Computer Security at Concordia: Past Problems, Proposed Plans[*]

Anne Bennett
IITS
Concordia University
anne@alcor.concordia.ca

Michael J. Assels
Computer Science
Concordia University
mjassels@cs.concordia.ca

August 4, 1998

Abstract

Computer security has been a problem since the 1970s, and unfortunately, many of the problems described then are still present on today's systems. However, with the popularization of the Internet, and with the increasing internetworking of computers which perform important or sensitive functions, we find that threats to the security of our computer installations are no longer theoretical and remote possibilities, nor at this point do they remain exciting but isolated incidents *à la* "Cuckoo's Egg". On the contrary, they have graduated to the mundanity of everyday annoyance, yet they have potentially serious consequences for Concordia University.

In this report, we describe a series of incidents which took place during the summer of 1995, we provide a brief summary of elementary concepts in computer security, and we propose some directions the University can take to decrease the risks to its computer installations and the data they contain.

It is our hope that after reading this report, members of the University community will understand why computer security issues must be addressed, and will offer their enthusiastic cooperation for the implementation of the measures proposed in this document.

[*] Thanks to Steven Winikoff, Mike Marak, Frank Maselli, and Wendy French of IITS, Paul Gill, Larry Thiel, and Chairperson Clement Lam of Computer Science, Steve Robbins, der Mouse, Tom Levasseur, and Jacek Slaboszewicz of McGill University, sysadmins at several sites whose identities have been disguised to protect their reputations, and most of all, Corporal Bob Beaulieu of the RCMP, for their assistance with incident investigations.

Contents

Part I

How I Spent my Summer Vacation.

by Michael Assels

In late July 1995, Steve Robbins, a colleague at McGill, complained that a ne'er-do-well from our site had been making unauthorized use of an account at McGill to run a password cracking program. The scoundrel had not been positively identified, but had been using the account "jsmith" on our machine named "sunset" to launch his attack against McGill. McGill had closed the compromised account, but some damage had already been done: evidence had been found that the password cracker had successfully figured out the passwords of several users at McGill and truckloads of users at Concordia. Steve kindly passed on the list of cracked accounts, and left the matter in our hands.

We — which usually means Paul Gill and I — looked into the "jsmith" account and immediately saw traces of unusual activities. A chat with the real jsmith confirmed what everyone suspected: he wasn't the villain; the real bad guy was just using his account, which was on the cracked list from McGill.

Fortunately, the intruder had left enough "junk" lying around in jsmith's account to give us some idea of his habits, and some clues about things to look for in other places. When we looked at the other accounts on the cracked list, we found that several had been used by the intruder, while most had not. What was most interesting, though, was that there were *two* patterns of activity: sometimes the intruder came in by phone; sometimes he sat at a terminal in the Hall Building. When he came by phone, he liked to hop over to a university in the Southern U.S. — let's call it "Sussex". When he sat at a terminal, on the other hand, he liked to visit McGill. In both cases, files were left scattered about containing credit card numbers, calling card numbers, 1-800 numbers (presumably for compromised PBX[1] equipment), and pirated software.

It looked as though we had not one, but at least two intruders, and they were clearly up to no good.

We disabled most of the accounts identified by McGill as having been cracked; it would certainly not have been appropriate to let criminals run free with them, especially when the compromised accounts included those of three senior professors, as well as two accounts used for such administrative purposes as grading courses. These accounts might have had very sensitive contents. There were others, though, that were old unused accounts without special privilege. These, we decided, could

[1] Private Branch Exchange: telephone switching equipment.

be left open, but booby-trapped, so that if the intruders used them, we could see everything they were up to.

It didn't take long. As soon as "their" current accounts were closed, the bad guys switched to new ones. Good news: now we could follow them more closely. Bad news: only one of them chose from our collection of booby-trapped accounts. The other came in promptly as the secretary of the Graduate Studies Programme, who had *not* been on the cracked list, so the list was apparently incomplete. We quickly blocked the secretary's account, and within minutes, our culprit was using one of the booby-trapped accounts.

Now we watched, and waited for a chance to identify the intruders.

On one occasion, Paul saw that one of the bad guys was using a terminal in our lab in the Hall Building. Paul hurried over to the lab and confronted a rather large young man at the terminal, asking to see his identification. The man refused and hurried away.

On another occasion, seeing that the other cracker was coming in by phone and staying for an unusually long time, we tried to use internal resources to find where he was coming from. Mike Marak of the Computing Services[2] Network Engineering Group contacted Wendy French of Telesis, who was able by obscure means to find that the number at the other end of the connection was in an Ontario Government office building in Toronto! We guessed that this wasn't the real origin of the call.

In August, the RCMP were brought into the case. Cpl. Bob Beaulieu of the Computer Crime Section took responsibility for the case. He asked us to send the RCMP a formal letter of complaint from a person with suitable authority, and to designate technical people to help in the investigation and prosecution of the case. Without at least this level of commitment from the University, the police couldn't proceed.

Orally, we were quick to agree. A letter would be sent, and Paul Gill and I were assigned to provide technical help. It took longer than we had hoped for a formal complaint to go out, but we managed to convince Cpl. Bob to get started anyway, because we had evidence that the intruders were involved in crimes affecting victims other than Concordia: the credit card and calling card numbers.

Cpl. Bob wanted us to be very quiet about the investigation, not informing anyone who didn't already know about it. Word spreads fast in the electronic world, and without secrecy, the bad guys might quickly come to know they were under investigation. Moreover, we were not sure where they had friends. Our traces already suggested that they had free run of Sussex's Computer Science labs, and it was pos-

[2] Computing Services is now called Instructional and Information Technology Services (IITS), after its merge in 1997 with the Audio-Visual Department; however, in this section, we will continue to use the name as it existed during the events we describe.

sible that there might be a “rogue” system manager there helping them out. Even McGill wasn’t known to be clean. They had had several recent intrusions, and it wasn’t clear which systems were being reliably managed and secure, and which were under “enemy” control.

So we began our investigation in secrecy. We had two objectives: to identify the intruders, and to gather incriminating evidence. We were already pretty sure that there were at least two intruders; one on the phone, who used the IRC[3] nickname “Shrunk”, and one in person, who used several nicknames, including “Blackrat”. We didn’t yet have any indication of their real names. Cpl. Bob asked us to call his pager whenever either of them came in.

Blackrat was easy to identify: he would come into the Hall Building and spend hours there, either in H966 or in H511. We called the Mounties twice, and each time they came to make a visual identification of the suspect as he typed at a terminal. They didn’t arrest him, though, because they didn’t want to scare off Shrunk; and without arresting him, they couldn’t attach a real name to him, so he remained “Blackrat”.

Before we got a chance to identify Shrunk, we had some problems. First, Blackrat took a look at his “.login” configuration file and saw a line he didn’t quite understand, so he deleted it. That was the booby-trap! Now Blackrat was using our machines and we didn’t know what he was up to. Paul and I discussed our options and decided that it was too risky to put the line back: if he noticed it again he’d know we were onto him. Instead, we wrote a quick and dirty program that would capture his session from the network whenever he logged into any of his favourite sites. He could still operate unobserved on our local machines, but he usually did his dirty work at McGill anyway, so we wouldn’t miss much.

As soon as that crisis was resolved, another arose. Blackrat and Shrunk met on IRC, and Blackrat gave Shrunk an account at McGill, which was “better” than ours because it wasn’t restricted by our very tight undergraduate disk quota. Shrunk immediately disappeared from our network. We could only assume that he was now using “his” McGill account, which we couldn’t trace at all.

At this point I decided that it was time to stop overplaying the secrecy game. I tried to call Steve Robbins, only to discover that he was no longer working at McGill. Instead, I called Anne Bennett, an analyst in Computing Services, and asked her to put me in touch with a reliable person at McGill. She gave me the number of Mike Parker, whom the Unix community knows as “der Mouse”.

Der Mouse was very cooperative. We briefed him on our situation and told him that we’d like him to shut our villains out of McGill so that they’d come back to

[3] Internet Relay Chat: a popular Internet “party line” where people can “chat” with other people by typing at their keyboards.

Concordia where we could keep track of them. We gave him all the information we had about compromised accounts at McGill. He promptly disabled the ones in his domain (CIM, the Centre for Intelligent Machines), and referred us to Jacek Slaboszewicz and Tom Levasseur in Electrical Engineering for the others. The EE guys made my day: they gave us a name for Blackrat. He was a former McGill student who had been expelled for academic reasons, but who had been known to engage in cracking activity before.

With the McGill door closed, the intruders came back "home" to Concordia, where we continued to amass a large volume of evidence against them. In fact, the most frustrating aspect of the project was sifting through the enormous logs to find snippets of significant, damning material. Among other things, we found that:

- Blackrat and Shrunk were actively working to crack the password files of Alcor and Vega (two Computing Services machines), the Computer Science Department, McGill, and another large Canadian university;

- Shrunk had an account at Sussex U under the name "shrunk" — a clear indication that the system manager was collaborating — and he ran an active pirated software exchange (a "Warez site") there;

- Shrunk obtained accounts at two commercial Internet service providers in the U.S. by giving fraudulent credit card information;

- Blackrat had an account at an Indiana Internet service provider under the name "blackrat" and appeared to collaborate in piracy with someone there named "elduet", who turns out to have been a crooked system administrator;

- Blackrat had a cracked account at a Southeastern U.S. academic service provider, on which he ran his password cracking program. We found many of our own passwords there!

- Blackrat traded child pornography with a mysterious "daveman" on IRC;

- Shrunk had — and traded — phone numbers and access codes for compromised PBX (telephone) equipment and voice-mail systems;

- Both had accounts at another Canadian university — let's call it "Essex" — where they and other bad guys were running a pirated software exchange.

The Essex case was particularly disturbing. We watched an IRC session in which

Shrunk was told how to break into the AIX[4] operating system and obtain root[5] privilege. Then we watched Shrunk do just that. And we saw Blackrat do the same. Apparently, every cracker on the Internet could get into Essex's machines as root. (We later learned that Essex had just moved their MIS system to AIX machines, so these would have been vulnerable too!)

This was too dangerous to let go. We contacted a reliable person at Essex, and soon we saw all of their machines go down for a day. When they came back up, they were no longer vulnerable. It was great fun watching the villains arguing amongst themselves on IRC as to who had ruined their cosy setup at Essex.

We eventually got lucky with Shrunk's identity. We intercepted a "private" IRC conversation between Shrunk and another bad guy in which Shrunk invited his friend to call him at home, and provided a phone number. This was quite a break. I called Frank Maselli of Computing Services, who was able to look up the number in a Canada-wide telephone number database and associate it with a name and address in a Toronto suburb. (Shrunk is a young offender, so we can't identify him in any detail here.)

A nice pointer, but not the hard evidence that Cpl. Bob would need. For that, we needed a live phone trace. Cpl. Bob asked us to call his pager as soon as Shrunk logged in, day or night. We tried that several times before Shrunk obliged by staying on long enough for a trace. In the movies, it takes two minutes. In this case, we needed more than an hour, because Shrunk played a mean game of phone tag.

Eventually, though, he had to stay on for a long time to download a big batch of pirated software. When I saw the connection, I called Cpl. Bob's pager. He called back within two minutes, and I told him we had a live connection. I confirmed that it was indeed Shrunk, and identified the telephone number to which he had connected. Cpl. Bob made a conference call to Bell Canada's Montreal security office to ask for a trace. After a few minutes, Bell's technical people came back with the origin of the call: it was coming from a number belonging to Unitel on rue Notre Dame Ouest. That was all they could give us. We needed to contact Unitel's security people for more details.

Unitel was more helpful. They identified the Notre Dame number as the Montreal end of a trunk line between Montreal and Toronto, and with some difficulty they picked up the trace and followed it back to the Toronto end. It was coming from a "secure" PBX belonging to the Government of Ontario, located in the Provincial Parliament Buildings in Queen's Park.

Cpl. Bob and I guessed that this was not the work of Premier Mike Harris! So

[4] AIX: a brand of Unix operating system distributed by IBM for use on their machines.

[5] Root (also known as "superuser" or system administrator) privilege allows the user to examine and modify any file on the computer, including user files, and including the files which make up the operating system of the machine.

we went to yet another conference call, this time to the technical person in charge of the PBX. He had to do some digging, but he finally came up with a number. To nobody's surprise, it was the same Toronto number that we'd seen before. That was what Cpl. Bob needed.

Over the next few days, the Mounties in Toronto monitored all calls originating from Shrunk's house. There were hundreds. Almost all went through the Ontario Government PBX, and most were to Concordia's Computer Science Department.

Cpl. Bob obtained search warrants for the homes of both Shrunk and Blackrat. On Thursday, September 7, he called me to say the warrants would be executed the following Tuesday. Unfortunately, as he was speaking, I was looking at a trace of a Blackrat session from the previous evening, in which he launched attacks against the X servers[6] of Tom Levasseur at McGill and of a system administrator in the VLSI Lab at Concordia. He was capturing every keystroke they made, using an ancient security hole in the X server. It was only a matter of time until one of them would type the root password, then Blackrat would have root privileges!

I told Cpl. Bob that we couldn't reasonably wait more than another day before shutting the villains down. They had become too dangerous. He agreed, and moved the date up to Friday: the next day. He would go to Toronto to arrest Shrunk, and his colleagues would arrest Blackrat in Montreal. In the meantime, I called Tom and the local sysadmin[7] to let them know that their keystrokes were being snooped, so they shouldn't type anything sensitive at their terminals until further notice.

On Friday at noon, Cpl. Bob was at the Bowmanville RCMP Detachment — the nearest one to Shrunk's house. Two of his colleagues were in the Hall Building, and a third was with me at my desk, watching for the culprits to come in. True to form, Blackrat arrived at about 12:30, sat at a terminal in H511, and began his daily routine of bad behaviour. His login tripped an alarm on my terminal, and the two Hall Building Mounties took up positions outside H511. They didn't arrest him immediately, though, because we hoped to catch the two crackers communicating together.

At about 1:00pm, Shrunk came in over the phone lines, again setting off an alarm on my terminal. I called Cpl. Bob's cellular phone, and he immediately started towards Shrunk's house. Meanwhile, I kept an eye on the trace of Shrunk's session. He had started an FTP[8] session to an illegal software depot and was browsing in search of something interesting. This was a good sign. He'd be on for a while.

[6] X: a "window system" whereby a program running on one computer can display its output on, and obtain input from, a screen, keyboard, and mouse on a different computer. The X window system is widely used on Unix machines.

[7] Sysadmin: shorthand for "system administrator", the technical person in charge of a computer system.

[8] File Transfer Protocol: a means of transmitting data files over the computer network.

Unfortunately, before Cpl. Bob arrived, and before we noticed that Shrunk was logged in, Blackrat logged out and left H511. The Mounties arrested him in the hallway. One down.

Cpl. Bob was disappointed that we couldn't get the bad guys talking to each other, but those are the breaks. With the cell phone connection still open, he and his backup officer arrived at Shrunk's house at about 1:15pm, search warrant in hand, and rang the bell. The door was answered by Shrunk's mother, who called her son from upstairs. The keystrokes in Shrunk's FTP session stopped.

Cpl. Bob: Are you (Shrunk's name)?
Shrunk: Yes.
Cpl. Bob: Where is your computer?
Shrunk: Upstairs.

They go upstairs.

Cpl. Bob: DON'T TOUCH THAT KEYBOARD!

The keystrokes now resumed:

```
ftp> This is Bob Beaulieu
?Invalid command
ftp>
```

With that, our loop was closed,[9]and we were done. Just in time to start the Fall term.

Epilogue: The Snails of Justice

Did I say we were done? I'm sorry, that's not quite right. We were done with the nail-biting aspect of the affair, but the hair-tearing had not yet begun.

Blackrat and Shrunk were taken to RCMP detachments in Montreal and Toronto, respectively, where they were photographed and fingerprinted. They were not charged immediately; the Crown would have to determine what charges should be laid in view of the evidence.

Shrunk, who was only 15 years old, would be tried under the *Young Offenders Act*. He didn't put up much of a fuss, though. His parents had been completely unaware

[9] Cpl. Beaulieu typed into Shrunk's session so that his keystrokes would appear in Michael's session log, thus making it easier to prove in court that the sessions captured at Concordia did indeed belong to the physical person who was arrested. Of course, the text "This is Bob Beaulieu" is not a valid command in the FTP program, which is why the program returns "?Invalid command".

of his activities, and were not at all pleased to find out about them in a police raid. Shrunk's father took charge of his son's case immediately, and made it very clear that the unfortunate boy would plead guilty to any charge and accept his sentence, whatever it might be. No lawyer would be needed, and no defence would be mounted. Shrunk Senior wanted to have the whole business behind him by Christmas — not an unreasonable hope, I thought, given his enthusiastically cooperative attitude.

Sadly, the justice system has a different way of approaching the matter. As this was the first case of its kind in Quebec, the Crown prosecutors were not quite sure what to do with it. The evidence was quite technical, so it was a struggle for them to understand what had actually happened, why it was wrong, and what law had been violated. The law is clear, though, and given enough time, even a lawyer can understand it. After a few months, it was decided that both Shrunk and Blackrat would be charged under Section 342.1 of the Criminal Code: "unauthorized use of a computer."

In April of 1996 — on time for Easter — Shrunk pleaded guilty to the one count with which he was charged. He was sentenced to six months' probation, community service, and confiscation of his computer for six months, but I suspect that his father's displeasure was a stiffer penalty than anything the courts could have imposed.

Blackrat was a tougher customer. He was quite uncooperative. In his view, if he *could* do something with a computer system, it must therefore be legal. So he refused to plead guilty to anything. As a result, the full fury of the law would be unleashed against him.

On the day of Blackrat's court appearance, Tom Levasseur and I arrived at the Palais de Justice at 9:00am and made our way to Room 406, where his case was to be heard. One by one, various petty criminal cases were processed. In each case, a lawyer would would appear with a calendar and ask that his client's case be deferred until such-and-such a date. The Crown would object that she would be out of town on that day, or perhaps the judge would be on vacation. After a minute or two of calendar shuffling a new date would be found, and the next case would come up. And so the session proceeded for two hours until Blackrat appeared. The judge asked if he was represented by counsel, at which his lawyer appeared and asked for a new date.

Over the next year, we would all learn just how inexorable the law can be. Anne Bennett, Tom Levasseur and I must have appeared ten times before any resolution was made. In fact, we appeared more often than the defendant himself, who seemed far less interested that we did. Each time, there was some reason to delay, but each time, it was necessary for the parties to make a full formal appearance before a judge to reschedule the hearing. The Palais de Justice apparently has no computer system for scheduling appointments. As a result, criminal lawyers seem to spend most of their time rustling the leaves of their calendars in full formal dress.

By the time Blackrat's case actually came to trial, he had finally been convinced to plead guilty to the three counts against him, but his lawyer and the Crown had been unable to agree upon an appropriate sentence. The Crown, at Corporal Bob's urging, wanted jail time — at least a suspended sentence. The defence wanted probation and community service, which would leave his client with no permanent criminal record.

The judge sided with the defence: Blackrat was sentenced to two years' probation and 180 hours' community service. *Now,* on April 23, 1997, we were done.

Part II

An Extremely Brief Course in Computer Security.

1 A few definitions.

We have just described a series of incidents in which computer security was clearly compromised, both at Concordia and at other sites. But what *is* computer security, and why should we care about it? A few brief definitions are in order.

The linchpins of computer security are:

- Availability,
- Confidentiality,
- Integrity,
- Non-repudiation,
- Access control, and
- Authentication.

Computing resources should be **available** as expected, which they are not if the computer has been damaged by crackers. The resources are also unavailable if the computer is not responding properly because it is performing unauthorized tasks, which happens in denial-of-service attacks, or when unauthorized people are using the system (often due to account sharing). In an environment where students, researchers, and employees depend on the availability of the computer to do their work, its unavailability can be very costly.

Sensitive information must be kept **confidential**, and indeed in some cases the University has a legal responsibility to ensure that it remains so.

The **integrity** of data must also be protected against accidental or malicious alteration or destruction. This applies not only to sensitive or confidential University data, but to valuable user data, and to the system itself in the form of its programs, operating system, and configuration files.

As electronic communications take over an increasing proportion of business and professional communication, **non-repudiation** is increasingly important. This means that the sender of a message should not be able to legitimately deny that she has sent it, the receiver should not be able to legitimately deny that he has received it, and neither party should be able to alter the contents of a message so validated.

Access control is a means whereby we allow only authorized entities to use our resources, and **authentication** is how we determine that an entity is what it claims to be.

2 Some common vulnerabilities and their consequences.

2.1 Technical problems.

Failures of access control and authentication are a common cause of loss of integrity, confidentiality, and availability of computer systems. In turn, the most common causes of failures in access control and authentication are:

2.1.1 System programming and configuration errors,

System programming and configuration errors, whereby programs access data they should not (*e.g.*: most of the sendmail bugs, bugs in other privileged programs, and careless system configuration), are especially difficult to combat when they originate in proprietary software to which we do not have the source code.

> *One of the most dangerous incidents in Michael's "summer vacation" was, from the affected site's point of view, the obtention of root privileges by the crackers on the machines at "Essex". Using the well-known*[10] *AIX bug "*`login -froot`*" to get in, the crackers then had full run of the machines. They could have damaged or subverted the MIS systems, and gained access to confidential data. This could have been disastrous for that university's reputation, if publicized. Also, if the crackers had altered the data in subtle ways and had not been detected, the human cost could have been considerable. Leakage of confidential data could have led to lawsuits against that university. As it was, it lost a full day of use of those machines while they were secured from the attack, and the cost in wasted staff time of everyone there who could not work effectively must be tallied as well. The bug that was exploited in this case was in a privileged program (*`login`*) which came as part of the operating system of the machines. IBM has made patches (corrected versions of the faulty programs) available, but many sites have not yet applied them.*

[10] The bug in question was reported in CERT Advisory CA-94:09.bin.login.vulnerability on May 23, 1994. The Computer Emergency Response Team (CERT) Advisories are widely circulated among both the good guys and the bad guys.

2.1.2 The use of reusable passwords

The use of reusable passwords which can be cracked or sniffed is rapidly rising to become another one of the top vulnerabilities on computer accounts, and attacks based on this vulnerability are particularly difficult to detect or counter.

> *Shrunk and Blackrat obtained most of the accounts they used by cracking password files.*

While using shadow password files[11] can help prevent password cracking attacks, reports of password sniffing attacks (whereby the attacker "listens on the wire" as the passwords are transmitted unencoded across the network) are becoming very common. Until end-to-end data encryption is widely available, the only solution to these problems is to avoid using reusable passwords, at least on network segments vulnerable to sniffing. This entails carrying lists of one-time ("disposable") passwords, or installing software which can negotiate authentication via a challenge-response mechanism (but this depends on having the software at both ends of the connection, which is not always possible), or using "SmartCard" technologies, which currently cost on the order of $50 per user. Because they are somewhat inconvenient, none of these solutions is likely to gain user acceptance quickly.

2.1.3 Inadequate access controls for remote access

Inadequate access controls for remote access can open up sensitive systems to break-in attempts from offsite, or can make it very hard to trace where activities really originate. In particular, when there are no access controls on dial-up lines, anyone with a telephone and modem can try to access our systems, and if they succeed, it is very difficult to track them down.

> *Shrunk was coming into our systems through the dial-in terminal server Macduf, which requires no authentication to use: anyone can call Macduf and get a connection. It was quite difficult for Michael and Cpl. Bob to find out where he was really coming from; it required not only police involvement, but it required Shrunk to stay on the line long enough for the call to be traced, and it required Michael to monitor his connections day and night.*

2.1.4 Inadequate logging

Inadequate logging of system activities, especially network connections, can make it difficult to trace problematic activities back to the originating account.

[11] Shadow password file: a mechanism for hiding the encrypted password database on a Unix system.

> *Fortunately, Computer Science does log the provenance of incoming logins to their system. Thus, they were able to find out the origin of some of the connections to the compromised accounts on their system, and warn the other sites that there were crackers in their midst.*

2.2 People problems.

Aside from failures of technology, there are attacks which are not readily amenable to a technological solution:

2.2.1 Account sharing

Account sharing is the most mundane problem, and the most widespread.

> *Of course, in the "summer vacation" example, Shrunk and Blackrat mostly shared accounts which they had obtained illegally, but in the case of at least one Alcor account, we believe Blackrat had been given the password by the account owner.*
>
> *In an unrelated incident that took place in early 1995, a legitimate Alcor user gave her password to a companion, with whom she later broke up. She had quite forgotten about the Alcor account, though, and meanwhile the ex-companion had shared the account liberally. When we realized that unusual things were going on, the account was being used by at least five people:*
>
> - *Someone who was using Alcor as a way to get around the firewall*[12] *of a Montreal company, which was allowing connections from Concordia, but not from elsewhere.*
> - *Two people who did lots of newsreading, and subscribed to multiple mailing lists, apparently mostly harmless, but who meanwhile were depriving our legitimate users of hours of modem connect time. One of these people seemed to be coming from the account of a senior administrator at another Canadian university! (This later turned out to be another shared account.)*
> - *One shady character who was very interested in Warez (pirated software), and was using our site to find and download it.*
> - *Another shady character who discussed, in intercepted e-mail, possible break-ins at a large company in Toronto, whose reputation would have suffered had these break-ins succeeded and become known.*

[12] Firewall: a computer security device designed to provide a secure barrier between a protected site and the network at large.

> *It took over a week of staff time to find out what was going on, warn neighbouring sites which might have been compromised, and shut the activities down.*

2.2.2 Authorized users performing unauthorized tasks,

Authorized users performing unauthorized tasks, for example people using our computer systems for entertainment and thereby depriving others of the opportunity to make use of our limited resources for academic, research, or administrative purposes.

> *In fact, this has been a frequent complaint of our student users, as evidenced by the traffic in the netnews group* `concordia.dept.comp-services.help`. *It seems that many people use the terminals and dial-in lines available to them for distinctly non-academic purposes, and do so to such a great extent that it becomes difficult for students with legitimate needs to get their fair share of resources.*

2.2.3 Dishonest users,

Dishonest users, for example clerks who purposely enter incorrect information into databases, or system administrators who look at user files without permission or legitimate need, or otherwise subvert the computer system.

> *"Elduet", a sysadmin at one of the U.S. sites that were involved in the "summer vacation" incidents, turns out to have been crooked. As far as we know he was involved mainly in software piracy and in giving accounts to his co-conspirators. It is possible that he also had access to, for example, credit card numbers from his employer's client database.*

2.2.4 Idle accounts and accounts for unauthorized users.

In general, a user who uses an account actively will report files that appear and disappear, and other similar signs of unauthorized activity.

> *Shrunk and Blackrat were able to operate with near impunity on so many computer accounts because the legitimate account owners were absent or no longer used the accounts.*

2.2.5 Lack of security awareness in the user population.

Many users simply have no idea what to look for, and even with the best of intentions might fail to notice unauthorized activity. Again through ignorance or carelessness, many users transmit their reusable passwords over insecure networks (which may be sniffed), leave files on their accounts with incorrect permissions, use “easy” passwords, and generally pay little attention to the security of their own accounts.

> *One security problem that is often exploited as a prank in the Computer Science labs arises when a user types “*`xhost +`*” to allow the X window system to display a remote application on the local screen. Unfortunately, this leaves the screen wide open to manipulation by an attacker. Pranksters will sometimes pop up an unwanted window on the screen, or worse, they may destroy a useful window. A* really *hostile attacker might do what Blackrat tried to do: intercept the legitimate user’s keystrokes and capture his password.*

2.2.6 Difficulty in determining how serious an incident is.

When symptoms of a possible computer security breach first come to light, it is often not at all clear whether further attention is warranted.

> *When Michael first received the list of cracked passwords from McGill, he could have simply asked the local users to change their passwords, and left matters at that. It was not at all obvious at that point that anything worse was going on than students trying out a new (to them) password cracking program, perhaps just for a lark, or to play pranks on their friends. No one could have foreseen that his extensive investigation would reveal compromised PBXs, credit card fraud, and child pornography.*

2.2.7 Lack of clear procedures for incident response.

There is as yet no widely available, clear, University-wide document outlining procedures to follow in case of suspected computer intrusion, or other computer security related incident. As a result, staff time is spent in creative but *ad hoc* responses to ever more frequent attacks. Also, most incidents are not confined to one site, and an effective set of procedures is needed to guide us in our interaction with other sites to resolve security incidents.

> *For these reasons, and also because many staff members were on vacation, it took longer than necessary for the University to complete its formal complaint to the RCMP in the case of Michael’s “summer vacation”. We were fortunate that, in this case, the delay seems not to have interefered with the investigation.*

2.2.8 Problems with the security and appropriate use policies.

The policies should clearly state the rights and responsibilities of computer users and of those who own and maintain our computers and networks. Further, these rights and responsibilities should be balanced appropriately, so that users maintain a reasonable expectation of privacy, but sysadmins can resolve incidents effectively.

> *A few years ago, McGill University put in place a new computer usage policy, after much uproar and consultation with its users. The result is in most ways admirable. However, the policy puts very strong constraints on what system administrators may do, because of the need to protect user privacy. As a consequence, it was in some cases difficult for the McGill system administrators to cooperate in the "summer vacation" investigation as fully as they would have wished.*

2.3 Consequences of an intrusion.

While most people see reason to deploy resources to protect the root privileges of a system, they are likely to be much more casual about regular user accounts, including their own. However, not only can unauthorized use of a regular account damage the account owner's files, but it provides a good base for cracking a system by finding out more information about it, and trying to exploit bugs that cannot be tripped from the "outside". What can intruders do with a "cracked" account?

- They can download and try to crack[13] the password file.

  > *This is what Shrunk and Blackrat did, and they were rather successful at it. Had they been really lucky, or had a system administrator used an "easy" password, they might even have obtained root privileges that way.*

- They can attack other sites, using ours as a "cover".

  > *One of the "summer vacation" crackers took a stab at a U.S. Navy site. Had he succeeded in penetrating that site, what would Concordia's liability have been with respect to the consequent damages?*

- They can damage the user's files.

[13] Password cracking involves guessing at a password, encrypting the guess, and comparing the encrypted guess with the real encrypted password. While this procedure is computationally expensive, it becomes much more feasible as computers get faster.

> *Our crackers sometimes deleted files in a user account just to make room for the files they wanted to store. What if the damage had not been noticed immediately, and what if the system administrators in Computer Science had not been diligent about doing filesystem backups? Valuable data could have been irrecoverably lost. What if that data had been someone's Ph.D. thesis?*

- They can damage the user's reputation.

> *If our crackers had been malicious instead of expedient in their file modifications, they could have subtly altered data files instead of just deleting them. What if a researcher had published findings based on the corrupted data?*
>
> *Also, what if the crackers had sent out e-mail from the cracked account in a way that made it seem to have been written by the bona fide account holder? What kind of damage to a professor's reputation, for example, could such forgery have caused?*
>
> *In fact, we had an incident in 1996 whereby someone posted a commercial advertisement to netnews from a cracked Alcor account, claiming to be from the real account holder. The account holder received countless "flames"*[14] *and complaints from all over the net, and was very embarassed. Common similar incidents which occur regularly in netnews involve posting materials purporting to be statements of sexual desires or requests for sexual favours from the account holder, who then returns to find a mailbox full of upsetting, embarrassing, and possibly quite offensive material.*

- They can steal information.

> *A researcher working in conjunction with industry, for example, might have proprietary information on her computer account, whose disclosure might significantly affect the competitiveness of the company in question.*

- They can damage system files, or read confidential system files, to which the user account has access.

> *Our "summer vacation" crackers gained access to the account of a senior secretary, who in turn had access to a student information database. Information could have been altered or disclosed.*

[14] Flame: a very strongly worded criticism or insult.

- They can engage in criminal activities.

> *Our crackers were in fact involved in software piracy and child pornography, which is why the RCMP became interested in the case. Their activities abroad caused the U.S. Secret Service to contact Concordia about the case.*
>
> *In a possibly related incident at about the same time, it was discovered that a department here at the University had given root privileges to some students, who had then become involved in Warez distribution. While the activities were shut down immediately, what might the University's liability have been had Microsoft decided to sue?*

It is difficult to place a cost figure on the consequences of a breach in computer security, because most of the time there is no physical damage, and it is not always possible to place an accurate value on stolen information and damaged reputations. On the other hand, it is possible to estimate the value of the computer resources used by intruders, and the value of the staff time required to deal with incidents. In Appendix A, we try to determine the cost of the summer vacation incident.

3 The Top Ten Security Problems.

Every year, SANS[15] creates a "Network Security Roadmap" poster which lists, among other things, the top security problems that plague organizations. The 1996 edition of the poster listed these as the top ten security problems:

1. **Insufficient site resources,** whereby an organization fails to assign sufficient resources to implement the level of security it needs. This is addressed in section 6 below.

2. **Insufficient support or authority,** whereby management fails to assign sufficient authority to staff members responsible for computer security, or fails to support their decisions. This is addressed in section 6 below.

3. **Systems shipped with security problems.** See section 2.1.1 above for an example. This is addressed in section 7.1 below.

4. **Unused vendor patches,** whereby sites fail to install vendor patches for known security vulnerabilities. See section 2.1.1 above for an example. This is addressed in section 7.1 below.

[15] SANS: you can reach the System Administration, Networking & Security conference office at `sans@clark.net`.

5. **Unencrypted reusable passwords,** i.e. passwords transmitted in the clear over the network, which can be sniffed, or passwords whose encrypted form is stored in a publicly readable file, which can be cracked. See section 2.1.2 above for an example. This is addressed in section 7.1 below.

6. **Poor dialup security measures,** whereby insufficient authentication is required before access is granted to the network. See section 2.1.3 above for an example. This is addressed in section 7.2 below.

7. **Open network access,** whereby anyone can get into the organization's network from the Internet. This is addressed in sections 7.1 and 7.2 below.

8. **Inconsistently installed user accounts,** which occur mainly on systems where user accounts are created manually. This is addressed in section 7.3 below.

9. **Poor account monitoring and expiration,** where idle accounts, or accounts belonging to people no longer with the organization, are exploited. See section 2.2.4 above for an example. This is addressed in section 7.3 below.

10. **Poorly configured and audited new systems,** whereby new hosts may be installed on the network without going through any kind of security checks. This is addressed in section 7.1 below.

Unfortunately, the top ten problems have remained fairly constant from year to year, and reflect Concordia's situation fairly accurately in most respects; this is what we hope to change.

4 Solutions Are Needed.

We now know that the linchpins of computer security are: availability, confidentiality, integrity, non-repudiation, access control, and authentication. We understand that the cost of computer security breaches can be difficult to estimate, and that the consequences can be wide-ranging, from lost staff time to damaged reputations, from corrupted information to legal liability. We have seen examples of technical problems and people problems which can increase the risks of a breach in computer security, and we have seen a list of those considered to be the most harmful in practice.

Some of the above problems have technical solutions; for example, programs can be installed to do better connection logging, patches for known system programming errors can be installed, and system configurations can be checked for flaws. Some of the problems will require better policies and procedures to be put into place, and many are amenable to better user education. In some cases, new services could be put into place, such as a centralized patch and security tool repository, and a central computer emergency response team for the University. Some of the problems will always remain difficult to prevent, such as user dishonesty and account sharing, but it may be possible to monitor systems in such a way as to make the detection of such abuses more likely.

Where do we start? Any decisions that are made to allocate resources to computer security (or to not allocate them!) must be made in view of the risks involved, the cost of the associated problems, and the cost of preventing the problems. For this, it is necessary to have an overall view of the importance of various parts of the Concordia Computing Facilities to the mission of the University.

On the other hand, a formal risk analysis is a large endeavour, and might be beyond the resources available to the University. IITS management will decide how to proceed in this respect, but the authors are of the opinion that it would make sense to proceed with such an exercise only if a large project were envisioned, for example converting the entire University to a centralized authentication scheme such as Kerberos, or buying expensive equipment to ensure physical security of our facilities (cooled machine rooms with uninterruptible power, video cameras in all the labs). However, we find that many effective security-enhancing projects are relatively small, and that it is less costly to implement them than to study them. In view of the decentralized nature of computing at Concordia, we have not deemed it productive at this time to consider projects which would require centralizing control of Concordia's computing facilities. We do, however, make proposals whose purpose is to improve the coordination of information flow and the sharing of expertise; with time, an improved awareness of computer security, and an improved climate of communication between computer specialists and University management about computer security issues, might well lead to projects involving some centralization of control (for example for authentication purposes). But it is, in our opinion, too soon for that now.

It is important to understand that computer security cannot be a one-off project — rather, it must be a thread that runs through everything we do. A gradual change in mentalities and procedures is what is required. It is in that spirit that we offer the following section.

Part III
Where Do We Go from Here?

5 Introduction.

5.1 Our Successes and Failures.

Nearly three years have gone by since the incidents described in the first part of this report, and IITS and departments with large computer installations (such as Computer Science and some of the Engineering departments) have not been idle during that time. Many improvements in computer security have been made, and they will be mentioned in italics below, under the appropriate headings.

However, despite those activities, much work remains to be done: not only are there problems still unresolved on equipment managed by IITS and "large" departments (defined for our purposes here as those having large computer installations), but we have not been successful at transferring our expertise to "smaller" departments, whose systems are now bearing the brunt of cracker attacks. This latter lacuna is addressed in sections 6.3 (on centralized services) and 7.1 (on O/S specific issues) below, while the first is the subject of the entire section 7 (on system and network issues).

In addition, the increasing use of the Web by the University to serve sensitive data requires careful management. This is a sufficiently important issue that we've devoted an entire section (9 on world wide web issues) to it.

Finally, we have done a poor job of educating the user community with respect to computer security issues. In view of the fact that some authors have labelled "people problems" (in the form of social engineering, dissatisfied insiders, and simple ignorance) as the single greatest cause of security breaches, and in view of the fact that most of the staff time the authors have spent on incident response in the past few years has been due to causes that originated with a compromised or misused user account, we attach great importance to the solution of this problem, and again have devoted a section (8 on user education) to it.

We recommend that IITS create and manage a Concordia Computer Emergency Response Team (CERT), with structure and mode of operation to be determined by IITS. Despite its name, this CERT would not only coordinate the response to computer security related incidents, but, more importantly, would facilitate the transfer of expertise required to *prevent* those incidents from occurring in the first place. This group would also advise on procedure and policy matters related to computer security. This is the subject of section 6.

5.2 Four Areas Targetted.

Thus, we have identified four major areas which we believe require immediate attention at Concordia:

- Incident response and centralized services (section 6)
- System and network issues (section 7)
- User education issues (section 8)
- World Wide Web issues (section 9)

In the sections below, we identify the specific issues we believe need to be addressed, and propose solutions. Which work would be done by members of the new CERT, which would be done by other IITS staff members or other computer analysts in the course of their regular duties, and which would require the formation of temporary working groups, or be assigned to other existing bodies, must be decided by IITS, though we make some suggestions in that direction.

6 Incident Response and Centralized Services.

6.1 Purpose and Composition.

The purpose of the new CERT will be twofold: first, to assist the University community in implementing proactive measures, which will reduce the risks of computer security incidents; second, to implement procedures for responding to such incidents when they do occur. The CERT will also act as advisor to its (IITS) management in the matter of computer security policies and procedures in general, and will be responsible for coordinating any centralized services related to computer security at the University.

In keeping with the tradition of existing similar teams throughout the world, we propose to name this permanent group the Concordia University Computer Emergency Response Team (Con-CERT). Con-CERT will apply for membership in FIRST, the Forum of Incident Response and Security Teams.

> *IITS participation in the upcoming FIRST*[16] *conference and workshop, to take place in June 1998, is planned. In addition, several IITS staff members have attended network and computer security conferences, and/or participated in security-related working groups, for example within the IETF*[17]*.*

[16] FIRST: Forum of Incident Response and Security Teams, an umbrella group for national and institutional CERTs

[17] IETF: Internet Engineering Task Force, the body which creates technical standards for the Internet.

Con-CERT will operate under the auspices of IITS, which will determine the details of its operational procedures, provide management and computing resources, and appoint a coordinator and additional staff as required. IITS will actively solicit the participation of other University departments.

Con-CERT will be composed primarily of technical experts experienced in the various systems in use at the University. The "core" membership will consist of technical people representing each major operating system in use at the University, computer networking experts, and a representative from IITS management.

The Con-CERT will also have an "extended" membership, which will act as a consultant to the core group, and where individual "extended" members will participate more directly when circumstances require their particular expertise. We expect a pool of consultants similar to the following:

- Sysadmins from various departments
- A representative from the University Security Department
- Someone to deal with Procedures and Policy issues
- Someone to deal with User Education issues
- A public relations specialist
- Legal counsel

6.2 Incident Response.

The Con-CERT will coordinate response to computer security incidents. This will involve acting as a clearinghouse for information concerning incidents in progress, assisting affected parties in notifying the appropriate people (be they within the University or at other sites), issuing alerts when appropriate, notifying vendors or other CERTs when necessary, and, in some cases, contacting the police.
The goals[18] for forming an incident response team are:

- Support centralized, coordinated, and consistent handling of security incidents.
- Provide rapid, effective response to incidents.
- Provide technical guidance on security issues:
  - Periodically re-evaluate risks, policies, procedures.
  - Provide consistent interface to investigative agencies, vendors, media, etc.
  - Develop and/or distribute tools to support security efforts.

[18] These goals are taken from a presentation given by Moira J. West-Brown of the CERT Coordination Center at Carnegie-Mellon University, at Network Security '95, Washington D. C., November 16, 1995.

  - Evaluate incident trends.

- Provide proactive and reactive computer security capabilities.

6.3 Centralized Services.

The details of the following services will be elaborated by the various O/S and network specialists (and some are mentioned in section 7 below), and those people will supply and maintain information as part of their regular duties. Con-CERT will be responsible for coordinating access to the information.

Information services will be provided to the community at large, or to sysadmins and management only, as appropriate. Likely means of transmitting information include anonymous ftp, Web pages, newsgroup announcements, and electronic mailing list announcements.

- Make available the list of departmental security contacts, administrative and technical.
- Maintain Majordomo mailing lists to inform security contacts of new information concerning various O/Ss.
- Maintain a repository of vendor-provided (and other) security-related patches for various O/Ss.
- Maintain a repository of security tools for use by sysadmins, for various O/Ss.
- Assign members to read various newsgroups, mailing lists, and other sources of information, and archive and report on important issues.

Auditing Services will be provided either on request from departments, or, if the Con-CERT judges it advisable, with the permission of IITS.

- Run a central and tamper-proof `tripwire` server, to provide filesystem integrity checking services to Unix machines throughout the University.
- Make available an experienced "tiger team"[19] (or, better yet, a program) which can scan machines for vulnerabilities and report to the sysadmin.

[19] The use of tiger teams to improve computer security awareness should not be discounted. Here's a short anecdote from 1994, about a year before the "summer vacation" events.

One of the authors (Anne Bennett) had just been shown the CERT advisory concerning the AIX `login` bug (subsequently used by Blackrat and Shrunk to gain root privilege at Essex University) by her spouse (a fellow sysadmin at another university), and neither of them could believe the seriousness of the problem. They wanted to try the exploit, but neither of them managed an AIX system. When asked whether she knew of any AIX systems managed by colleagues who wouldn't

Some programs in this vein already exist and can be used, for example the well-known `Satan`.

- Audit the security of the various networks and machines, and assign "security levels" to them – it is important to realize that many machines and subnets on campus cannot be trusted any further than can the Internet at large.

Archiving Services will result in a central repository of information about previous security incidents, and information which might assist in investigating and prosecuting such incidents. This information will be made available to the proper authorities when needed.

- Offer a central logging machine, so that `syslog` messages can be archived in a tamper-proof way (for those operating system, such as Unix, which are capable of such remote logging). Periodically dump the logs to a write-only medium, such as CD/ROM, which can be stored in the vault and retrieved when reliable records are needed by the courts or by other investigatory authorities.
- Keep records of security incidents handled. These most be stored in a safe location, since some of the information may be sensitive. These records

mind a small experiment, Anne thought of Carolyn Beckman, a researcher in the Concordia Biology department, who manages, on a somewhat unofficial basis, several Unix machines in that department. Anne and her spouse then tried the published exploit on Carolyn's system. Here we really must add that the ethics of doing this without asking first were highly questionable, and that Anne would definitely not do this today without first having obtained the appropriate authorization — in fact, people can be fired over such things, and user accounts have been revoked for similar reasons.

In any case, with a one-line command emitted from a machine outside the Concordia network, Anne and her spouse immediately obtained root on the target system! Extremely upset and apologetic, Anne urgently contacted Professor Beckman to explain what had happened.

At the time, the response was "Oh, well, less qualified people are root on that system every day" (fortunately not the crackers in this case!); in all fairness, that response was probably designed to calm Anne down — she hadn't really expected the exploit to succeed and was quite agitated over the whole thing. The system was subsequently secured by the application of a vendor patch, and Prof. Beckman has since become something of a resource person among fellow "unofficial sysadmins". In fact, in June 1998, while rescuing a compromised system in another department as a favour to her colleague there, she wrote:

```
  ``A long time ago you told your boy friend about the old AIX cytox and
he tried to become root on it.  You say it gave you heart failure, but it
was the right move.  I was just beginning and it taught me to pay some
attention to security.''
```

None of Professor Beckman's systems have been compromised to date, whereas there have been several compromises in other departments where Unix systems are managed by researchers and students who are not particularly aware of or interested in computer security.

can be analyzed periodically to obtain reports on incident trends, and on the effectiveness of the measures in place.

6.4 Procedures and Policy Advising.

A new Policy on Computing Facilities has been issued since the events of 1995, which greatly clarifies the rights and responsibilities of users and sysadmins; this has facilitated decision-making during incidents.

As of this writing, the existing *Policy on Computing Facilities* document is under review by John Woodrow (Director of IITS) and Bram Freedman (University Legal Counsel). Security procedures based on that policy, and other existing University policies, will be proposed by Con-CERT. One of the methods that will be used is to analyze past incidents in the light of existing policies.

Anne Bennett has already provided such an analysis, and her proposed policy amendments were discussed at the CCSA meeting of June 11, 1998.

If a situation is encountered which no existing policy can satisfactorily address, then policy changes will be suggested. Also, recommendations may be made concerning additional means of publication of the policies and procedures, for example by making them part of the Calendar, or by including them in any user education materials.

Documents whose production will be considered include:

- Delegation of responsibilities (e.g. to subnets) — responsibilities of subnet managers. Conditions for attaching a host to the Concordia University Network.

- Appropriate Use Details, an interpretation of the Policy on Computing Facilities which gives examples and details on what is and is not permitted behaviour, from a computer security point of view.

- The granting and deletion of accounts, where specific guidelines are set concerning procedures for obtaining an account, and grounds for account revocation, from misbehaviour to idle time to graduation. Grounds for account access suspension, how many warnings will be issued, periods of account suspension for various offenses, repeat offenses, etc.

- Incident response procedure, based on existing workflow in IITS and in the University, including:
  - Determination of incident severity

  - Procedures for escalating locally:
    * From a user to the sysadmin
    * To the Con-CERT
    * To other internal University bodies, such as the Office of Rights and Responbisilities, Security, etc.
    * Issuing alerts within the University
  - Procedures for escalating outside the University:
    * To other sites
    * To vendors
    * To other CERTs
    * To external authorities, such as the police

- Guidelines for internal "prosecution" of offenders. Which policies apply which regulate the computer behaviour of faculty, staff, and students? What are the procedures to file a complaint under those policies, and what kind of evidence is necessary? What sanctions may be applied? Some of this will be based on IITS policies and procedures, and some on more general University policies.

- Guidelines for external prosecution of offenders. Which laws apply? What evidence is admissible, and how can we gather it in ways which (a) don't contravene the law, and (b) will hold up in court? What severity of offense does the University wish to have prosecuted? What severity of offense does the police believe to justify the cost of investigation and prosecution?

7 System and Network Issues.

7.1 O/S Specific Issues.

There are many operating systems, in many versions, in use at the University; the four main families are Unix (SunOS, Solaris, Ultrix, Digital Unix, Linux, HP/UX, AIX, Irix, and Unix-like systems such as Apollo Domain, Mach, and NeXT), VMS (including OpenVMS), PC-based (DOS, Windows 3.x, Windows 95, Windows NT, and OS/2), and Mac-based (the Macintosh operating system).

7.1.1 Technical Issues.

In list in this section many of the technical issues which must be addresses in order to secure a system. The applicability of each item to a particular host will of course vary depending upon the physical location of the host (in a protected machine room, in someone's locked office, in someone's open cubicle, in a public area), and will also vary depending on whether the host in question is a multi-user system, a dedicated server with no interactive users (e.g. Web servers, backup servers, file servers), part of a lab, or a personal desktop system.

- Ensuring proper access control and authentication via *all* access points:
  - Physical security, especially for desktop or lab systems (systems not in a protected room): recommend passwords, screen savers, and boot-time restrictions to force boot-up in a certain state.

    IITS InfoNote M-002 evaluates Mac security products which address this issue.
  - All privileged ("suid") programs. In particular, errors and weaknesses in the operating system design and implementation can cause unintended access to be granted.

    On IITS Unix systems, privileges have been removed from all programs for which they are not strictly necessary.
  - All network services (incoming connections), whether started by the system or by the users, e.g. ftp, e-mail, web or gopher service (including CGI scripts), interactive logins, FAL (PCs), Appleshare (Macs), Timbuktu (Macs), DECnet.

    Remote access controls (via `tcp_wrappers`*) were already in use on IITS and Computer Science Unix systems; their configuration has been checked, and their use has been extended to a greater number of services (sendmail, ssh).*
  - User passwords:
    - Are they guessable?

      Stringent password selection guidelines are now enforced on all IITS Unix systems and on all Computer Science systems; this effectively negates the threat of password guessing.
    - Are they sniffable?

      End-to-end encryption software has been installed on all IITS and Computer Science Unix systems, whose use (where it is possible!) obviates the threat of data and password sniffing. Encryption is now routinely used for most system administration purposes, though its use has not yet spread to the user community. In particular, session encryption client software is not available at most sites outside Concordia.
    - Are they crackable?

Password shadowing has been implemented on Alcor; this effectively removes the threat of password cracking. (Password shadowing has not *been implemented in Computer Science because it is not easily feasible in a heterogeneous network using NIS, but it is on the list of problems to be solved.)*

- Can they be compromised in other ways?

  The risk represented by a user's losing their Alcor "yellow card" (assignment of account and password) has been minimized by forcing the user to select a new password at the time of first login. The new password not only meets the anti-crack, anti-guessing guidelines, but also cannot be identical to the one printed on the yellow card. Computer Science expects to implement a similar scheme soon.

- User accounts:
  - Are they allocated to a legitimate person?

    The threat represented by accounts whose owners are no longer associated with Concordia was already small, but the integration of the SIS (student information system) with the IITS computer account management system has impproved the timeliness of account removal in such cases. In Computer Science, where programmatic access to the SIS is not possible, these accounts are removed in a batch job at the end of each term.

  - Are they in active use?

    The threat represented by idle accounts on Alcor has been diminished by the implementation of an aggressive policy of blocking inactive accounts.

- File and directory permissions, where applicable (for example Appleshare on Macs).
- Dial-ups (modems) and the associated software (on PCs: PCAnywhere, kermit, on Macs: ARA, SLIP, PPP, Kermit).

  Changes were made to the account management Unix client software to permit "terminal server accounts" to be managed; this means that we will soon be in a position to enforce authentication of all dial-in connections in IITS.

- Viruses.

- Ensuring intrusion detection by performing appropriate monitoring, and making sure that logs are reviewed either by humans or automatically. In particular:

  - System activities should be logged appropriately. Is Unix process-level accounting useful and justifiable? What kinds of auditing are available and should be enabled?

    > *A small pilot study was done to show the feasibility of a central Unix log server; the study was successful. Computer Science is in the process of moving to a central log server this year.*

  - User logins and logouts should be recorded reliably. The use of the Venema `logdaemon` package may facilitate this for Unix systems.

    > *User login logging has been greatly improved on Alcor; since November 1997, all user sessions, including not only regulat logins but also POP, IMAP, and X, are logged.*

  - The provenance of network connections to the system (and access to specific services) should be logged. The Venema `tcp_wrappers` package is an invaluable tool on Unix systems.

    > *As mentioned previously, remote access controls (via* `tcp_wrappers`*) are used on IITS and Computer Science Unix systems.*

  - System integrity should be monitored, possibly in the form of filesystem integrity checks. `COPS, tripwire`, and other static integrity checkers should be evaluated, and an appropriate tool made available, where possible.

    > *Several tools have been developed to assist sysadmins in investigating suspicious activity; tools to audit system integrity were already in use on IITS Unix machines before the 1995 incident.*

- Ensuring accountability and nonrepudiation for activities originating from the host. For example:

  - Ensure that our outgoing connections can be identified, for example by running `pidentd` or another port-113 daemon on multi-user systems. If this is not possible, thought should be given to disallowing outgoing connections from those hosts. If dynamic address allocation is used, the address assignments should be logged, or a transition to static addressing should be made.

    > *Identification to remote locations of the local (user) provenance of connections (via* `pidentd`*) was already in use on IITS and Computer Science Unix systems; its use has been extended to a greater number of hosts in various departments.*

  - Perform appropriate logging of certain kinds of network traffic, such as header information from e-mail messages and netnews postings which transit through our systems. The latest versions of sendmail (`sendmail v.8`) perform fairly comprehensive such logging for e-mail. The news server must also be patched to perform similar logging.

7.1.2 Recommendations.

For each major group of O/Ss in the context of multi-user hosts, servers, or labs, the following tasks should be accomplished. The task list is geared mostly to mutli-user and server systems, but similar tasks, where applicable, must be accomplished for desktop machines as well, probably in conjunction with user education (section 8).

- Assign a specialist to "represent" the O/S to the Con-CERT.
- Determine what constitutes an acceptable level of security for hosts, or if the answer depends on the circumstances, determine a scale or security rating system. It is important to realize that many machines and subnets on campus cannot be trusted any further than can the Internet at large.
- In order to minimize the chances of unintended access because of errors and weaknesses in the operating system design and implementation, gather, monitor, and react to:
  - Vendor reports of problems (in some cases available via a vendor's security mailing list), and patches available from vendor.

    > *Many operating system patches have been applied, as they have become available from our vendors. This is a continuing effort.*
  - Other reports of problems, patches, and workarounds, available on third-party mailing lists which report on vulnerabilities (such as BUGTRAQ), on the Web, and via netnews (the `comp.security.*` groups).

    > *In the case of a few very well-known Unix exploits, not only has the vulnerability been closed, but booby traps have been installed on the IITS general-purpose Unix service, Alcor, to detect attempts to gain system privileges using those vulnerabilities — several attempts have already been detected in this way.*
  - Some vendors maintain patch sites; it may be possible to arrange to mirror them for the local community.
- In order to minimize the chances of unintended access because of errors in the system's configuration, prepare a "security checklist" for use by novice system/lab administrators; armed with this checklist, it should be possible for an inexperienced person to ensure a reasonable level of security on a host. For example, AUSCERT provides an excellent such cheklist for Unix systems, which

could be adapted for local use, or possibly even used outright. In the case of single-user systems, checklists aimed at users should be prepared. These checklists will address the issues brought up in the previous section (7.1.1).

- Make available a "tiger team" which can, upon request, audit the level of security of a host.

- Identify and gather useful resources, and submit them to the Con-CERT for distribution to the University community via an ftp archive or Web pages. The maintenance of this archive will be the responsibility of the group's representative to Con-CERT. The COAST archive already contains most freely available tools; where appropriate, vendor-provided tools should also be considered. The group should evaluate the available tools, and recommend a prioritized subset for local use. Sources for this set should be mirrored locally. Where installation conventions can be agreed upon, precompiled versions could even be supplied.

7.2 Network infrastructure issues.

Since IITS already manages the Concordia University network, the appropriate IITS subgroups (Network Engineering Group, Data Communications Group, and Telesis if required) should take on the responsibility of addressing issues related to the campus backbone network, the various subnets, all equipment that routes network traffic (routers, bridges, hubs, switches, FastPaths, modems, etc.), terminal servers (dial-up or hard-wired), and print servers and smart printers. The following tasks should be accomplished:

- Create and maintain the following documentation:

  - Lists of contact people for subnets, and, where appropriate, individual hosts. It should be possible for anyone to find out who to call in an emergency, given only the IP address of the problematic host.

    > *A small pilot study was done to show the feasibility of using the DNS to store per-department and per-host contact information, using the "RP" (responsible person) record type.*

  - Inventory of equipment and O/S, to be used (for example) for sending appropriate CERT and other advisories to the sysadmins concerned.

- Create guidelines for router configuration, with respect to packet and traffic filtering, logging, and monitoring:

  - General security (IP spoofing, connection hijacking)

  - Protecting specific hosts (e.g. those running MIS applications)
  - Protecting specific services (e.g. e-mail, NFS, etc.)

    The threat of e-mail "relay hijacking", whereby badguys at foreign sites use Concordia machines to send spam[20] *to thousands of users here and offsite, was removed in December 1997.*

  - Monitoring physical security (new devices appearing on net?)
  - Detecting misconfigured devices
  - Detecting malfunctioning equipment
  - Monitoring traffic levels to detect denial-of-service attacks or simple traffic load problems before they lead to network availability problems.

- Protect the integrity of the DNS:
  - Implement DNSSec to protect the security of DNS data and prevent DNS-based attacks.
  - Implement automated DNS checking programs to ensure the consistency of DNS data.

    Programs were installed in April 1998 to monitor DNS data consistency, and report problems.

- Create a formal procedure to hand off responsibility in the case of subnets not managed by IITS:
  - Provide contacts
  - Have department formally accept responsibility
  - Rate the subnet as more or less secure
  - Compliance checklist to assist novice netadmins

- Evaluate the physical security of each network segment, and recommend the use of end-to-end encryption where privacy cannot otherwise be assured.

7.3 User Account Management.

As mentioned in section 3 on the top ten security problems, inconsistently installed user accounts and poor account monitoring and expiration are two very common security vulnerabilities. In fact, while the Concordia Policy on Computing Facilities[21]

[20] Spam: unwanted bulk e-mail; also used for inappropriate multiply posted netnews articles.

[21] The Concordia Policy on Computing Facilities can be found on the web at

is very clear as to exactly who is allowed access to our computers, all too often sloppy tracking of accounts allows the policy to be violated, and creates security vulnerabilities.

The IITS Department is very fortunate in having a locally written computer account management facility, called "AGEM", which interfaces with the SIS (student information system) and human resources databases to determine who is entitled to have a computer account, and which is able to track personnel and student departures from the University and issue account blocking directives in consequence.

> *As mentioned previously, the threat represented by accounts whose owners are no longer associated with Concordia was already small, but the integration of the SIS (student information system) with the IITS computer account management system has improved the timeliness of account removal in such cases. In Computer Science, where programmatic access to the SIS is not possible, these accounts are removed in a batch job at the end of each term.*

Since all user accounts (on those systems managed by AGEM) are created by a program, the accounts are created in a consistent manner not subject to human error. Of course, having such a system at all makes it possible for IITS to manage the thousands of computer accounts which are created and deleted every year. In addition, the presence of AGEM makes it possible to monitor and block idle accounts in an automated way.

> *As mentioned previously, the threat represented by idle accounts on Alcor has been diminished by the implementation of an aggressive policy of blocking inactive accounts.*

One criticism of AGEM is that directives are neither encrypted nor digitally signed, which provides a point of attack to network sniffers. Fortunately, encryption and digital signatures can be added quite easily with PGP, since directives are transmitted via ordinary e-mail.

The use of AGEM contributes to computer security in IITS, and the maintenance and continued use of this program is essential. Unfortunately, support of AGEM has not always received a high priority within the IITS Department.

> *As mentioned previously, changes were made to the account management Unix client software to permit "terminal server accounts" to be managed;*

- `http://compserv.concordia.ca/Computing_Services/geninfo/Policy/policy.html` (html).

> *this means that we will soon be in a position to enforce authentication of all dial-in connections. However, support for this at the level of the AGEM server has not been forthcoming.*

In view of the enormous usefulness of AGEM, we must recommend that its use be expanded to other systems (in particular to the dial-ins) and other departments where possible, and that IITS recognize its importance and prioritize its support accordingly. Also, we recommend the use of PGP to encrypt and sign directives issued by the server.

8 User Education Issues.

Better user education is crucial to improving computer security at Concordia; in addition, it is highly desirable to make users better aware of the many computer resources available to them, and to teach them to use those resources correctly and effectively. IITS already offers some training courses and a fair amount of documentation, both online and in paper form; few of those, however, emphasize computer security issues.

We recommend setting up a group to review existing user education procedures and to recommend changes (if needed) to improve the level of computer security awareness in the Concordia user population.

The group would obtain from the O/S, network, and WWW specialists, prioritized lists of information which, it is felt, users should know, concerning computer security. The group would determine which constituencies within the University community must be addressed, and, after consulting with any existing user education groups, would recommend methods to best reach these people. If possible, methods would be recommended which cannot be repudiated, for example tests or signatures.

The group would begin by making an inventory of existing resources, and of met and unmet needs. If it were deemed likely to be helpful, the user community might be polled directly. The group would, of course, recommend the best use of resources which already exist within the University where possible, and, with the help of the O/S specialists, would help find and adapt resources in use elsewhere. Where necessary, the development of new resources (documentation, courses, computer programs) would be recommended. All methods would be considered: courses and seminars, short presentations (for example at Orientation), paper documentation (for example InfoNotes and User Guides), online documentation (such as Web pages), computer programs for training users, tests (both paper and online), videos, games, demonstrations...

Examples of topics which would be expected to be covered are:

- Account sharing is forbidden.

- Selecting good passwords.
- Protecting one's password.
- Avoiding being victimized by "social engineering", for example in IRC.
- Protecting sensitive e-mail, for example with PGP.
- Avoiding running unknown code, whether obtained explicitly, or received via e-mail (application macros) or the web (Java).
- Monitoring one's accounts for signs of tampering.

  > *Part of the Alcor web site has been devoted to user account security issues, and includes a checklist of ways to detect whether one's account is being used by someone else.*
  >
  > *Also, a new program,* `filecheck`*, has been implemented on Alcor, which assists users in looking for suspicious files on their accounts.*

- The protection of data on one's desktop system:
  - Physical protection of the system.
  - Using screen savers and passwords to prevent access to existing sessions.
  - Using "Boot lock" programs to prevent unauthorized reboots, or unauthorized access after a reboot.
  - Avoiding introducing viruses, and scanning for viruses.
- The safe use of the Web (see section 9).

9 World Wide Web issues.

9.1 Technical Issues.

The World Wide Web has been identified as an institutionally important application. However, there are many issues to be addressed with respect to this popular application:

Attacks on users. Users are often completely unaware of the risks they take in using the World Wide Web.

- Browsers: What risks do the various "client" software packages pose to user accounts? Can these risks be reduced by the system administrator, or can the user easily override any safety devices installed? What kinds of user education are needed, and how far can we count on their effectiveness? All browsers in use at the University should be examined for security, including Lynx, Mosaic, Netscape, Explorer, url_get, and Java byte code and its purported security.
  - Protecting user account privacy and integrity: are users aware of the dangers of particular browsers, and are they taking appropriate measures to protect their data?
  - User accounts on sensitive systems: in view of the dangers of some browsers, should their use be allowed at all on "sensitive" systems?
- Data sniffing: Are users aware that data on the networks can be sniffed? Do they take appropriate measures if they wish to transmit, for example, passwords or credit card numbers?

  > *"Secure" (encrypting) web server software has been installed on several machines in IITS.*

- Web home pages: Are users aware that the information they place on their "home pages" can be accessed by anyone, anywhere in the world? If they use CGI scripts or code, are they aware of the security implications?

  > *Web/CGI vulnerabilities have been greatly reduced on Alcor, while still permitting our users to install and run their own CGI scripts. This has been done by both running the web server in a restricted filesystem (to avoid attacks on the system and on the users' non-web data), and by ensuring that user's CGI scripts run with their privileges, not the web server's. These measures have been in place since September 1996.*

Attacks on the system via the Web server software. Server software must often run as a privileged user, either to bind a "privileged port" (Unix), or to access protected data (MIS Applications). Can the system be protected adequately against attacks on the server? Are accesses logged and are logs reviewed in such a way as to permit the detection and tracing of attacks?

- Exploitation of server security holes.
- Exploitation of errors in application programs (for example, programs which pass user-supplied data to privileged programs), for example erroneous CGI scripts and code.

Attacks on sensitive data and applications. The issues above recur, with the addition of the access control considerations outlined below.

Denial of service on production applications. This popular service is almost an invitation to denial-of-service attacks, especially if it is used for applications such as student registration and payment, where some student may have an "interest" in disabling the system, at least temporarily. What can be done about such attacks?

Subverting access control. When sensitive data is served via the Web, reliable methods must be used to ensure that only authorized users have access to the data.

- Password guessing: What is our vulnerability to brute force attacks to obtain password-protected data?
- Password sniffing and data sniffing: What are the implications of sending sensitive data (authentication information, confidential University data, private student or employee information, possibly eventually payments) over insecure networks? How vulnerable are we to sniffing attacks? Should we refuse to transmit unencrypted data over networks of unknown security?
- Breaking the system or the Web server: As mentioned above, server security holes or errors in application programs may be exploited to gain unauthorized access to information.
- DNS and IP spoofing: All TCP/IP connections are subject to IP spoofing (source routing) and connection hijacking (IP sequence number guessing) attacks. DNS subversion is also common. How to we deal with these problems?

Some of the above issues are clarified in *The World Wide Web Security FAQ*,[22].

9.2 Recommendations.

Web working groups already exist within the University, and those groups should be persuaded to acquire the expertise needed to handle security issues (possibly by adding members familiar with those issues). The groups should produce:

[22] *The World Wide Web Security FAQ*, Lincoln D. Stein (`lstein@cshl.org`), Version 1.8.1, April 16, 1998. Copies can be obtained at

- `http://www-genome.wi.mit.edu/WWW/faqs/www-security-faq.txt` (text only) and
- `http://www.w3.org/Security/Faq/` (html).

- Checklists for Webmasters, to assist in securing server software against attack, and to assist in securing the system against weaknesses in the server software. For example, on Unix systems, `chroot` may be used to limit the damage to the system should the server software be broken. Where such checklists already exist, they should be referenced.

- A checklist for designers and programmers of Web applications, to prevent the exploitation of these programs to obtain access to confidential data, or to otherwise attack system security. For example, all data not generated by the program itself must be treated with caution, and parsed for dangerous "metacharacters", buffer overruns, etc. Where such resources already exist, they should be referenced.

- A user education package (in conjunction with any user education working group) which will instruct users in the safe use of their client software, including warnings against transmitting sensitive data (such as credit card information) in an insecure manner.

- A recommendation to IITS management concerning the safe implementation of MIS-type Web applications, based on a risk analysis.

10 Conclusion.

In this section, we identified four major areas which we believe require immediate attention at Concordia, and we made recommendations for improvements in each area, which we now summarize:

- Incident response and centralized services (section 6): we recommend setting up a Concordia CERT, which will:

  - Assist with incident response.
  - Provide centralized resources to assist sysadmins in maintaining security on their systems.
  - Advise IITS on policy and procedures where they relate to computer security.

- System and network issues (section 7): we listed many technical issues which must be addressed:

  - It is most urgent to provide sysadmins at Concordia with local standards, in the form of a "securing the system checklist", and in the form of pre-evaluated tools and patches. Many of these tasks should be done in the context of the above CERT.
  - It is also essential to develop contact lists which will provide rapid ways to notify the appropriate people in case of incidents or new vulnerabilities.
  - We must improve our use of router-level packet filtering to protect our services from attack.
  - Network security requires subnet administrators to consciously take responsibility for their subnets; the development of a formal subnet hand-off procedure would be helpful in this regard.
  - IITS should recommit itself to the support and expansion of its excellent AGEM account management package, and add encryption and digital signatures (via PGP).

- User education issues (section 8) were discussed without much detail in this report, but their importance should not be minimized. On the contrary, computer security issues should be integrated into existing user education courses and user documentation, and should be publicized more aggressively. We recommend that a temporary working group be set up to identify the issues in detail and to ensure that they are addressed.

- World Wide Web issues (section 9) should be dealt with by existing WWW working groups. The most important issues are:
  - Elaboration and enforcement of guidelines on the safe installation of web server software.
  - Elaboration and enforcement of guidelines on the safe implementation of MIS and SIS applications, which require authenticated access to sensitive data.
  - Selection and configuration of browser software to avoid user data compromise.

While the amount of work to be done is large, much of it falls under the existing responsibilities of the people who manage the University's computing facilities. Our hope in this respect has been to cut the task down to size by organizing it into manageable components. We also hope that the establishment of a Con-CERT as a clearinghouse for security-related information will reduce the amount of duplication that is currently occurring among those already addressing computer security issues, and that it will make addressing those issues feasible for those who have thus far not known where to turn for information or assistance.

However, no security measures can be effective unless they are supported by the user community. Therefore, the authors' fondest hope in issuing this report is that we will have succeeded in sensitizing the University community at large to the importance of good computer security practices in the everyday work of each and every one of us. Happy — and safe! — computing to all, and to all a good night.

A (Appendix) Cost Analysis of a Summer Vacation

The "Real" Cost, Whatever That Might Be

We should say at the outset that there is no way of assigning a "real" dollar value to the damage caused by our intruders, because the harm caused is mostly intangible. They did delete some files, but none that couldn't be recovered easily from backup tapes. They did use some disk space, but none that couldn't be recovered easily by deleting their files. To be grossly materialistic about it, we ought to say that the only damages caused to Concordia consisted of demonstrable stolen services — 312 hours' dialup connection time in the case of Shrunk, and 86 hours' physical use of terminals in the case of Blackrat; the estimated 100 hours' use of terminal time in H511 were not logged, and therefore not "demonstrable." At any rate, that is how a Court might see it if the Court were convinced that "unauthorized use of a computer" is essentially just a form of theft, as its listing under *Offenses Resembling Theft* unfortunately suggests. To assess the monetary value of the damages, the Court might reasonably look at the market value of connection time as charged by an Internet service provider — typically about $1 per hour. Our damages would thus have added up to about $400; not much, in view of all the fuss.

As people concerned on a daily basis with computer security, however, we must reject this view as based on a fundamental misconception of the rôle of computers — especially internetworked computers such as the ones involved in our story — in the modern workplace. Authorized computer users depend on the integrity of the programs and data they use, and on the security of the channels of communication with other authorized users, whether on the same machine or on another continent. In our particular case, professors must be confident that the exams they compose are not readable (or writable!) by arbitrary intruders, that their research work is accessible only to legitimate members of their research group, and that their mail is delivered quickly and reliably to its intended recipients. Students must be confident that they will have access to computing equipment when they need it, that their work will not be corrupted, erased, or stolen, and that they will be able to use the Internet as both a source of information and a forum in which to exchange ideas. The university's administration depends on its computer systems for the full range of its activities, *e.g.,* student record keeping, payroll, and MIS, and the law stipulates that much of this information must be kept confidential.

Our summer vacation intruders did not significantly damage any of these functions directly, nor was that the immediate purpose of their intrusion. In fact, they were not especially interested in Concordia at all. It was quite evident from the pattern of their activities that they were members of an informal community of “hackers” (crackers) whose objective was to gain access to as many machines as possible, with as much privilege as possible, always with the complete anonymity that comes with a “hijacked” account. From these accounts they could launch attacks against other sites on the Internet without fear of adverse consequences; and this is exactly what they did. From Concordia, they attacked McGill, “Essex” and a number of other sites. From their stolen accounts, they joined several IRC channels on which they participated actively in various forms of illegal commerce. In fact, Shrunk got access to Concordia accounts from Blackrat in exchange for access to Shrunk’s pirated software repository at “Sussex.”

The *real* harm done to Concordia was not the misappropriation of some otherwise idle CPU time during the least busy period of the year. It was the conversion of Concordia from a responsible, law-abiding site on the Internet into a source of anonymous and untraceable attacks against any number of other connected sites, and into an unwilling accessory to miscellaneous breaches of copyright, fraud, and obscenity laws. While the intruders were present (and undetected), Concordia was a danger to its neighbors. We had lost effective control of our site, and while that was going on, our reputation as a good network citizen was in danger. From this perspective, it makes little sense to assimilate the intrusion to a common theft. If we must use the model of theft, it would be more appropriate to consider it on a par with theft of firearms; the monetary value of the object stolen is trivial in comparison to the danger posed by the thief in possession of the loot. Perhaps it would be better to assimilate this sort of crime to trespassing. But these trespassers have not just occupied our property; they’ve set it up as a common bawdy house in full view of our neighbours and the police. Again, not much damage to the property has been done, but that’s obviously not the real issue.

The Practical Cost

After due deference to the fact that we didn’t suffer great losses directly attributable to the intruders in a narrow legal sense, we come now to the practical cost of the affair — the cost that any manager might reasonably assign to it.

Apart from the approximately $400 of direct costs related to connection time and resource use, the most obvious cost is the investment of employee time: roughly two person-months of “Grade 11” analyst time ($6000) at Concordia, not to mention similar costs at McGill and the RCMP. From a legal perspective, this might be seen as a routine “cost of doing business,” but this presumes that we have computer security staff sitting on their hands waiting for security breaches. In the real world, we don’t have the staff, so the expense is directly related to this particular breach. “Essex” lost the use of its computers for a day, which implies significant cost in wasted staff time.

Others at Concordia were involved as well: Telesis, Legal Counsel, the Security Department, as well as several analysts and managers in Computer Science and IITS (then Computing Services). It would not be unreasonable to suppose that their combined efforts might have cost $5000.

We are not even counting the cost of regular system security procedures, whose implementation is made necessary by precisely the Shrunk and Blackrat type of criminal.

We won't pretend that the figures are exact, but it is at least reasonable to estimate that Concordia spent more than $10,000 on this incident alone.