Author: van Hauser / THC

I. INTRODUCTION

Please excuse my poor english - I'm german so it's not my mother language I'm writing in. Anyway if your english is far better than mine, then don't think this text hasn't got anything to offer you. In contrast. Ignore the spelling errors & syntax - the contents of this document is important ...

NOTE : This text is splitted into TWO parts.
The first one, this, teaches about the background and theory.
The second just shows the basics by an easy step-by-step procedure what to type and what to avoid.
If you are too lazy to read this whole stuff here (sucker!) then read that one. It's main targets are novice unix hackers.

If you think, getting the newest exploits fast is the most important thing you must think about and keep your eyes on - you are wrong. How does the best exploit helps you once the police has seized your computer, all your accounts closed and everything monitored? Not to mention the warrants etc. No, the most important thing is not to get caught. It is the FIRST thing every hacker should learn, because on many occasions, especially if you make your first hacks at a site which is security conscious because of many break-ins, your first hack can be your last one (even if all that lays back a year ago "they" may come up with that!), or you are too lazy to change your habits later in your career. So read through these sections carefully! Even a very skilled hacker can learn a bit or byte here.

So this is what you find here:

Section I - you are reading me, the introduction
Section II - the mental things and how to become paranoid

1. Motivation
2. Why you must become paranoid
3. How to become paranoid
4. Stay paranoid

Section III - the basics you should know BEFORE begin hacking

1. Preface
2. Secure Yourself
3. Your own account
4. The logs
5. Don't leave a trace
6. Things you should avoid

Section IV - the advanced techniques you should take a notice of

1. Preface
2. Prevent Tracing of any kind
3. Find and manipulate any log files
4. Check the syslog configuration and logfile
5. Check for installed security programs

6. Check the admins
7. How to "correct" checksum checking software
8. User Security Tricks
9. Miscellaneous

Section V - what to do once you are under suspect
Section VI - the does and dont's when you got caught
Section VII - a short listing of the best programs for hiding
Section VIII - last words, the common bullshit writers wanna say

Read carefully and enlighten yourself.

II. MENTAL

CONTENTS:

1. MOTIVATION

The mental aspect is the key to be successful in anything.

It's the power to motivate yourself, fight on if it hurts, being self disciplined, paranoid & realistic, calculate risks correctly and do stuff you don't like but are important even if you'd like to go swimming now.

If you can't motivate yourself to program important tools, wait for the crucial time to hit the target, then you'll never get anywhere with your "hacks"

A successful and good hacker must meet these mental requirements. It's like doing bodybuilding or a diet - you can learn it if you really try.

EVEN THE BEST KNOWLEDGE WON'T HELP YOU UNTIL YOU ARE REALLY CONCERNED TO DO THE PREVENTIONS AND ACTUAL MAKE THEM !

2. WHY YOU MUST BECOME PARANOID

It's right that normally being paranoid is not something which makes your life happier. However if you aren't expecting the worst, anything can hit you and throw you off balance. And you are risking very much with your doings. In your normal life you don't need to worry much about cops, thieves and therelike. But if you are on the other side remember that you make other people a hard life and bring them nightmares plus work - and they want to stop you.

Even if you don't feel like committing a crime - you actually do. Hacker-Witchhunting pops up fast and gets everyone who might be involved. It's the sad thing : YOU ARE GUILTY UNTIL PROVEN OTHERWISE ! Once you've got the stigma being a hacker you'll never get it off. Once having an entry in your police record it's very hard to find a job. Especially no software company, even no computer related company will ever hire you, they will be afraid of your skills, and you will see yourself being forced to emmigrate or your life lost. Once you fall down only a few can get up again.

Become paranoid!
Protect yourself!
Remember you have got everything to loose!

Never feel silly doing THAT extraordinary action against tracing!
Never bother if someone laughs on your paranoid doing!
Never be too lazy or tired to modify the logs!
A hacker must do his work 100% !

3. HOW TO BECOME PARANOID

If you've read the part above and you think thats true, it's easy - you've got already become paranoid. But it must become a substantial part of your life. If you made it becoming a good hacker always think about whom to tell what, and that you phone calls and emails might be monitored. Always reread the section above.

If the above didn't helped you, then think about what happens if you are caught. Would your girlfriend stay at your side? Even if her father speaks a hard word? Do you want to see your parents cry? Thrown from your school/university/job?

Don't give this a chance to happen!

If even this is not enough to motivate you: KEEP AWAY FROM HACKING! You are a danger to the whole hacking society and your friends !

4. STAY PARANOID

I hope you learned now why it is important to become paranoid. So stay paranoid. One mistake or lazy moment could suffice to ruin your life or career.

Always maintain motivation to do it.

III. BASICS

CONTENTS:

1. PREFACE

You should know this and practice it before you start your first hack. These are the absolute basics, without them you are in trouble soon. Even an experienced hacker can find a new hint/info in here.

2. SECURE YOURSELF

What if a SysAdmin reads your email?
What if your phone calls are recorded by the police?
What if the police seizes your computer with all your hacking data on it?

If you don't receive suspicious email, don't talk about hacking/phreaking on the phone and haven't got sensitive/private files on your harddisk then you don't need to worry. But then again you aren't a hacker. Every hacker or phreaker must keep in touch with others and have got his data saved somewhere.

Crypt every data which is sensitive! Online-Harddisk-Crypter are very important and useful:

There are good harddisk crypters free available an the internet, which behave fully transparent to your operating systems, i.e. the packages listed below are tested and were found to be a hacker's first-choice:

•If you use MsDos get SFS v1.17 or SecureDrive 1.4b •If you use Amiga get EnigmaII v1.5 •If you use Unix get CFS v1.33

File Crypters: You can use any, but it should use one of the well known and secure algorythms. NEVER use a crypting program which can be exported because their effective keylengths are reduced!

•Triple DES •IDEA •Blowfish (32 rounds)

Encrypt your emails!

•PGP v2.6.x is used most so use it too.

Encrypt your phonecalls if you want to discuss important things.

•Nautilus v1.5a is so far the best

Encrypt your terminal sessions when connected to a unix system. Someone might be sniffing, or monitoring your phone line.

•SSH is the so far most secure •DES-Login is fine too

Use strong passwords, non-guessable passwords which are not mentioned in any dictionary. They should seem random but good to remember for yourself. If the keylength is allowed to be longer than 10 chars, use that, and choose a sentence from a book, slightly modified. Please crypt phonenumbers of hacker friends twice. And call them from payphones/officephones/etc. only, if you don't encrypt the conversation.

The beginner only needs PGP, a filecrypter and an online-hardisk-crypter. If you are really deep into hacking remember to encrypt everything.

Make a backup of your data (Zip-Drive, other harddisk, CD, Tape), crypted of course, and store it somewhere which doesn't belong to any computer related guy or family member and doesn't belong to your house. So if a defect, fire or fed raid occures you got a backup of your data.

Keep written notices only as long as you really need them. Not longer. Keeping them in an encrypted file or on an encrypted partition is much more secure. Burn the papers once you don't need them anymore. You can also write them down with a crypt algorythm which only you know of, but don't tell others and don't use it too often or it can be easily analyzed and broken.

Really hardcore or ultra paranoid hackers should consider too the TEMPEST Project. Cops, spies and hackers could monitor all your doings. A well equipted man could have anything he wants : Electronic pulse emanation can be catched from more than 100 meters away and show your monitor screen to somebody else, a laserpoint to your window to hear private conversations, or identifying hifrequency signals of keyboard clicks ... so possiblities are endless Lowcost prevention can be done by electronic pulse jammers and therelike which become available on the public market, but I don't think this is secure enough to keep anyone dedicated away.

3. YOUR OWN ACCOUNT

So let's talk about your own account. This is your real account you got at your school/university/job/provider and is associated with your name. Never forget to fail these rules:

Never do any illegal or suspicious things with your real accounts! Never even try to telnet to a hacked host! Security mailing lists are okay to read with this account. But everything which seems to have to do with hacking must be either encrypted or be deleted as once. Never leave/save hacking/security tools on your account's harddisk. If you can, use POP3 to connect to the mailserver and get+delete your email (or do it in an other way if you are experienced enough using unix) Never give out your real email if your realname is in your .plan file and/or geco field (remember the EXPN command from sendmail ...) Give it only to guys who you can trust and are also security conscious, because if they are caught you may follow (or if it's a fed, not a hacker) Exchange emails with other hackers only if they are encrypted (PGP) SysAdmins OFTEN snoop user directories and read other's email! Or another hacker might hack your site and try to get your stuff!

Never use your account in a way which shows interest in hacking. Interest in security is okay but nothing more.

4. THE LOGS

There are 3 important log files:

WTMP - every log on/off, with login/logout time plus tty and host
UTMP - who is online at the moment
LASTLOG - where did the logins come from

There exist others, but those will be discussed in the advanced section. Every login via telnet, ftp, rlogin and on some systems rsh are written to these logs. It is VERY important that you delete yourself from those logfiles if you are hacking because otherwise they

a) can see when did you do the hacking exactly
b) from which site you came
c) how long you were online and can calculate the impact

NEVER DELETE THE LOGS! It's the easiest way to show the admin that a hacker was on the machine. Get a good program to modify the logs. ZAP (or ZAP2) is often mentioned as the best - but in fact it isn't. All it does is overwriting the last login-data of the user with zeros. CERT already released simple programs which check for those zero'ed entries. So thats an easy way to reveil the hacker to the admin too. He'll know someone hacked root access and then all you work was worthless. Another important thing about zap is that it don't report if it can't find the log files - so check the paths first before compiling! Get either a program which CHANGES the data (like CLOAK2) or a really good one which DELETES the entries (like CLEAR).

Normally you must be root to modify the logs (except for old distributions which have got utmp and wtmp world-writable). But what if you didn't made it hacking root - what can you do? Not very much : Do a rlogin to the computer you are on, to add a new unsuspicous LASTLOG data which will be displayed to the owner when he logs on next time. So he won't get suspicious if he sees "localhost". Many unix distributions got a bug with the login command. When you execute it again after you logged already on, it overwrites the login-from field in the UTMP (which shows the host you are coming from!) with your current tty.

Where are these log files by default located? That depends on the unix

distribution.

UTMP : /etc or /var/adm or /usr/adm or /usr/var/adm or /var/log
WTMP : /etc or /var/adm or /usr/adm or /usr/var/adm or /var/log
LASTLOG : /usr/var/adm or /usr/adm or /var/adm or /var/log

on some old unix dists the lastlog data is written into $HOME/.lastlog

5. DON'T LEAVE A TRACE

I encountered many hackers who deleted themselves from the logs. But they forgot to erase other things they left on the machines : Files in /tmp and $HOME

Shell History
It should be another as you current login account uses. Some shells leave a history file (depends on enviroment configuration) with all the commands typed. Thats very bad for a hacker. The best choice is to start a new shell as your first command after logging in, and checking every time for a history file in you $HOME. History files :

sh:.sh_historycsh:.historyksh:.sh_historybash:.bash_historyzsh:.history

Backup Files :

dead.letter, *.bak, *~

In other words: do an "ls -altr" before you leave!

Here're 4 csh commands which will delete the .history when you log out, without any trace.

```
mv .logout save.1
echo rm .history>.logout
echo rm .logout>>.logout
echo mv save.1 .logout>>.logout
```

6. THINGS YOU SHOULD AVOID

Don't crack passwords on an other machine than your own, and then only on a crypted partition. If you crack them on a e.g. university and the root sees your process and examines it not only your hacking account is history but also the site from which the password file is and the university will keep all eyes open to watch out for you. Download/grab the passwd data and crack them on a second computer or in a background process. You don't need many cracked accounts, only a few.

If you run important programs like ypx, iss, satan or exploiting programs then rename them before executing or use the small common source to exchange the executed filename in the process list ... ever security conscious user (and of course admin) knows what's going on if he sees 5 ypx programs running in the background ... And of course if possible don't enter parameters on the command line if the program supports an interactive mode, like telnet. Type "telnet" and then "open target.host.com" ... which won't show the target host in the process list as parameter.

If you hacked a system - don't put a suid shell somewhere! Better try to install some backdoors like ping, quota or login and use fix to correct the atime and mtime of the file if you don't have got another possiblity.

IV. ADVANCED

CONTENTS:

1. PREFACE

Once you installed your first sniffer and begin to hack worldwide then you should know and use these checks & techniques! Use the tips presented here - otherwise your activity will be over soon.

2. PREVENT TRACING OF ANY KIND

Sometimes your hacking will be noticed. Thats not a real problem - some of your sites will be down but who cares, there are enough out there to overtake. The very dangerous thing is when they try to trace you back to your origin - to deal with you - bust you!

This short chapter will tell you every possiblity THEY have to trace you and what possibilities YOU have to prevent that.

1. Normally it should be no problem for the Admin to identify the system the hacker is coming from by either:

•checking the log entries; if the hacker was really lame, •taking a look at the sniffer output the hacker installed and he's in too, •any other audit software like loginlog, •or even show all estrablished connections with "netstat" if the hacker is currently online

- expect that they'll find out! Thats why you need a gateway server.

2. A gateway server in between - what is it? Thats one of many many servers you have accounts on, which are absolutely boring systems and you have got root access on. You need the root access to alter the wtmp and lastlog files plus maybe some audit logs do nothing else on these machines! You should change the gateway servers on a regular basis, say every 1-2 weeks, and don't use them again for at least a month. With this behaviour it's unlikely that they will trace you back to your next point of origin : the hacking server.

3. Your Hacking Server - basis of all activity From these server you do begin hacking. Telnet (or better : remsh/rsh) to a gateway machine and then to the target. You need again root access to change the logs. You should change your hacking server every 2-4 weeks.

4. Your Bastian/Dialup server. This is the critical point. Once they can trace you back to your dialup machine you are already fried. A call to the police, a line trace and your computer hacking activity is history - and maybe the rest of your future too. You *don't* need root access on a bastion host. Since you only connect to it via modem there are no logs which must be changed. You should use a different account to log on the system every day, and try to use those which are seldom used. Don't modify the system in any way! You should've got at least 2 bastion host systems you can dialup to and switch between them every 1-2 month.

Note: If you have got the possiblity to dialup different systems every day (f.e. due blueboxing) then do so. you don't need a hacking server then.

5. Do bluebox/card your call or use an outdial or any other way. So even when they capture back your bastion host, they can't trace you (easily) ... For blueboxing you must be cautious, because germany and the phone companies in the USA do have surveillance systems to detect blueboxers ... At&t traces fake cred card users etc. Using a system in between to transfer your call does on the one side make tracine more difficult - but also exposes you to the rish being caught for using a pbx etc. It's up to you. Note too that in f.e. Denmark all - ALL - calling data is saved! Even 10 years after your call they can prove that *you* logged on the dialup system which was used by a hacker ...

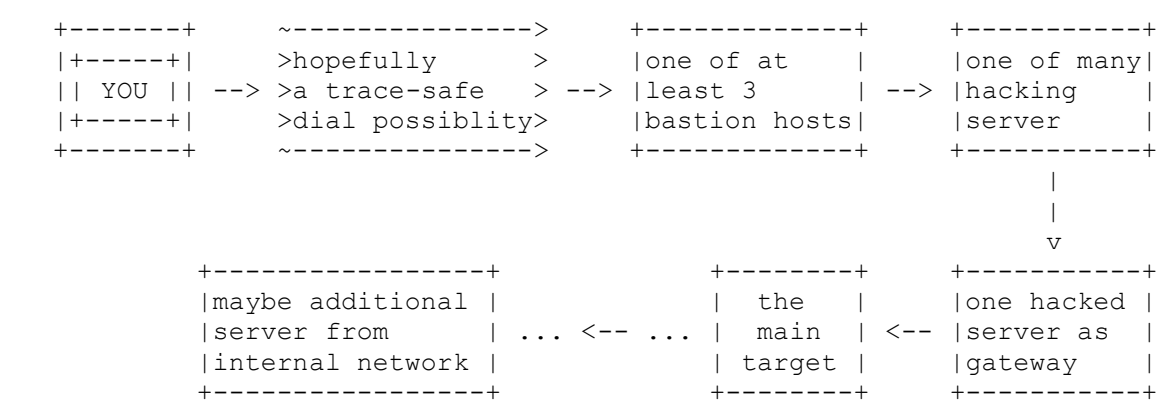
6.Miscellaneous

If you want to run satan, iss, ypx, nfs filehandle guessing etc. then use a special server for this. don't use it to actually telnet/rlogin etc. to a target system, only use it for scanning. Connect to it as if it were a gateway server.

Tools are out there which binds to a specific port, and when a connection is established to this port, it's automatically opening a connection to another server some other just act like a shell on the system, so you do a "telnet" from this socket daemon too. With such a program running you won't be written in any log except firewall logs. There are numerous programs out there which do that stuff for you.

If possible, the hacking server and/or the gateway machine should be located in a foreign country! Because if your breakin (attempt) was detected and your origin host identified then most admins will tend to give up to hunt after you. Even if the feds try to trace you through different countries it will delay them by at least 2-10 weeks ...

CONCLUSION: If you hack other stuff than univerisities then do it this way! Here is a small picture to help you ;-)



3. FIND AND MANIPULATE ANY LOG FILES

It's important that you find all logfiles - even the hidden ones. To find any kind of logfiles there are two easy possibilities:

•Find all open files.
Since all logfiles must write somewhere, get the cute program LSOF - LiSt

Open Files - to see them ... check them ... and if necessary correct them.

•Search for all files changed after your login.
After your login do a "touch /tmp/check" then work on. Later just do a "find / -newer /tmp/check -print" and check them if any of those are audit files. see>check>correct. Note that not all versions of find support the -newer option You can also do a "find / -ctime 0 -print" or "find / -cmin 0 -print" to find them.

Check all logfiles you find. Normally they are in /usr/adm, /var/adm or /var/log. If things are logged to @loghost then you are in trouble. You need to hack the loghost machine to modify the logs there too ...

To manipulate the logs you can either do things like "grep -v", or do a linecount with wc, and then cut off the last 10 lines with "head -LineNumbersMinus10", or use an editor etc. If the log/audit files are not textfiles but datarecords ... identify the software which writes the logfiles. Then get the sourcecode. Then find the matching header file which defines the structure of the file. Get zap, clear, cloak etc. and rewrite it with the header file to use with this special kind of logfile (and it would be kind to publish your new program to the hacker society to safe others much work)

If accouting is installed then you can use the acct-cleaner from zhart, also in this release - it works and is great!

A small gimmick if you must modify wtmp but can't compile a source and no perl etc. is installed (worked on SCO but not on linux) : Do a uuencode of wtmp. Run vi, scroll down to the end of the file, and and delete the last 4 (!) lines beginning with "M" ... then save+exit, uudecode. Then the last 5 wtmp entries are deleted ;-)

If the system uses wtmpx and utmpx as well you are in trouble ... I don't know any cleaner so far who can handle them. Program one and make it available for the scene.

4. CHECK THE SYSLOG CONFIGURATION AND LOG

Most programs use the syslog function to log anything they want. It's important to check the configuration where syslog does print special types. The config file is /etc/syslog.conf - and I won't tell you here what the format is and what each entry means. Read the manpages about it. Important for you are kern.*, auth.* and authpriv.* types. Look where they are written too: files can be modified. If forwarded to other hosts you must hack those too. If messages are sent to a user, tty and/or console you can do a small trick and generate false log messages like "echo 17:04 12-05-85 kernel sendmail[243]: can't resolve bla.bla.com > /dev/conso
or whichever device you want to flood so that the message you want to hide simply scrolls over the screen. These log files are very important! Check them.

5. CHECK FOR INSTALLED SECURITY PROGRAMS

On most security conscious sites, there are security checkers run by cron. The normal directory for the crontabs are /var/spool/cron/crontabs. Check out all entries, especially the "root" file and examine the files they run. For just a fast investigation of the crontabs of root type "crontab -l root".

Some of those security tools are most time also installed on the admins' accounts. Some of them (small utils to check wtmp, and if a sniffer is installed) are in their ~/bin. Read below to identify those admins and check their directories.

Internal checking software can be tiger, cops, spi, tripwire, l5, binaudit, hobgoblin, s3 etc.

You must examine them what they report and if they would report something that would be a sign of your breakin. If yes you can

•update the data files of the checker (learn mode) so that it won't report that type anymore •reprogram/modify the software so that they don't report it anymore. (I love fake cpm programs ;-) •if possible remove the e.g. backdoor you installed and try to do it in another way.

6. CHECK THE ADMINS

It is important for you to check the sysops for the security counter-measures they do - so first you need to know which normal accounts are they use. You can check the .forward file of root and the alias entry of root. Take a look into the sulog and note those people who did a successful su to root. Grab the group file and examine the wheel and admin group (and whatever other group are in this file which are related to administration). Also grep'ing the passwd file for "admin" will reveile the administrators.

Now you should know who the 1-6 administrators on the machines are. Change into their directories (use chid.c, changeid.c or similar to become the user if root is not allowed to read every file) and check their .history/.sh_history/.bash_history to see what commands they type usually. Check their .profile/.login/.bash_profile files to see what aliases are set and if auto-security checks or logging are done. Examine their ~/bin directory! Most times compiled security checking programs are put there! And of course take a look into each directory they've got beside that (ls -alR ~/). If you find any security related stuff, read 5.) for possibilities to bypass those protections.

7. HOW TO "CORRECT" CHECKSUM CHECKING SOFTWARE

Some admins really fear hacker and install software to detect changes of their valuable binaries. If one binary is tampered with, next time the admin does a binary check, it's detected. So how can you

a.find out if such binary checkers are installed and b.how to modify them so you can plant in your trojan horse?

Note that there are many binary checker out there and it's really easy to write one - takes only 15 minutes - and can be done with a small script. So it's hard to find such software if it's installed. Note that internal security checking software sometimes also support such checking. Here are some widely used ones :

SOFTWARE STANDARD PATHBINARY FILENAMES:

tripwire/usr/adm/tcheck
/usr/local/adm/tcheckdatabases
/usr/local/adm/audit

But as you can see there are too much possibilities! The software or database could even be on an normally unmounted disk or NFS exported partition of another host. Or the checksum database is on a write protected medium. There are too much possibilities. But normally you can just do the fast check if the above packages are installed and if not go on exchanging binaries. If you don't find them but it actually is a very well secured

site then you should NOT tamper with the binaries! They sure have got them hidden very well.

But what do you do when you find that software installed and you can modify them (e.g. not a write protected medium, or something that can be bypasswd - for example unmounting the disk and remounting writable)? You've got 2 possibilities :

•First you can just check the parameters of the software and run an "update" on the modified binary. For example for tripwire that's "tripwire -update /bin/target".

•Seconds you can modify the filelist of the binaries being checked - removing the entry of the replaced one. Note that you should also check if the database file itself is checked too for changes! If yes - update/delete the entry as well.

8. USER SECURITY TRICKS

This is a rare thing and is only for sake of completeness. Some users, named admins and hackers, usually don't want their own accounts to be used by someone else. That's why they sometimes put some security features into their startup files.

So check all dotfiles (.profile, .cshrc, .login, .logout etc.) what commands they execute, what history logging and which searchpath they set. If f.e. $HOME/bin comes before /bin in the search path you should check the contents of this directory ... maybe there's a program called "ls" or "w" installed which logs the execution time and after that executing the real program.

Other check automatically the wtmp and lastlog files for zap usage, manipulation of .rhosts, .Xauthority files, active sniffers etc. Never mess with an account a unix wizard is using!

9. MISCELLANEOUS

Finally, before some last words about being under suspect or caught, here are some miscellaneous things which a worth to take a notice off.

Old telnet clients do export the USER variable. An administrator who knows that and modified the telnetd can get all user names with that and so identify the account you are hacking from, once he notices you. The new clients have been fixed - but a clever admin has got other possiblities to identify the user : the UID, MAIL and HOME variables are still exported and makes identifying of the account used by the hacker easy. Before you do a telnet, change the USER, UID, MAIL and HOME variable, maybe even the PWD variable if you are in the home directory.

On HP-UX < v10 you can make hidden directories. I'm not talking about . (dot) files or similar but a special flag. HP introduced it v9, but was removed from version 10 (because it was only used by hackers ;-). If you do a "chmod +H directory" it's invisible for the "ls -al". To see the hidden directories you need to add the -H switch to ls, e.g. "ls -alH" to see everything.

Whenever you are in need to change the date of a file, remember that you can use the "touch" command to set the atime and mtime. You can set the ctime only by raw writes to the harddisk ...

If you install sniffer and it's an important system, then make sure that you either obfusicate the sniffer output (with an encryption algorythm [and i'm not talking about rot13] or let the sniffer send all the captured

data via icmp or udp to an external host under your control. Why that? If the admin finds somehow the sniffer (cpm and other software checking for sniffers) they can't identify in the logfile what data was sniffed, so he can't warn hosts sniffed by you.

V. UNDER SUSPECT

Once you are under suspect (by either police and/or administrator) you should take special actions so they won't get evidence on you.

> NOTE : If the administrators think you are a hacker,
> YOU ARE GUILTY UNTIL PROVEN INNOCENT

The laws means nothing to the admins (sometimes I think the difference between a hacker and an administrator is only that the computer belongs to them). When they think you are a hacker you are guilty, without a lawyer to speak for you. They'll monitor you, your mails, files, and, if they are good enough, your keystrokes as well.

When the feds are involved, you phone line might be monitored too, and a raid might come soon.

If you notice or fear that you are under suspect then keep absolutely low profile! No offensive action which points to hacking should be done.

Best thing is to wait at least 1-2 month and do nothing. Warn your friends not to send you any email, public normal only, non-offensive mail is wonderful, put pgp encrypted emails will ring the alarm bells of monitoring admins and feds. Cut down with everything, write some texts or program tools for the scene and wait until things have settled. Remember to encrypt all your sensitive data and remove all papers with account data, phone numbers etc. Thats the most important stuff the feds are looking for when they raid you.

VI. CAUGHT

Note that this small chapter covers only the ethics and basics and hasn't got any references to current laws - because they are different for every country.

Now we talking about the stuff you should/shouldn't do once the feds visited you. There are two very important things you have to do:

1. GET A LAWYER IMMEDEATELY! The lawyer should phone the judge and appeal against the search warrant. This doesn't help much but may hinder them in their work. The lawyer should tell you everything you need to know what the feds are allowed to do and what not. The lawyer should write a letter to the district attorney and/or police to request the computers back as fast as possible because they are urgently needed to do business etc. As you can see it is very useful to have got a lawyer already by hand instead of searching for one after the raid.

2. NEVER TALK TO THE COPS! The feds can't promise you anything. If they tell you, you'll get away if you talk, don't trust them! Only the district attorney has got the power to do this. The cops just want to get all information possible. So if you tell them anything they'll have got more information from and against you. You should always refuse to give evidence - tell them that you will only talk with them via your lawyer.

Then you should make a plan with your lawyer how to get you out of this shit and reduce the damage. But please keep in mind : don't betray your friends. Don't tell them any secrets. Don't blow up the scene. If you do,

that's a boomerang : the guys & scene will be very angry and do revenge, and those guys who'll be caught because of your evidence will also talk ... and give the cops more information about your crimes!

Note also that once you are caught you get blamed for everything which happened on that site. If you (or your lawyer) can show them that they don't have got evidences against you for all those cases they might have trouble to keep the picture of that "evil hacker" they'll try to paint about you at the court. If you can even prove that you couldn't do some of the crimes they accuse you for then your chances are even better. When the judge sees that false accuses are made he'll suspect that there could be more false ones and will become distrusted against the bad prepared charges against you.

I get often asked if the feds/judge can force you to give up your passwords for PGP, encrypted files and/or harddisks. That's different for every country. Check out if they could force you to open your locked safe. If that's the case you should hide the fact that you are crypting your data! Talk with your lawyer if it's better for you to stand against the direction to give out the password - maybe they'd get evidences which could you get into jail for many years.

(For german guys : THC-MAG #4 will have got an article about the german law, as far as it concerns hacking and phreaking - that article will be of course checked by a lawyer to be correct. Note that #4 will only discuss germany and hence will be in the german language. But non-germans, keep ya head up, this will be the first and last german only magazine release ;-)

VII. PROGRAMS

Here is a small list of programs you should get and use (the best!). DON'T email me where to get them from - ask around in the scene! I only present here the best log modifiers (see III-4 and IV-3). Other programs which are for interest are telnet redirectors (see IV-2) but there are so many, and most compile only on 1-3 unix types so there's no use to make a list.

First a small glossary of terms: Change - changes fields of the logfile to anything you want. Delete - deletes, cuts out the entries you want. Edit - real editor for the logfile. Overwrite - just overwrites the entries with zero-value bytes. (Don't use overwriters (zap) - they can be detected!)

LOG MODIFIERS:

ah-1_0b.tar	Changes the entries of accounting
informationclear.c	Deletes entries in utmp, wtmp, lastlog and wtmp
xcloak2.c	Changes the entries in utmp, wtmp and lastlog
invisible.c	Overwrites utmp, wtmp and lastlog with predefines values, so it's better than zap.

Watch out, there are numerous inv*.c !marryv11.c
Edit utmp, wtmp, lastlog and accounting data - best!

wzap.c	Deletes entries in wtmp
wtmped.c	Deletes entries in wtmp
zap.c	Overwrites utmp, wtmp, lastlog - Don't use! Can be detected!

VIII. LAST WORDS

Last fucking words: Don't get caught, remember these tips and keep your ears

dry. If someone would like to correct some points, or would like to add a comment, or needs more information on a topic or even thinks something's missing - then drop me a note.