1. Introduction

Computers are used maliciously in a number of different ways, ranging from annoying acts like spam through denial of service attacks, computer crime, identity theft, child solicitation, and so on. Every computing technology that has been developed to help us has been followed by people finding ways to use the technology to harm us.

This is, unfortunately, a huge area, and so these notes can never be comprehensive.

The sections below are in no reasoned order, but are just based on when I decided to write about them. They are not in order of importance to the community.

2. Spam

Whereas we used to use phone calls and sometimes letters, and then later faxes for most of our personal communications at distance, email now dominates how we do this. We have come to rely on this technology, and spammers have taken advantage of our need to use it.

Spam is the junk mail of the Internet. Whereas everything else that I describe in these notes has the potential to have significant negative effects on the affected individuals, most of the time spamming is just an annoyance, although it does take time to recognize and delete messages, uses up bandwidth, makes us use filters which slow down the processing of email, and clogs disk space, all of which have associated costs.
Long before there were enough people with emails to make spam practical, junk mail was a fact of life in our (non-electronic) mailboxes. The economics of regular junk mail were, however, very different from the economics of spam email. Bulk mail, even though it is only a bit more than half of the cost of first class mail, is still expensive on a per piece cost, not only because of the mailing costs but also because the mailing lists are expensive and the printing costs are large. So if it cost, say, $1 per package mailed and the potential profit for each hit (when someone bought what you were advertising) was $10, then you needed to have a hit rate of 10% to just break even, and that sort of hit rate was rarely achieved. The economics change completely with spam. The cost for spamming 1 million email accounts is usually claimed to be about $1000, so to break even with the same $10 profit figure you now need to only have a 0.01% hit rate. The cost per package sent has dropped from $1 to one tenth of a cent, a drop of 1000 times.

To protect against spam most large ISPs have spam blockers, and so, for example, AOL blocks over one million emails a day that they have determined are spam. In the College of Engineering we mainly rely on SpamAssassin, an Apache open source product. SpamAssassin has about 600 rules that it applies to each email, with a score for any violation of a rule. It adds up these penalty scores and this total becomes the penalty assigned to the email. The user can then decide how they want to proceed. E.g., one option is to delete any email with a score over 10 and send the email, with a warning, if the score is between 5 and 10. The current list of SpamAssassin tests is at http://spamassassin.apache.org/tests_3_0_x.html. They cover a wide range of different approaches. E.g., if the email comes from a new high volume site then this gives it a penalty of 1.0, it dings severely if the message has HTML which is shouting, it looks for words like Viagra, and really doesn’t like it if an attempt has been made to disguise the word by including spaces and matching backspaces, promises of body enhancements are dinged, emails from blacklisted sites have no chance, it has a number of Bayesian statistical filters, and so on.

The danger of relying too much on a spam filter is, of course, that it might get false positives, which means that legitimate emails (often called ham for obvious reasons) get deleted. For example, if your mother sends an email which mentions a friend at a bank who could help you to get a mortgage, includes a Viagra joke, and uses a new free email system which is very successful and is suddenly getting a lot of traffic then there is a chance that it will be deleted.

There are a number of different approaches to how to handle mail that a filter believes is spam. Most fit into one of the following categories:

1. Let all mail through to the user.
2. Use a spam filter that lets everything through, but marks potential spam.
3. Use a spam filter which deletes obvious spam, and lets through the rest, marking potential spam.
4. Use a spam filter which deletes obvious spam and lets through the rest.
5. Introduce a cost or other interference system to all Internet mail.
6. Use a challenge-response email system.
Option 1 has become impractical because of the volume of spam on the Internet. Most of us don’t have enough patience, or time, to deal with it, and so we want filtered email.

Option 2 is preferred by some, because they can’t stand the thought of getting some ham filtered out without seeing it. Typically color coding is used to designate different levels of spam confidence, and unread mail is sorted by color, so scanning and deleting potential spam is fairly quick.

Option 3 is probably better than option 2, because you can always set the spam threshold high if you are concerned.

Option 4 is common, but has no obvious advantages over option 3.

Option 5 has led to a lot of discussion over the last couple of years, particularly since Bill Gates got into the debate with his statement at the 2004 World Economic Forum\(^1\) that “two years from now spam will be solved.”\(^2\) One suggestion is that every time you send an email you are charged a cost, say 10¢, which is collected and goes into, say, a charity that you select. Another suggestion is that emailers must have a built in time delay between messages that they send out. So, for example, if you spam out a million emails and there is a 30 second delay between each one, then it would take a year to get out the spam. Another proposal is that when you send out an email you have to answer a very simple question before it is sent, for each recipient. Since an essay in 1998\(^3\) Gates has been pushing the concept that all unsolicited emails come with a cost that the recipient chooses. If the recipient reads the mail the sender has to pay the cost.

Option 6, challenge/response or C/R, has not really caught on much, although it is possibly the most effective in terms of spam elimination (some people say that this cure is worse than the disease). The idea is that when you receive an email from someone who isn’t on your whitelist of acceptable addresses your emailer automatically sends a response asking for a confirmation. This response will typically blow up when it tries to go back to a spam site. If the original sender responds with a confirmation then the email comes to you and they go onto your whitelist. The most common challenge/response email system is currently TMDA (Tagged Message Delivery Agent), which is an open source system that can be obtained from [http://tmda.net/](http://tmda.net/). Brad Templeton\(^4\) has a good page on challenge/response netiquette at [http://www.templetons.com/brad/spam/challengeresponse.html](http://www.templetons.com/brad/spam/challengeresponse.html) and has a page on the advantages and disadvantages of these systems at [http://www.templetons.com/brad/spam/crgood.html](http://www.templetons.com/brad/spam/crgood.html), including discussions of two of the major C/R problems, responding to a faked return address on spam mail (which can bury some innocent user), and how to handle mailing lists. C/R systems which handle mailing

---

2. Obviously this wasn’t one of his better predictions.
4. He’s an interesting guy, with a lot of accomplishments, including being the founder of ClariNet, the Chairman of the Board of the Electronic Frontier Foundation ([www.eff.org](http://www.eff.org)) and the founder and long term moderator of rec.humor.funny, which took the place of alt.sex as the most commonly read newsgroup, and is now on the Web at [http://www.netfunny.com/rhf/current.html](http://www.netfunny.com/rhf/current.html).
lists poorly can be annoying to list managers (e.g., class instructors who use the MSU class mailing system for announcements).

Spammers tend to find their way around the best attempts of the filterers. E.g. a recent trend has been to send a file of reasonable text followed by a JPEG (or other) image that covers up the text and contains a photo of the advertisement. This blows by most filters because they can’t recognize the text in the image.

So what is the long term solution? While I don’t, unfortunately, agree with Bill Gates that the problem will be solved by 2006, and don’t think that his “the user will select the receipt charge” approach will become part of email, he is probably correct that since the problem is based on economic decisions the ultimate solution, if it ever occurs, will be based on making spam too expensive for the spammers.

One of the methods that I mentioned briefly above for doing filtering is a Bayesian statistical filter. Many claim that this is the future of spam filtering, and so I’ll briefly describe how they work here. For more details do a search on Bayes, email, and filter, and you’ll get lots of choices. The Reverend Bayes was an 18th Century Englishman who studied statistics and probability when he was not doing his church duties. His most important result was published by the Royal Society in 1711, three years after his death. The result, which is still one of the best known theorems in probability, is:

\[
P(A \mid B) = \frac{P(A) \cdot P(B \mid A)}{P(B)}
\]

which means that the probability that A will occur given that B has occurred is equal to the probability that A occurs times the probability that B will occur given that A has occurred divided by the probability that B will occur. This sounds a mess, so let’s try an example. We’ll throw two standard fair dice. The events will be:

A: The first die is a 4
B: The total of the two dice is 9

Looking at all 36 possibilities for the two dice ((1, 1), (1, 2), …, (6, 6)) it is easy to see that \( P(A) = \frac{1}{6} \) and \( P(B) = \frac{1}{6} \). If the first die is a 4 then for the total to be 9 the second must be a 5, so \( P(B \mid A) = \frac{1}{6} \). If the total is 9, then the first die can be any of 3, 4, 5, and 6, with equal probability, and so the probability that it is a 4 is \( \frac{1}{4} \), so \( P(A \mid B) = \frac{1}{4} \). The Bayesian equation then becomes \( \frac{1}{4} = \frac{ \frac{1}{6} \cdot \frac{1}{6} }{ \frac{1}{6} } \), which is correct.

Now that we’ve got that discrete math out of the way, what does that tell us about filtering email. Using this for a Bayesian filter or classifier, we take two large sets of email, one of which is known to only contain spam and one of which is known to only contain ham, and use these sets to train the classifier. It can look at all of the words, or all of the phrases, in the two sets, and based on whether the emails containing them are ham or spam, assign a spam probability to each word or phrase. Then when a new email
comes in we use the dictionary of probabilities to assign a spam probability to the email. If this is above some threshold then it is determined to be spam. One good thing about Bayesian filters is that their training can continue as they are used. If some spam gets through you can submit it to the classifier to add further training, which will slightly change the probability dictionary, and if the worst happens, a false positive, you can also send this to the classifier for further training. Bayesian filters work fairly well, although spammers try to confuse them by adding large numbers of hidden random words.

Another approach to combating spam is through legislation. In the US this has been attempted in many states, and also at the federal level through the unfortunately named CAN SPAM (Controlling the Assault of Non-Solicited Pornography And Marketing) act of 2003. The acronym was supposed to be interpreted as canning or throwing away spam, but most cynics call it the YOU CAN SPAM legislation. The major complaint against the act has been that it has superseded a number of state laws, which in many cases were more restrictive. For example, California had an anti-spam law that was in effect until December 31, 2003, which was to be replaced by a new and much stronger law in January 1, 2004, which was effectively blocked by the CAN SPAM act, which also came into effect on that day.

Most state laws included a requirement that all mails whose primary purpose was advertising had to include ADV in the subject line, to make things easier for spam blockers. The CAN SPAM act was supported by the Direct Marketing Association (DMA), the Association of National Advertisers (ANA), and the American Association of Advertising Agencies (AAAA), which are the major US organizations related to direct marketing.

If an email comes from a business which you already have some kind of relationship with then CAN SPAM requires that the return address and computer can’t be spoofed or hidden. Commercial emails to recipients who have agreed to receive them (or didn’t ask to be removed from a list when they had the chance) are also required to include an opt-out mechanism (and also the snail mail address), although the online opt-out can be used by people outside the law (e.g., emailing from outside the US) to confirm that this is an active e-address. Unsolicited commercial emails must also have clear and conspicuous text that says that this is an advertisement, and if it is a sexually explicit ad then the message must be in the subject line.

The CAN SPAM Act was co-sponsored by Senators Conrad Burns (Republican, Montana) and Ron Wyden (Democrat, Wyoming). It was the only successful bill out of a number of competing related bills in the US House and Senate in 2003 and 2004, including an Anti-Phishing Act, a number of other anti-spam acts, and the Computer Owner’s Bill of Rights, which would have established a “Do Not Email” registry for unsolicited commercial emails similar to the “Do Not Call” registry for telephone

---

5 The CAN SPAM legislation includes the clause “This Act supersedes any statute, regulation, or rule of a State or political subdivision of a State that expressly regulates the use of electronic mail to send commercial messages, except to the extent that any such statute, regulation, or rule prohibits falsity or deception in any portion of a commercial electronic mail message or information attached thereto.
owners. The text of the CAN SPAM Act, as well as descriptions of the failed federal legislative efforts, is at http://www.spamlaws.com/federal/summ108.shtml#s877.

3. Denial of Service (DoS) Attacks

DoS attacks are usually malicious, with no intent for profit, although there have been some blackmail attempts associated with DoS attacks. In 2001 researchers at the University of California in San Diego estimated that there were about 4000 DoS attacks each week, worldwide, with half of them lasting less than 10 minutes. So in some sense DoS attacks are a relatively small problem, as compared, say, to the number of spam emails that go out every week, but the targets of DoS attacks tend to be the more visible, and in some cases more important, sites on the Internet, and so the effects can be significant. Also it is generally believed that the number of attacks have increased steadily since that report in 2001.

By its very nature, a DoS attack will not come from a single machine, or it will also become overloaded. So the usual approach is to write a worm which attempts to distribute itself widely and quietly around the Internet. It does nothing malicious (and so tries to stay under the virus protection radar) until a specific time when all of the copies of the worm attempt to send as many messages (e.g., pings or Web hits) as possible to the target site. Attacks like this are also often called DDoS (Distributed DoS) attacks.

The most popular target for DOS attacks has been www.microsoft.com, but other visible sites like www.cnn.com are also commonly attacked. In October 2002 an attempt was made to take down the Internet with a DoS attack on the root name servers (see the description of these servers in the pharming section of these notes, below). At the time there were 13 root name servers, which were attacked with ping flooding from a large number of machines. Normally ping sends one 64-bit request per second, but here ping had been modified to send the packages as fast as possible. A root server run by the Internet Software Consortium reported 80 Mbps traffic into its machine, an increase of ten times over normal, but it stayed up. Overall four of the 13 systems kept performing with some degradation, two were severely affected, and seven were described as completely crippled. The attack lasted for an hour, but the system as a whole handled it without most users seeing any degradation. As reported on www.internetnews.com, “A spokesman for UUNET, which is the service provider for two of the root servers, told internetnews.com it was the "largest, most targeted attack" ever seen. ‘This did not affect the end user but it was huge and concerted. It was rare because it was aimed at all 13 servers. It was an attack on the Internet itself and not a particular Web site or service provider,’ he explained.” Post mortems on the incident said that the redundancy on the root name servers was only just sufficient to stave off the attack, and that if it had lasted much longer than an hour the remaining six systems would have been taken out by the cascading effect as the others dropped out.

The usual approach is that the DoS attack messages will forge, and keep changing, their source address, so that the machine under attack cannot just block all incoming traffic
from the attacking machine. The Network Working Group recommends best practices for thwarting this kind of attack in one of their RFCs (Request for Comments) named RFC 2827 - Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing at www.faqs.org/rfcs/rfc2827.html.

One of the most famous DoS attacks was part of the Blaster worm, which spread rapidly in August 2003. Part of the goal of this worm was to, beginning most months on the 16th of the month, initiate DoS attacks on the Windows update computer at Microsoft to make it hard for people to get the updates needed to block the worm once they became infected. Microsoft had enough warning before August 16th, and had also had to increase system capabilities before the 16th to handle all of the people who needed updates to protect against the worm, and so their update capabilities were not severely affected when the DoS attacks began that day.

4. Child Solicitation

This has been a particular problem with chat rooms and social networking sites, although email and instant message systems are also used heavily. There have been a depressingly large number of cases where pedophiles have contacted children through, say, a chat room devoted to activities ranging from school sports through Barbie dolls. Typically they have pretended to be slightly older than their targets. They have formed relationships, usually moved from chat rooms to email, and have sometimes arranged meetings. The size of the problem is staggering. Researchers from the National Center for Missing and Exploited Children interviewed a representative sample of 1501 children between the ages of 10 and 17 and claim that approximately 20% of these children were solicited online every year6.

Most statistics that are available precede the spectacular growth in social network sites (e.g. MySpace and Facebook) over the last couple of years, but folklore predicts that they will have made the problem much worse. People are often very free with pictures, videos, and personal information on these sites, which makes life much easier for trolling predators. The numbers are huge; MySpace, for example, had about 65 million visitors in April 20077, and so law enforcement efforts to penetrate these predatory trolling efforts tend to get overwhelmed.

To combat these predators law enforcement agencies have taken to going out onto chat rooms pretending to be young girls or boys and this has led to many arrests and convictions, despite some unsuccessful claims of entrapment. The really depressing part of this has been how successful these trolling efforts have been, because it implies that many children are being sought and solicited through these systems. In 2000 the British Home Office also estimated that 20% of children using chat rooms have been solicited by

---

6 http://www.missingkids.com/adcouncil/
7 http://blog.compete.com/2007/05/14/top-social-networks-april-facebook/myspace
pedophiles. For example in February 2007 a Las Vegas couple contacted a 15-year-old Florida girl through a chatroom, and began soliciting her to be the husband’s sex slave. They sent her a plane ticket to Las Vegas where she was supposed to have a sexual encounter with the husband and then meet the rest of the family, telling her that “the idea here is for you to escape without a trace.” Fortunately the “girl” was a male Florida detective on their child pornography and exploitation task force, so the husband was arrested at Las Vegas airport. On September 5, 2007, the husband and wife were sentenced to 15 and 10 years, respectively.

As another example of this kind of enforcement sting, on September 17th 2007 an Assistant US Attorney from Florida was arrested at Detroit Metropolitan Airport after he thought that he had made an arrangement online with a mother to have sex with her 5-year-old daughter after ensuring the detective pretending to be the mother online that “I’ve done it plenty” and that there would be no harm to “her” daughter. He was arrested with gifts for the “girl.” On October 5, 2007, he committed suicide in his cell while awaiting trial on charges that included crossing state lines with intent to have sex with someone under 12, which carries a minimum penalty of 30 years.

Florida has been very proactive on attempts to protect children. They estimate that 75 million children use the Internet each day, and that one out of seven will be solicited for sex by adults. On September 5, 2007, they announced a new program called SafeSurf that will be taught in all Florida middle schools and high schools beginning in Fall 2007, which will provide information on child-safe Internet usage. Florida has also passed a law with stronger penalties called the CyberCrimes Against Children Act of 2007.

Efforts to catch people through impersonation have not been restricted to children or law enforcement, but have also been used by newspapers. In 2005 there were many calls for Mayor West of Spokane to resign, and subsequently in December 2005 he was removed from office in a recall election, after a reporter for their Spokesman Review newspaper claimed that “On one recent occasion, West offered a man he believed to be an 18-year-old – whom he met online at Gay.com – gifts, favors and a City Hall internship, Internet dialogues retained by the newspaper reveal. The 18-year-old was actually a forensic computer expert working for the newspaper.” The story, including the Mayor’s response, is at www.spokesmanreview.com/jimwest/.

The biggest problem is that pedophiles can relatively easily obtain information that lets them establish relationships with children. One way, as I’ve discussed, is by going through directed chat rooms. Another is through information on online services, school awards, family Web pages, and so on. COPPA, the Children’s Online Privacy Protection Act of 1998, blocked online services from obtaining any information about children under the age of 13 without parental consent. Schools have become much better about not releasing any information that could be used to access students (there was a time when it was not uncommon for schools to publish student email directories). Despite efforts like these it is clear that computers have made it much easier for pedophiles to establish relationships with children, and it is up to parents and teachers to monitor their

8 The text can be found at http://www.ftc.gov/ogc/coppa1.htm.
activities, and up to the courts to severely punish predators to set examples. Unfortunately these kinds of protections have not been very effective at making a dent in this problem.

Unlike pornography, where the distribution of much of the most extreme material comes from outside the US borders, pedophiles usually want to meet their targets (for the worst reasons) which means that fortunately most are within range of the legal system where the solicitation occurs. This does not, however, always occur. E.g., a 32-year-old US Marine named Toby Studabaker was charged with meeting a 12-year-old girl from Manchester, England, in July 2003 after “meeting” with her since she was an 11-year-old on a chat room related to neopets\(^9\) and subsequently through sexually explicit text messaging and emails, described by the judge as cyber sex. The girl told her parents that she was going shopping with friends, but instead met Studabaker at Manchester Airport. From there they flew to Charles de Gaulle Airport in Paris, where they stayed for two days before taking trains and ending in Stuttgart. After an international search, which included the US Naval Criminal Investigative Service, she flew home after five days and he was arrested on his way to the US Embassy in Frankfurt by German police. In February 2004 he pleaded guilty to child abduction and inciting a child to commit an act of gross indecency. The case showed some of the problems of international crimes. His solicitation occurred when he was in Michigan and she was in Manchester. The (voluntary) abduction occurred in England. They both said that they only had sex in France, where it is not a crime for an adult to have sex with a child under 13, even though he first gave her vodka. He was charged both in Manchester Crown Court and also through a Federal Indictment issued in Michigan\(^10\). Since he couldn’t be charged with having sex with the child he was sentenced to only 4½ years in England, but it is expected that when he completes his sentence in England he will be extradited to the US where he will face the four count federal indictment. Reports say that the girl’s parents sold their story to the *News of the World* newspaper for £40,000 (about $75,000).

Using the Internet, and particularly using chat rooms, is a problem with no easy solutions. There are many examples of convictions for a wide variety of crimes against children which developed out of meetings over the Internet. This includes murder. For example, in 2002 Christina Long, a 13-year-old alter girl living with her aunt in Danbury CT, was raped (statutory) and murdered by a 25-year-old. She often met offline and had sex with people she had met in the chat rooms, including with the man who killed her (he claimed that her death was accidental during sex), which is an extreme case of why it is important for parents or guardians to monitor children who are using chat rooms. The number of chat rooms is so huge that they cannot effectively be monitored by law enforcement. For example, when I googled swimming and “chat room” I got about a million hits.

\(^9\) Neopets is a virtual pet site, [www.neopets.com](http://www.neopets.com), which claims a community of 70 million virtual pet owners around the world.

\(^10\) It is against the law for any United States citizen to travel abroad to engage in sexual activity with any child under the age of 18 (18 U.S.C. 2423b). Individuals who partake in this illegal activity are subject to prosecution in the United States even if they committed the crime on foreign soil. (Quoted from [www.cybertipline.org](http://www.cybertipline.org), the site of the National Center for Missing and Exploited Children.)
As an example of using a site designed to lure children, on July 21, 2006, a 23-year-old, Eric Fischer, was charged with 25 crimes, including rape, after using a vampire and gothic site to lure children to cemeteries for sex. At that time five potential victims aged 14-16 had been identified. He was caught because he attempted to solicit a detective from a computer crime squad who was pretending to be a 13-year-old girl.

There are a number of organizations that attempt to fight online predators. The most significant is the National Center for Missing and Exploited Children, which were mentioned above. They have mechanisms for reporting sexual predators who are seen online, either by parents or children, as well as providing advice to children with the titles “Know the Dangers,” “Situations to Avoid,” and “Surf Safer.” Parents are given advice on how to talk to their children about Internet use. This is a large organization whose primary mission historically has been to locate and assist missing children, but it has now added a major emphasis on combating, identifying, and convicting online predators.

In addition there are a number of federal initiatives, which include Internet Crimes Against Children (ICAC) task forces in all states. Montana is in theory part of a three state ICAC consortium centered in Utah that also includes Idaho, although the page for that ICAC doesn’t mention either Montana or Idaho.

5. Phishing

I first noticed phishing several years ago when I (and everyone else) began to get emails from Nigeria, which offered me huge profits if I could help the sender to launder some money through my American bank. There were two basic scenarios; either the sender was a bank officer whose bank had an account belonging to someone who had died without any relatives or estate, or he or she was a relative of a government member (heads of security services were popular) who had been killed in the recent revolution, but had left the relatives with access to millions of dollars that they couldn’t easily get out of the country. The emails told me that if they could launder the money through my bank account I could keep a reasonable percentage, usually about 25%, which would be something like $20 million. To make this possible I’d just have to give them my bank account routing information and they’d transfer all of the money there. Basically it wasn’t much better than the old “you have won a huge prize, but we need you to send money as a processing fee before we can send it to you” scam.

The word phishing usually refers to the somewhat more sophisticated approach where you get an email appearing to be from your bank or a related system like PayPal which explains that you need to update some personal information, and the email politely provides you with a link to the bank page so that you can do it. This approach, where the page looks just like the real bank page, has been remarkably successful as a mechanism

11 http://tcs.cybertipline.com/
12 http://attorneygeneral.utah.gov/ICAC/icacmain.htm
13 This is known as the 419 or 4-1-9 scam after the relevant section of the Nigerian penal code. U Penn has a great Web site on the scam at http://www.upenn.edu/computing/security/advisories/419scam.html
for stealing identities. The belief is that is because the average user has become accustomed to purchasing online by filling out credit card information. One way to combat this is to always check not only the listed URL, but to look at the link information to see the actual URL. It is very easy for me to include a link like `<A HREF="http://www.cs.montana.edu/starkey/ripoff"> www.etrade.com</A>` where my ripoff page will look just like the etrade page. A better solution is to never respond if you get an email from a financial institution that asks you to update personal information online. No reputable financial institution will ever do this.

Most spam filters try to detect phishing emails. One distinguishing feature of nearly all phishing emails is that the link and the displayed address don’t match, as in my example above, which is something that filters can recognize and penalize. In addition the link will often be to a computer on the filter’s blacklist, which will increase the penalty assigned to the email.

6. Pharming

With phishing the user takes some action, like responding to the former Nigerian police chief’s daughter with their bank information or clicking on a link that appears to be to PayPal and filling out personal information. Pharming is a much scarier, and much more sophisticated, approach since the user can be caught without doing anything unusual, and so the failure is with the system, not the user.

As an example of pharming, say that I bank online at Fred’s Bank. To pay bills I log onto www.fredsbank.com and enter my user account and password through an encrypted link, so even if someone is snooping packages off my network link, or on one of the computers that my message passes through on the way to the bank’s computer, I am still fairly safe.

Now the problem is that this all looks good, but it is a simplification of what actually happened. There isn’t really a computer whose network name is www.fredsbank.com, since the machine that I wanted is really called something like 123.234.135.012, which is its IP (Internet Protocol) address consisting of four numbers between 0 and 255 called octets. However although the IP address is all that will work for network routing I can’t remember it, and so the computer lets me use the domain name and then uses the Domain Name Servers (DNS) to convert to the numeric form.

The DNS system is what makes the Web and email systems practical. It is easily the world’s most heavily accessed and most dynamic database. It contains billions of entries.

---

14 There was a great Dilbert cartoon where Dilbert and his girlfriend are discussing how they would never enter credit card information over the Internet because people could steal the number, while he is paying for dinner with a card. The waitress returns with his receipt wearing jewelry and a mink coat.

15 As a historical note, we didn’t used to do email this way. Each computer had a .hostname file which users could augment with their own .hostnames file for machines that they accessed often. These files contained domain names and corresponding IP addresses. If you wanted to access any other system you
and it is accessed billions of times every day as people send emails or access Web pages. Millions of the entries are changed and nearly that many new entries are added to the database every day. To further complicate the database there are both static and dynamic IP addresses.\textsuperscript{16}

Most of the domain name system database is distributed since, for example, MSU is responsible for all domain names that end with \texttt{montana.edu}. So we maintain a local part of the DNS system which is responsible for all additions and deletions of machines on the \texttt{montana.edu} network, and for providing the IP addresses for anyone who wants to access one of our systems.

Now let’s get back to what happened when I accessed the bank. My Web server went to the local name server, and if I’d accessed the bank recently enough the IP address might have been in the name server’s cache\textsuperscript{17}, and my problems would be over. The more likely occurrence is that it wasn’t in the cache, and so my name server would go to one of the root name servers that are scattered around the world. The root name server would then (if it didn’t have \texttt{www.fredsbank.com} in its cache) direct us to a domain server for the .com domain. This server will then send my name server (again unless it has the address in its cache) to the Fred’s Bank name server, which will finally tell my name server the IP address that it needs.

This is a really amazing system, with lots of redundancy everywhere. So, for example, there are multiple .com servers, and they will list multiple \texttt{fredsbank.com} name servers, and the name server will try each one in order until it gets a response.

Now, having digressed through all of this, what can go wrong? When I used the system I relied first on my local name server, then a root name server, then a .com name server, and finally a Fred’s Bank name server, and any one of these servers can say “I’ve got this in cache, here is the result, don’t look any further” to my name server. What happens if one of these four systems has been hacked and an incorrect IP address for \texttt{www.fredsbank.com} has been forced into that system’s cache? This is what is now called pharming, and is a great security concern. When I access my online banking page the URL will show as \texttt{www.fredsbank.com}, but that will now be on the hacker’s computer, where there will be a page that is identical to the bank’s page, and where I will give them my login name and password. They will then give me some kind of Web failure message, and use the information to access my account. Even worse, they can assume that it is likely that I have accounts on some systems like etrade.com or orbitz.com, where I have selected the same user name and password, and will attempt to use the information to hack into a number of them as well.

\textsuperscript{16} When you go through an ISP you will usually be assigned a dynamic IP address which is only good for that session.

\textsuperscript{17} The entries in the caches have TTL (time to live) stamps to ensure that they don’t stay too long and block the propagation of changes.
How does one combat pharming? The HTTPS protocol gives a secure connection and passes a certificate to the user’s machine. If the certificate does not match correctly then your browser will display a security alert message saying that the name of the secure site doesn’t match the certificate, and asking you whether you want to proceed, with the options Yes, No, or View the certificate. Unfortunately most people have no idea what is going on and click Yes. It is almost always much better to select No.

A better approach to combating pharming is for sites to separate the user name and password component of the login process and to provide a user-specific response page to the user before they submit their password. I first saw this used by Bank of America for their electronic banking, and it is very slick. The user goes to www.bankofamerica.com and enters their account name. If this is a valid account the bank responds with a new page that has a picture of a familiar object\(^\text{18}\) that was given to the user when they first set up their online account, and they instruct the user to only enter their password if the picture is correct. A pharmer won’t know what picture to put on the page for any specific user that they capture, and so the pharming process will fail. The pharmer will have the user name for that system, but won’t have the password. Obviously this can only work if the bank is using a very large number of response pictures. The system isn’t, unfortunately, perfect since the pharmer could capture the user name, then use that to log into the bank site and get a response from the bank with the picture, and then give a page, with that picture, to the user. However this requires that the user won’t care about the response delay before they get the picture and password page.

Currently I don’t know of any other protections against pharming apart, of course, from efforts being made at all four levels\(^\text{19}\) of the name resolution process to block against hacking, but this is impossible to guarantee.

---

\(^{18}\) Something like London Bridge, the White House, or a grizzly bear, that is easy to remember.

\(^{19}\) There could be more levels than this since although domain names most commonly have three or four components, like www.montana.edu and www.cs.montana.edu, up to 127 components are allowed.
purchases and then I send them a bill at the end of the month. So say that I send out a bill printed off my spreadsheet with considerable detail like:

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/3</td>
<td>Flank Steak</td>
<td>$11.20</td>
</tr>
<tr>
<td>3/3</td>
<td>Cat food</td>
<td>$8.17</td>
</tr>
<tr>
<td>3/6</td>
<td>Wine and beer</td>
<td>$37.94</td>
</tr>
<tr>
<td>3/6</td>
<td>Gardening supplies</td>
<td>$28.37</td>
</tr>
<tr>
<td>3/22</td>
<td>Wine and beer</td>
<td>$23.39</td>
</tr>
<tr>
<td>3/22</td>
<td>Hot dogs and hamburgers</td>
<td>$28.00</td>
</tr>
<tr>
<td>3/22</td>
<td>Potato chips</td>
<td>$15.50</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL Charged in March</strong></td>
<td><strong>$162.57</strong></td>
</tr>
</tbody>
</table>

Most people are untrusting enough to check over each item and if, for example, they didn’t have a party on the 22\textsuperscript{nd} or don’t own a cat then they’ll challenge those charges. However if the seven items all appear to be legitimate then they will pay me the $162.57. The problem is that the actual total of the seven items is $152.57\textsuperscript{20} and so I’ve just ripped off $10. To put this in perspective, if you have a credit card when was the last time that you added up the charges to make sure that the computer got the total correct? If you do check the numbers and come to me to complain I’ll say “I’m so sorry – it’s those damned computers” and fix the problem and then program the computer so that it never cheats you again, although I’ll continue to rip off everyone else. The really nice thing about this is that you’ll accept this excuse and won’t report me to the cops.

The thing that makes this so successful is that (a) people trust computer output – if the bill had been hand written they would have checked the total, and (b) they also don’t trust computers – “the computer screwed up” always works as an excuse. Most computer crimes work well because of this dichotomy.

There are many variations of this basic scam. For example if I employ you I have to make weird deductions from your monthly pay check for taxes, FICA, etc. Each month I show on your pay slip how much has been deducted and then at the end of the year I give you a W-2 which shows the totals of all of the deductions and you use that to do your taxes. The trick is to deduct more each month than I should, and keep the difference, but then on the W-2 I list the total amount that I should have deducted. Unless you keep your 12 pay slips and add up the deductions to compare them against the W-2 (which nobody does, except for people who are nervous about this crime) you’ll never know that I’ve robbed you. And, of course, if you do find out then I’ll fall back on the “the computer screwed up” excuse.

Other standard computer crimes, which used to be common, have now been largely eliminated through better auditing of financial institutions. For example, it used to be possible in many places for programmers to truncate partial cents from all financial transactions in, say, a bank. So if the interest due to me was $33.555, then I would only get $33.55 credited, which I didn’t care about. However if the bank had a million

\footnote{People sometimes notice the cents, and so it is a good idea to make them, and possibly the least significant digit, correct.}
transactions per month then this would generate $5,000 without a home, with the potential that it could be put into the programmer’s account.

Some crimes that are usually classified as computer crimes are so distant from the computer that maybe they shouldn’t classify. One classic, which doesn’t work any more, is to have a large number of deposit slips printed up that look like the blank deposit slips available in the bank, but with your account number printed on the bottom. Replace the blank slips by ATMs with these slips, and wait a day before removing all of the money that people have kindly put into your account. Or there was the person who printed checks with the picture and text describing one bank but with the computer codes describing another. He then opened a large number of bank accounts using these checks for initial deposits. The way that banks work is that if the deposit doesn’t bounce within five days it is accepted. The checks went to the clearinghouse which sent them to the bank shown in the computer code. People there looked at the checks and saw that they were for another bank, and sent them back to the clearinghouse. This cycle continued until the checks were so worn out that they had to be processed by hand. By then the accounts had been cleaned out.

On the grand scale some of the accounting practices that have ultimately brought down huge companies like WorldCom, with the loss of billions of dollars, would not have been possible without computers helping to disguise the movement of money.

There are an amazing number of different computer crimes that have been committed, many of which are very creative. A large book is needed to get heavily into this topic. The only real way to reduce the level of crime is (a) not trust computers or programmers and (b) greatly increase the penalties for some of these so called white collar crimes. In most cases your penalty for robbing a bank of $1000 with a pistol will be very many times the penalty for robbing the same bank of $1,000,000 by hacking their computer.

8. Viruses and Macro Viruses

In theory viruses, worms, and Trojan horses are different, since a virus needs the user to take some action to activate it, a worm replicates itself without user action, and a Trojan horse pretends to be useful software that it expects the user to install. However the word virus is also commonly used to include worms and Trojan horses. In these sections I’ll distinguish between the three most of the time.

A virus will usually be attached to some executable file, and becomes active when the file is executed. A typical situation is that it is in an email attachment like the Love letter or Melissa viruses, and when the recipient opens the attachment to the email the virus is let loose. At this point the level of damage varies dramatically; at the extreme level the virus infects some files, destroys others, and prints out gloating messages before your system dies. E.g., you didn’t know that you had the Yankee Doodle virus until your computer played the music at 5:00 p.m., and by then your system was, when this first spread around the world in 1989, a pain to repair. (According to Symantec it is still in the wild, but with
low occurrence.) Kinder viruses will just do something so that you know you have been caught, but will do no damage. Most viruses will then attempt to replicate themselves through emails to everyone that they can find in an address book.

Let’s take the love letter virus as an example. The original virus hit worldwide in 2001, and since then there have been 82 variants according to Symantec. Targets receive an email with the subject I LOVE YOU, and a request to open the love letter attachment, which is named love-letter-for-you.txt.vbs. The first warning should have been the VB script extension, but people got tempted by the idea of getting a love letter, and subsequent analysis said that it spread world-wide in about five hours, clogging mail systems and leaving a mess to clean up on millions of computers, including a very large number at MSU. For example Mike Malone, our former MSU President, opened a love letter from the Commissioner of Higher Education and got infected. The virus looks for passwords, which it emails back to its originator, distributes itself to everyone in the Outlook address book and through open IRC channels, and copies itself into all files ending with .vbs, .vbe, .js, .css, .wsh, .sct, .hta, .jpg, .jpeg, .mp2, and .mp3. Unless they are backed up, all of those files have to be destroyed.

So what does the user have to do with respect to viruses. Obviously the biggest imperative is that it is wrong to ever initiate a virus. Other than that, practicing safe networking is essential, which means that every computer should have anti-viral software written by one of the major groups (e.g., McAfee, Norton, Panda, or one of many others), and that this software be updated daily. Major anti-viral software systems now combat over 100,000 viruses, and so there is a lot of dangerous stuff out there. In addition, if your operating system has security patches available, make sure that you are automatically informed about all new patches. Finally, take extreme precautions before opening an email attachment.

It might appear that as soon as a new virus comes out most systems will become infected before the anti-viral software companies can react. There are three reasons why this isn’t usually true. First, most viruses are variants of previous viruses and will be picked up by the anti-viral programs, second most viruses share some characteristics in terms of how they behave, which lets many be detected without updates, and finally it takes some user action, usually very ill-advised, to initiate the virus.

If you think that you have been infected with a virus, despite taking precautions, before you do anything else check out information about the virus (this will, as I’ll discuss later, also stop you from doing something stupid if you are conned into believing a hoax). I use either http://www.symantec.com/avcenter/vinfodb.html, or http://vil.nai.com/vil/default.asp, to get the latest information about any virus that is in the wild, but there are many reputable information sources. These pages include instructions on how to deal with the virus, and these instructions should be followed precisely.

A macro virus is the fancy name for a virus that is a macro embedded into a document. Applications like Word support a very powerful macro system, and the macro virus in, say, a Word document will be executed and let loose any time that you open the document. In addition it can place itself in any document that you create in the future
until you clean off your system. Standard anti-viral procedures will protect you as well as is possible against macro viruses.

9. Worms 21

On November 2nd, 1988, Robert Morris, a graduate student at Cornell, let loose a program on the Internet which became known as the Internet Worm. By midnight the Internet had been brought to its knees, and any machines that were still able to run were being disconnected from all network connections. For about three days, before a team managed to reverse engineer the worm and the fix had been spread, we didn’t have a network.

Morris apparently didn’t anticipate how far the virus would go (the word worm wasn’t popular until after this disaster), and apparently he didn’t intend that it would take down systems. He claimed that he expected that all infected systems would just get a small extra process that would never be noticed.

The Internet Worm didn’t destroy any files or retain passwords that it broke, and its only real problem was that it created so many processes that it overwhelmed the Unix systems that it ran on.

There are many very detailed descriptions of the worm available. Its primary method of attack was through the Unix fingerd command, which uses gets() to get the input, which is up to 512 bytes long. Unfortunately on most systems it didn’t check for an overflow, so the worm called fingerd with a 536 byte argument. The extra 24 bytes overwrote the system stack, which is where the system goes once the fingerd completes, still in root mode. This let the worm bring itself into the computer. It also used rsh to attempt to get into systems, and if all else failed it took advantage of some debugging code that had been accidentally left in the Berkeley sendmail distribution which let mail be sent to processes.

Once it was in a machine the Worm used a number of methods to get the names of other machines to attack. It looked in /.rhosts and /etc/hosts.equiv, as well as using the /etc/passwd file to get to individual user accounts and their .forward files.

During its sleep routines the Worm would listen for other worms running on the same system, and if one was found then one of them (random selection) would get a flag named pleasequit set. However there were a couple of significant bugs in this part of the code, and also one in seven worms were exempt from pleasequit control, and so this is the failure that caused infected machines to be overwhelmed with processes.

Initial estimates of the cost of the Worm ranged from $10 million to $150 million, based on the loss of network access, but later estimates dropped below $1 million. Of the 60,000 computers on the network at that time someone guessed that 10% were probably

21 The name worm comes from the tapeworm in the movie Shockwave Rider.
taken out by the Worm, and so a figure of 6000 computers became fixed in the newspapers. Later this was also believed to be high, and 2000 is now the accepted figure. However nearly all other computers were pulled off the network, and so the impact went far beyond those 2000 computers. Robert Morris was convicted under the computer Fraud and Abuse Act, and sentenced to three years of probation, 400 hours of community service, a fine of $10,050, and costs.

Some good things came out of the Worm. Teams of programmers got to work without break to try to stop the worm. The group at Berkeley had a partial fix within 12 hours, MIT came up with a complete definition a bit later, etc.\textsuperscript{22} It was recognized that in the future this needed to be better coordinated, and so as a result of the Morris Worm CERT (the Computer Emergency Response Team) was created, which has added significant security to all systems.

Protection against worms is the same as for other viruses. I.e., keep up to date on virus protection and on system security updates and you are relatively safe.

10. Trojan Horses

As the name implies, a Trojan horse is something that looks safe but contains a hidden danger. The classic example is when anti-virus software, typically available free on the Web, actually contains a Trojan horse. Usually it is let loose when it is installed or in some cases run.

Once a Trojan horse is let loose it can do any of the usual nasty things to the host computer that a virus can do like doing strange things to the desktop, deleting files, playing music, or whatever. A favorite for Trojan horses seems to be to do nothing obvious, but to establish a back door so that the designer of the Trojan horse can sneak into the computer.

A Trojan horse is not self-replicating, and won’t, for example, send copies of itself out through email, but relies on copies of the software that it is attached to being installed on computers.

Usual safe computing practices should take care of Trojan horses, as long as you have your anti-viral software set to scan all disks, CDs, email attachments, and downloads before they are executed. As a general rule, don’t trust software downloads that look too good to be true.

\textsuperscript{22} It turned out to be very difficult to distribute these solutions because the networks were toast.
11. Hoaxes

Virus hoaxes are spread by user naivety. An email arrives warning the user about some problem, and usually says that the warning message should be forwarded to everyone possible. The jdbgmgr.exe hoax is a typical example, which was first seen in April 2002, but which still makes the rounds occasionally. The email for this hoax comes in as an apology from someone who believes that they are infected by a virus, which they think has gone to everyone in their address book, including you. They tell you to search your C: drive for a file called jdbgmgr.exe, which will have a teddy bear icon. If you find it they tell you (a) DO NOT OPEN IT, (b) delete it and (c) forward the email to everyone in your address book because you will have infected them. They also say that the virus is not detected by Norton or McAfee. The problem is that everyone finds that they have this file, because it is part of the standard Windows distribution (it is the Microsoft Debugger Registrar for Java). As a result there are two effects; one is that ridiculous numbers of emails have gone round the net because of this hoax, and b) lots of people have deleted a relatively unimportant file. The factor that made this hoax particularly effective is that jdbgmgr.exe does, for some reason, have the cute little teddy bear icon.

Some hoaxes are distributed because of the good intentions of many computer users. The classic example of this is the Craig Shergold\(^{23}\) story. The emails that still go around now describe him as a young boy dying of cancer with the goal of getting into the Guinness Book of World Records by getting more greeting cards than anyone else, with the effort being supported by the Make a Wish Foundation. The poor Foundation still gets flooded by cards and letters in support of this effort. There was a time, in 1989, when the story was mainly true, and Craig was nine, and in fact he did get into the Guinness Book, receiving 16 million cards by 1990, although it never had anything to do with the Make a Wish Foundation. He was in the Great Ormond Street Hospital in London, and for years they also got overwhelmed with mail. He was subsequently cured of his disease, and is now a healthy young adult. Letters to Make a Wish are so numerous that they get recycled, unread. This particular hoax has gone through at least four resurgences, and at times it has been claimed that it should be considered a virus or worm because of the way that it has replicated over the Internet.

So what is your responsibility if you receive an email which you suspect is a hoax? You should go out onto a reputable site and check the validity of the claims in the email. If the email contains things like “your entire hard drive will be destroyed” or “Norton and McAfee cannot detect this” then it is usually best to start off at a hoax site like http://preview.symantec.com/enterprise/security_response/risks/hoaxes.jsp. If you find that it is a hoax then immediately (but politely) respond to the sender and to everyone else who received the mail and include the URL of the page that described the hoax. Don’t just ignore the email, since the goal is to cut off the hoaxes as quickly as possible.

\(^{23}\) Later forms have had many variants on the spelling of his last name.