Because today's computers are so fast, using the brute-force method to find passwords is easy, especially if the passwords are short or have only letters. To make brute-force attacks harder, use passwords that are at least eight characters long and have at least one number or special character (!\$%?&* _-+=<>) in them.

Other criminal hackers learned to trick people into revealing their user IDs and passwords using deception or threats. The criminals called this *social engineering* and it remains an important method of getting into closed systems even today.

None of you should reveal your password to anyone. On some big networks (like AOL and CompuServe), some people (even kids) have been posting fake messages that appear on your computer screen when you are online. The messages claim that you have to send in your user name (user ID) and passwords; some of them ask for your credit card too. The messages threaten that if you don't send in this information you will lose your access to the service. This is not true.

No real official of any Internet Service needs to ask you for your password anytime. These demands on your screen are social engineering by criminals or sometimes by kids who are trying to trick you into letting them into your accounts.

Don't ever give anyone your password.

So what do we mean by *criminal hacking*? It means using a computer without the permission of the owners. Criminal hackers claim that what they do is harmless as long as they don't change any information on the systems they break into.

You may very well already know some criminal hackers, because some of them are still kids. Many kids who become criminal hackers think that what they're doing is just good harmless fun – like a neat video game. They are wrong, and I'm going to explain why they're wrong.

To understand why using a computer system without permission causes problems, you have to understand the goals of information security. Information security involves six different aspects that need protection. The six principles are as follows:

- confidentiality,
- control,
- integrity,

- authenticity,
- availability, and finally
- usability.

Let's look at these one by one.

Security experts talk about *confidentiality*. Confidentiality refers to limits on who can get what kind of information. For example, you might want to keep it secret that you have a crush on that cute kid who sits in front of you! If someone were to find that out and tell other people, that would be a breach of confidentiality.

Your parents' bank account number and their secret number (the PIN, or Personal Identification Number) can allow a thief to take money out of their account at the banking machine. If somebody were to find out those numbers, that would be a breach of confidentiality.

Control is another kind of protection for information. Imagine what would happen if you wrote down your parents' bank account number and PIN in an envelope and then gave it to a stranger. Even if the stranger said he wouldn't open the envelope, your parents would be frantic. They would be worried because they would no longer have *control* over their own secret number and over their own bank accounts.

Something very similar – a *loss of control* – is what would occur if strangers broke into your house when everyone was away. Even if they didn't do anything, you would still feel uncomfortable. You wouldn't know if maybe the strangers did something bad to your food. Your parents might even want to throw the food away just in case. You might feel uncomfortable because maybe the strangers looked in your diary. These would all be indications of a breach of control.

Security people next consider the issue of *integrity*. Integrity refers to being correct. For example, suppose somebody took one of your exam papers and made your answers wrong. This would be a breach of integrity.

If someone were to take a check that your parents wrote and then changed the amount payable, *that* would be a breach of integrity. Changing information without permission is a breach of integrity.

Criminals have altered medical data. Changing medical records can lead to very dangerous situations for the patients. For example, sick people can be given the wrong medication.

Some criminal hackers have played around in school records. They changed grades that were recorded by teachers. Now of course, this may sound funny, but it stops being funny when you think about what would happen if somebody changed *your* grades and made them worse or made someone else's grades a lot better. Unauthorized modification of data is a *breach of integrity*.

Another principle of security is *authenticity*. Authenticity means that we should label information correctly. For example sometimes a criminal hacker sends electronic mail in somebody else's name. In

one case, a professor in a Texas university found that someone had broken into his e-mail account. The hacker sent out two thousand e-mail messages in the professor's name. These e-mail messages were full of hateful, racist language and therefore people who got the messages became very angry with the professor. This was not fair, because the professor didn't write the messages. However, he and his family received death threats and had to be put under police protection when people threatened to burn the family house down. This was an example of the results of a *breach of authenticity*.

You might want to think about how embarrassing it would be if someone were to send e-mail messages in *your* name that said things you didn't agree with. Suppose someone were to insult your teachers by sending them messages signed with your name. You might get into a lot of trouble even though you had not done anything wrong.

Protecting authenticity is one of the reasons that you must never reveal your password to anyone else. You have to protect the authenticity of your communications.

The fifth principle of information security is preservation of *availability*. Availability means having timely access to information. *Timely access* refers to getting hold of the information you need when you need it. For example, suppose you have to write an essay on the novel, *The Wind in the Willows*. You want to read the novel before you write the essay. If someone hides all the copies of the novel at the library and at the bookstores, you can't read the novel in time for your essay. That would be a *breach of availability* of that novel.

We call one of the most serious and widespread problems in today's computers *denial of service*. Denial of service can occur when someone overloads a computer system or network with bogus requests.

One bad case of denial of service occurred when someone who called himself "Johnny [x]Chaotic" subscribed dozens of people to hundreds of e-mail lists. These poor people began receiving e-mail on basket weaving, engineering, plumbing – you name it. One writer received 20,000 e-mail messages in a single day. Imagine trying to find your own e-mail messages if someone sent you 20,000 messages you did not want. It would take hours just to read through the subject lines to find the e-mail you wanted. That would be a *denial of service*. Denial of service is a *breach of availability*.

Perhaps you have heard about the massive denial-of-service attacks that took place in February 2000 against many large e-commerce sites such as Amazon, e-Bay and so on. The FBI (in the US) and the RCMP (in Canada) cooperated to track down the perpetrator; the RCMP arrested a 15-year-old Montreal boy who went by the *handle* "Mafiaboy" in his interactions on the Net. A hacker's *handle* is their online nickname (or *pseudonym*). It seems that this child boasted about shutting down businesses in the denial-of-service attacks and gave detailed information about the exact timings of the attacks – clues which led the federal police forces to find him.

Finally, the sixth principle of information security is *utility*. Utility means usefulness. For example, suppose you went to the local store and all of the prices were in Dalgadian Blowati Units but nobody knew how many Blade Units there were in the Dollar. That pricing would not be very useful to you even though it might technically be correct.

If a criminal hacker were to change your grades in the school computer so that they were *encrypted* (converted using a special rule) but no one knew how to convert them back to normal numbers, that would be a *breach of utility*. This kind of problem has actually occurred in some businesses, where programmers encrypted the programs needed for work. Without the special *decryption keys* the information was no longer useful, even though there were no other breaches of security. After all, the information locked up in the encrypted form was still confidential, under the control of the owners, still had complete integrity, was authentic, and was available. It just wasn't *useful*.

Now that you have some idea of what information-security people worry about, it's easier to understand why breaking into somebody's computer system is really bad.

Of course the obvious problems concern confidentiality, integrity, availability and authenticity.

Going into a computer system and reading other people's documents, other people's e-mail, or information relating to national security in military computers are obvious breaches of confidentiality. Such breaches can cause real problems. For example, one thirteen-year-old kid in Florida got into the medical records of people at the clinic where her Mom worked one Saturday a few years ago. The girl called a dozen people who had gotten blood tests the day before and she lied to them: she told them they had AIDS. The victims of her sick joke were terrified. One teenager's parents stopped her just as she was about to shoot herself with her dad's pistol. So you see, taking and using confidential information can lead to terrible consequences.

Changing accounting records, stealing money by making false bank transfers, altering prescriptions so the people can become sick, sending out bad e-mail using other people's names – these breaches of integrity and authenticity are all obviously bad.

One of the most popular forms of criminal hacking today is Web vandalism: damaging Web sites by substituting often obscene pictures and offensive text for the original materials. The CIA was renamed the Central Stupidity Agency; the Florida Supreme Court's Web page was turned into an illustrated sex-manual – you get the idea.

The people doing the damage are often children or young teenagers. These *cybervandals* are just like the kids who throw rocks through people's windows or who spray-paint curses and foul words on buildings. Maybe they are expressing their rage and rebellion – or maybe they're just trying to be liked by the crowd they hang around with. From the point of view of the Webmasters, though, they're childish nuisances who cause extra work for nothing.

Another group of criminal hackers claim to be noble political idealists; they call themselves or are called *hacktivists* and they deface Web sites that they think belong to political enemies. In the recent Kosova war, both sides in the conflict damaged each others' Web sites. Hackers in China and in Taiwan have been attacking Web sites in each other's countries for years.

The recent denial-of-service attacks that may have been launched by children have caused billions (yes, billions) of dollars of lost sales and costs of recovery. These attacks used hundreds and perhaps thousands of computers to swamp the victims with requests for information. Criminal hackers installed special *slave* or *zombie* programs on poorly-secured computer systems. These slave programs were then ordered to attack the main victims using coded communications from the criminal hacker controlling

them. The slave programs made the computers they were on send out thousands of messages to the victims' computers, swamping their communications. No one else could get much of a response from the computers under attack

Part of the cost of cleaning up after the denial-of-service attacks came from having to pay employees to search out the slave programs and remove them.

Some criminal hackers claim that if they don't alter information, they haven't done anything wrong – or at least, they haven't done anything *really* wrong, as they say. This point of view is simply, flatly incorrect and I want to explain how it's incorrect.

The fundamental problem caused by unauthorized access to information systems is loss of control. Let me explain what really happens when somebody breaks into a computer system – that is, accesses the system without authorization..

First you have to understand that many people depend on computer systems to get their work done. The computer systems they depend on are known as *production systems*. For example the local banks need to have computers to process people's paychecks. The computers must take the right amounts from the employer bank accounts and must deposit the right amounts in the employee bank accounts. What do you think happens in the bank if someone breaks into their computer system? I'll tell you: it's a real mess.

The poor bank employees don't know whether the intruders have damaged some of their bank records or programs. Even if the criminal hackers leave a note (you know, "W3 DIDN'T DO 4NY7H1NG WR0NG C4USE W3R3 313373" using that silly code of theirs – see if you can figure it out), how do the employees really know if everything's still OK or whether the hackers have damaged something?

The only thing to do is to check. Security experts say that such a system is no longer *trusted*. We say that these systems have been *compromised*. The employees have to reestablish trust in the data and also in the programs.

Criminal hackers have been known to insert their own changes to certain computer programs. Criminal hackers often leave what are called *back doors* into the systems they've already broken into. Back doors allow the hackers to re-enter the compromised computer systems anytime they want. This kind of change to system software is a real threat to the people that have been victimized. It can take days to check all of the information on a computer system that has been broken into. Sometimes the checking costs hundreds of thousands of dollars in wasted salary or consulting fees.

I remember that when I was in charge of a big computer center in the 1980s, my staff and I would spend from midnight to 6 in the morning every day for five days testing the new version of the computer operating system (something like Windows or MacOS or LINUX) for the large computers or *mainframes* that we used. We would work hard to see if they new software was working properly.

Now remember, that software was sent to us by the computer maker. If we were willing to spend five nights testing the *manufacturer's* software, doesn't that tell you how important trust was for us? Now think about why on earth we would trust a production system that might have been damaged by a

criminal hacker. It wouldn't make sense. *We have to check the system after every intrusion.* So that's why it's not true that breaking into computer systems is harmless fun.

I hope you will think about this the next time someone suggests that you play with them by breaking into somebody's computer system. Try to tell them this isn't a video game. Hacking computers hurts real people. The victims of hacking spend sleepless nights away from their families working hard to see if their computer systems have been damaged by intruders. They worry about it. If there has been damage, it can cost lot of money to fix the data and the programs. This money lowers profits for companies or increases costs for nonprofit organizations.

If the criminal hackers laugh at the costs and tell you that "it's only a company – it's not real people" then you will know that they are either stupid or they are deliberately lying to you. Organizations are made of real people. Real people lose because of criminal hacking.

Criminal hackers also sometimes take services from the telephone companies without paying for them. For example, they use special phone numbers called *teleconference bridges* to talk to each other. The company that rents the bridge ends up paying a lot of money per minute for those stolen phone calls. Stealing telephone services is known as *phreaking*.

Some criminals claim that phreaking doesn't hurt anyone. I have spoken with phreakers who argue, "Well, the phone company's stuff is already paid for and it's just sitting there, so how can it hurt to use their networks as long as they are not all being used? It doesn't cost them anything."

This argument is wrong because in fact even ordinary telephone calls do cost something to the company that is being cheated. Any phone call that goes between the areas controlled by different local phone companies must be paid for by transfers of money from the originating phone company to the destination phone company. It gets even worse for international calls. International agreements govern long-distance phone calls; these laws force automatic payments from the originating telephone company to an *interexchange carrier* and then to every other company that carries the signal. So stealing phone calls is truly stealing, with money flowing from the company that is cheated to all the other companies involved in the call.

If your friends tell you they are playing with *blue boxes* or *red boxes*, it means they are stealing phone services. These boxes (which may be computer programs nowadays rather than physical boxes) generate the sounds that control the long-distance telephone switching computers and trick the companies into granting free calls. Don't get involved with these thieves.

Yet another kind of criminal hacking involves creating false credit-card numbers and using them to pay for stolen services or stolen products. This can be a very serious crime. One teenager in Australia got in over his head. He created fake credit-card numbers and bought about \$100,000 in products over the phone. The police caught him. Eventually he pleaded guilty to more than 400 separate counts of fraud and theft. This young boy's name became public and he will have to live down his criminal acts for the rest of his life. His dishonesty will hurt him every time he applies for college or for a job.

If someone tells you about a neat program for generating free credit-card numbers, don't get involved. In fact, you should tell your parents. Oh – and as for the often-repeated claim that this kind of theft is victimless because the banks pay for the stolen numbers? It's not true either: actually everybody who

owns a credit card and has an unpaid balance pays. Because of the costs of fraud and theft, interest rates on unpaid balances are higher than they might be – about three times higher. For instance, once I got a credit card that was *secured* (guaranteed) by money I had in the bank. Its interest rate was about 5% while normal *unsecured* credit cards had an interest rate of 15%.

Trading or selling stolen software, which criminals call *warez*, is another form of hacking. Again, the criminals tell kids and each other that no one loses by copying software without permission from the people who created it or who own the rights to it. But people wrote the programs and people depend on sales of software to earn their living. Taking the fruits of their hard work without fair payment is sneaky and mean.

The latest fad is trading stolen music using software like Napster and Gnutella. You may have heard that some bands (like Metallica, for instance) are suing individual users – many of them kids – for sending each other copies of Metallica's songs.

Why should anyone think that it's OK to steal music that took hours of work to produce? Some kids think that somehow no one is being hurt. It's easy to copy the music and you can make as many copies as you want without destroying anything, so these kids say there is no theft. But the musicians and the people working for the record companies are trying to earn a living using this music. The owners of the *copyright* have the right to control how this music is shared. For example, a radio station has to pay a fee (a *royalty*) for permission to play a song from a CD. When someone buys a CD, the laws state that it must not be copied for sale to other people; the law protects the musicians and the other working people involved in producing and distributing the CDs so that they can benefit from their work.

There is a debate about whether current US laws forbid copying of copyrighted materials for personal use and even for free exchange of copyrighted music and film. Some jurists argue that the laws do allow copying a CD, for example, to play the music on a cassette at home. Some legal cases have allowed taping of TV shows for later viewing on a VCR. However, the enormous volume of copying and free exchange through services such as Napster and Gnutella raise serious questions about fairness. Napster, for example, has been sued for encouraging copyright violations – and so have nearly half a million individuals who traded copied music through Napster.

Some kids argue that the record companies get enough money from people who buy the CDs and so it's OK to steal the music by sharing it illegally. But why should only honest people pay for the music? And if *everyone* did what the thieves say, there would be no one to pay for the music at all.

In some parts of the world such as Asia, illegal copying of entertainment is a terrible problem for local artists. So many illegal copies are made that very few of the original CDs or videos are sold. Because the people who make illegal copies don't pay the artists anything at all, the lack of sales makes it difficult for local artists to earn a living.

A similar problem occurs with the kind of software called *shareware*. Programmers create useful programs and put them out on the Internet for free trials. However, the software clearly states that the trial is limited to a specific period (for instance, a month). After that, if you want to keep using the software, you are supposed to pay a modest fee. Only a few users pay the fee.

Using shareware past the time limit without paying the fee is theft. The thieves are cheating the creator of the program out of his or her dues.

The thieves sometimes whine that the price is too high and they *want* (or even *need*) the music without having to pay for it. Tell them that no one ever said there was a right to use the fruit of other people's work without permission. How would you feel if you invented something useful and people just used it without *your* permission?

No one has a *right* to make copies of someone's music or software without permission of the copyright holder. Claiming otherwise is unrealistic – and makes the thief sound like a spoiled brat.

By the way, there is one more issue about copying *intellectual property*: the fuzzy guidelines called *Fair Use*. This is one of the most difficult areas of intellectual-property law. In brief, Fair Use guidelines suggest (but it's always a matter of judgement) that one can legitimately make a limited number copies of written material under certain circumstances as long as you don't harm the commercial value of what you are copying (to explore this subject, see the reference to Lessig, Post & Volokh in the Readings section at the end of this paper).

Criminal hackers often claim that their intrusions into systems and networks are *useful* for their victims. They think that they are showing up weaknesses or holes in company security systems. The claim that owners of the systems ought to be grateful to the criminal hackers because otherwise they would have to pay money to expensive consultants. Consultants charge thousands of dollars for *penetration tests* whereas, the criminals say, they are doing it for free.

The problem is that the owners of the systems being “tested” by the criminals never asked for such tests. Suppose you were riding a skateboard and someone threw a rock at you. What if they said they threw the rock to show how unsafe skateboarding is: would you thank them?

Damaging the trustworthiness of a system or network while claiming to help the owners does not make sense. It's just an excuse, and a poor one at that.

Some people get caught when they hack, and their reputation for dishonesty can follow them for many years. I have met young people who didn't think they were doing anything wrong at the time; nevertheless, they discovered how hard it is get into a university or to get a good job once they have shown themselves to be untrustworthy.

Another problem with doing illegal things is that you open yourself up to *blackmail*. A blackmailer threatens to tell others that you have broken the law or otherwise done something wrong. If you pay them money or do other things they tell you to, they claim that they will keep your secret. However, most blackmailers are greedy and cruel: once you've paid, you will keep on paying forever. Don't put yourself in this bind – stay honest.

If kids continue criminal hacking past their eighteenth birthday, they can go to jail for unauthorized access to some kinds of computers. The laws of the United States, for example, also prescribe fines of up to a quarter of a million dollars for some *computer trespass*.

There are many ways to learn about computing. It is not necessary to become a criminal in order to learn. Enjoy computers and respect your fellow human beings while you enjoy this wonderful new world of cyberspace.

For Further Reading

You may find my little book *Cyber-Space Safety* helpful: it's available free at
< <http://www2.norwich.edu/mkabay/cyberwatch/cybersafety.pdf> >

Explore my Web site for pointers to additional reading in information security and ethics:
< <http://www2.norwich.edu/mkabay/ethics> >

About the author

M. E. Kabay began learning assembler at age 15 and had learned FORTRAN IV G at McGill University by 1966. In 1976, he received his PhD from Dartmouth College in applied statistics and invertebrate zoology. In 1979, he joined a compiler team for a new 4GL and RDBMS in the U.S., being responsible for developing the statistical syntax, writing the parser, error traps and code generation for statistical functions in the command language. Kabay joined Hewlett-Packard in 1980 and became a performance specialist, winning the Systems Engineer of the Year Award in 1982. He attained the status of Certified Systems Security Professional (CISSP) in 1997.

He was Director of Education for the National Computer Security Association (NCSA, later ICSA and then TruSecure Corporation) from 1991 to 1999. He worked for AtomicTangerine until June 2001 and in July 2001, accepted a position as Associate Professor of Information Assurance in the Department of Computer Information Systems of the Division of Business and Management at Norwich University, where he became the Program Director of the Master of Science in Information Assurance (MSIA) program in 2002.

He has written security columns for *Computer World*, *Network World*, *Computing Canada*, *Secure Computing Magazine*, *NCSA News*, *Information Security Magazine* and several other trade magazines. Dr Kabay has published over 750 technical papers in operations management and security and currently writes two columns a week for *Network World Fusion*; archives are at

< <http://www.nwfusion.com/newsletters/sec/> >. For other articles by M. E. Kabay, please see his Web site at < <http://www2.norwich.edu/mkabay> >.

E-mail from readers is welcome: <mailto:mkabay@norwich.edu>

For more information about the Norwich MSIA program, see < <http://www3.norwich.edu/msia> >.