

Contents

1	Introduction (0:00–1:00)	2
2	Security (1:00–100:00)	2
2.1	Session Hijacking (1:00–24:00)	2
2.2	SQL Injection Attacks (24:00–32:00)	3
2.3	Viruses and Worms (32:00–60:00)	4
2.4	More on Internet Security (60:00–70:00)	5
2.5	Data Security (70:00–100:00)	6

1 Introduction (0:00–1:00)

- Quizzes have been graded and mailed out. Check your mail in the next few days.
- Grades will be viewable in the next few days via a tool on the course website. If you login and don't see your quiz grades and wiki/blog posts, it's probably because we don't have your FAS username on record. E-mail help@computerscience1.net and we'll fix it.

2 Security (1:00–100:00)

2.1 Session Hijacking (1:00–24:00)

- When you're at Starbucks or the airport and you connect to the public wireless access point (WAP), you might try to navigate to a website only to be redirected to a page where you must agree to some terms and conditions and pay a fee for internet access. After that, you're free to browse sites like Facebook, MySpace, Gmail, and the course website. When you login to any of these sites, you should pay attention to the URL in the address bar. If it begins with **https**, you know that any information you send to that website is first being encrypted with a generally hard-to-crack algorithm called SSL.
- However, with some websites, you'll notice that the **https** becomes **http** after you login. The practical reason for this is that SSL is computationally expensive which means that it takes more CPU cycles to serve content to a single user. More CPU cycles per user of course means that fewer users can access the site at the same time, a major concern for very popular websites like Facebook.
- As an analogy, consider sending a friend a letter. If the letter is plaintext, he'll be able to read it and understand it very quickly. If it's encrypted, he'll have to spend time decrypting it before he can read and understand it. Thus, there is a time cost inherent with encryption.
- Most banks actually use **https** for all pages past login because the cost of a compromised account is greater than the cost of supporting SSL. Without SSL, how might an account be compromised? If a malicious user positions himself between your computer and the internet, he can intercept all of your private information that is being transmitted in the clear. To prevent this, you could connect to a virtual private network (VPN)—for example, you could login to Harvard's network remotely—such that all of your internet traffic is encrypted. At home, adding a password to your wireless network will accomplish the same.
- HTTP, the protocol by which website content is transmitted to your computer, is a stateless protocol. This means that once your computer has

connected to a server and downloaded the content you requested, the connection is severed. But what happens when you request another page from the same site? If the connection has been severed, will you be required to login again? Experience dictates that you won't. But how, then, does the site remember who you are? Most sites that you login to will plant a small amount of information—for example, a very large random number—called a *cookie* on your hard drive. Whenever you connect to that site, this cookie will be sent to the server to identify you. Think of a cookie as a handstamp that tells a club or an amusement park that you've already been admitted, so you should be allowed back in.

- Recall from a few weeks ago, that a request to the server, as examined by Live HTTP Headers, consists of several lines of text, the first looking like `GET / HTTP/1.1`. A few lines below this, a line that begins with `Cookie:` will transmit the cookie to the server. During the initial connection to the server when you logged in, the server sends back as part of its response a line that begins with `Set-Cookie:`. This of course plants the cookie on your hard drive.
- The danger with cookies is that they can be intercepted if your connection to the internet is not encrypted. If you're connected to a public WAP and a website that doesn't use SSL, your cookie will be transmitted in the clear. A malicious user who uses a *packet sniffer* to get a hold of your cookies can then send them to the server himself and thereby impersonate you. Through this process of *session hijacking*, a malicious user will be able to see your Facebook profile and login to many other websites as you!

2.2 SQL Injection Attacks (24:00–32:00)

- Even if your connection is encrypted to prevent session hijacking, your data isn't necessarily safe on the server. Though it's beyond the scope of this course, there is a type of attack known as a SQL injection attack whereby a malicious user can enter a special username and password and have access to many other users' data on certain websites. This attack is relatively easy to prevent, but not all websites actually do. David and Dan's friend (wink wink) actually used this recently in a restaurant in New Jersey in order to gain access to its WAP. They alerted the manager and sadly he didn't even seem to care.
- Although we won't delve too deeply into this kind of attack, suffice it to say that SQL is a language used to ask for data from a database, which is really just a large conglomeration of users' information. A SQL query to retrieve a username from a database looks something like this:

```
SELECT * FROM users WHERE username = '...';
```

The ... is where the username the user provided at login will be inserted into the query. But what if instead of typing in a real username, a mali-

cious user typed something like `' ; DELETE FROM users ; '`? If the website is not protecting against this, the malicious user's injected SQL will actually delete a lot of data from the website's database.

- Obviously, the defense against SQL injection attacks would be to search the user's input for dangerous keywords like `DELETE` and dangerous characters like semicolons and quotation marks. In fact, this defense is relatively easy to implement in the form of a function—think of a function as a machine that takes input and gives output—which will massage the user's input until it is safe. Still, many websites *don't* implement this defense and thus remain vulnerable.

2.3 Viruses and Worms (32:00–60:00)

- Viruses are a form of malicious software or malware. Truth be told, differentiating between viruses and legitimate software is often difficult. Consider a program that scans your hard drive and tells you which files are the largest so that you can free up some disk space. This is arguably a useful program. But what if that program deletes the largest files without your knowledge or consent? You may think that sounds like a stupid example, but think how many times you've clicked Yes to a program prompt without really reading what it says.
- Worms are distinguishable from viruses in that they are self-propagating. Part of its programming is to find other computers on the same network that it can infect. Viruses, on the other hand, require that a user actually do something like download and open a file.
- Have you ever opened a Word document and read a warning that says the document contains macros? A macro is a small computer program in and of itself. These macros can introduce a lot of helpful shortcuts for you as you compose your document. However, they can also be malicious in nature and if you choose to allow them to run in a document that you don't necessarily trust, you're asking for trouble.
- The fundamental problem with many operating systems is that they don't properly *sandbox* applications. When you install a program, often you are handing it the keys to your computer and allowing it to install files in multiple different locations on your hard drive. This makes it extremely difficult to remove the program entirely if you ever decide to. Increasingly, Apple and Microsoft are moving toward a model where applications live in a single folder unto themselves and when executed, only have access to a small part of RAM.
- Trust is a thorny issue in the world of software. If you download a free Solitaire program, what are the chances that it will cause annoying pop-up ads or worse, that it contains a keystroke logger? Unencrypted web traffic has its own perils, but consider that downloading software is essentially

like letting someone into your home. You are implicitly trusting them! For that very reason, David will never do any kind of online banking on a shared computer. Who knows what's been installed on that machine?

- Saving your password in an unencrypted Word document is never a good idea! Even if you restrict physical access to your computer, malicious software may have access to this document unbeknownst to you.
- And please, please, please don't put your password on a post-it note next to your computer. A better solution would be to use a trusted program that allows you to enter all your passwords in a single encrypted file which can only be opened when a strong master password is provided.
- Even if you receive an e-mail from a friend containing something for you to download, you should be very wary about downloading it. There's a good chance that friend's e-mail account or entire computer was compromised and so will yours be if you go through with it.
- Two-factor authentication is a good method for stepping up security and combating some of these issues. When logging in, you'll be prompted not only for your password but also for something that you physically have on your person. For example, in addition to your password, you must provide the digits from a random number generator you have on your keychain or a passcode that was texted to your cellphone. It's less likely that an adversary will steal both your password and your random number generator or cellphone. Bank of America offers two-factor authentication for its online banking services.
- Bank of America also offers SiteKey authentication whereby once you've entered your username, they provide you with a small picture so that you can verify you are actually connected to Bank of America's website and not a fraudulent site. However, this doesn't protect against a man-in-the-middle attack. If a malicious user positions himself between you and Bank of America's servers, he can simply intercept this SiteKey, present it to you, and then steal your password when you enter it in.
- A few years ago, a malicious user sent out a mass e-mail with a link to **bankofthevest.com**. Many people actually clicked on it and entered their credentials thinking that it was Bank of the West's website. Best practice is not to click on any links in e-mails from supposedly reputable sites. Just go to the website by typing in the URL yourself or, if you don't remember the URL, Googling it.

2.4 More on Internet Security (60:00–70:00)

- Interestingly, with certain software, we can view details about all of the computers connected to the same router as we are. Realize that even if you are asked to agree to certain terms and conditions before you can use a public WAP, it doesn't necessarily mean that the connection is encrypted.

- One of the pieces of information we can see about computers connected to the same router is the MAC address. These MAC addresses are serial numbers that identify network cards in computers. On some home routers, you can restrict access to the network by MAC address, only allowing certain ones through. This is a good security measure, but it's definitely not foolproof. It's easy enough to sniff the MAC addresses of other computers connected to the network and then change your own to one of those that's allowed.
- MAC addresses generally have specific prefixes that identify the network card manufacturer. This is how a packet sniffer might know that you have a Hewlett Packard network card just by observing your network traffic.
- Similarly, back when cellphones were analog, they would identify themselves to a nearby tower by broadcasting a large unique number. Malicious users very quickly figured out how to sniff these numbers and use them to make free calls.
- We've already discussed how cookies are sent to the server as a means of identifying the client. Cookies have gotten a bad rap, but if they contain very minimal information such as a username or even just a very long string of numbers, they're really not a big cause for concern.
- Besides cookies, however, a lot of identifying information is sent to servers in the headers of a website request. The **User-Agent** field, for example, tells the server what browser you're using.
- In addition to the HTML which controls how a website looks, a server might send JavaScript code which adds interesting functionality to a site. For example, Google's autocomplete feature is powered by JavaScript. JavaScript can also be used to mine more information about your computer, including the size of your screen, what plugins you have installed, what version of Flash you're running, etc.

2.5 Data Security (70:00–100:00)

- One of David's coolest experiences as a grad student was working for the Middlesex County District Attorney's office doing data recovery on confiscated hard drives. Amazingly, even hard drives that have supposedly been wiped or formatted still contain a lot of information that can be recovered. A colleague of David's once did a study on a large number of hard drives that he purchased secondhand. Although they had all supposedly been erased, he was able to recover dozens of credit card numbers, addresses, and other bits of personal information.
- Recall that a hard drive consists of platters which have bits written onto their magnetic surfaces. Deleting files is generally a two-step process involving dragging them to the trash and then emptying the trash. But is

that file actually erased from the hard drive at that point? The answer is no. The locations of files on disk are stored in a table so that the operating system can know where to find a file when you ask to open it. When you move a file to the trash and empty the trash, its entry in this table is deleted, but the actual bits that represent the file remain intact on the hard drive platters. This was a design decision that was made years ago when overwriting the bits of a file would have been very time-consuming. Nowadays, some operating systems, including Mac OS, have begun to offer the Secure Empty Trash option, which will not only delete a file's entry in the location table, but will also overwrite the bits of the file.

- How does forensic data recovery work? Files of a given format generally have a well-known header and footer. For example, the sequence of bits that represents the beginning of a Word document is constant across all Word documents. Thus, if you are a forensic investigator looking to recover Word documents whose location on disk has been deleted, you can simply scan the hard drive for that sequence of bits that defines the start of a Word document.
- When David's colleague was doing his investigation of secondhand hard drives, he found that even those that had been erased with commercial software still left a great deal of information that could be recovered. So what's the solution for a typical user who is planning on selling or giving away his old computer? Your best bet is to physically destroy the hard drive, if you can. Some companies offer the ability to drill straight through the center of the drive, rendering it useless. Another option is to use programs that will erase the entire hard drive rather than attempting to selectively delete files. [Derek's Boot and Nuke](#) (DBAN) is a good option. Erasing the hard drive properly will take a long time, especially if you choose to make multiple passes. Know that it's never been shown conclusively that making multiple passes over hard drives is actually more secure than making a single pass.
- Another step up in security is hard drive encryption. It costs a few milliseconds in terms of computation time, but it prevents anyone with physical access to your hard drive from stealing your data. Even if someone steals your computer or your hard drive, they still need your password in order to decrypt the data on your hard drive. Right now, hard drive encryption is much easier on Macs, which come with native support for it.
- Question: as a time saver, could you randomly flip half the bits instead of all of them? Yes, that's a possibility, but saving a few hours of time at the risk of having your data compromised is probably not the best tradeoff.
- Question: if you're curious about how well a given piece of software works for erasing files on your hard drive, you'll probably have to run tests on your own. Chances are, though, if the software is well known, someone

out there has already run the same tests, so you might just search around to find the answer.

- As Dan advises, a particularly satisfying way of ensuring that your data is never compromised is to open up the hard drive and smash it with a hammer. Take out all your frustrations Office Space style!
- *Formatting* a hard drive is slightly different from erasing it. Formatting will either write a few zeroes to the beginning of the disk in order to clobber the partition table (so that the operating system will read the disk as being empty) or check the integrity of all the sectors on the disk. Thus, although it may take a long time and you may get a warning that the disk will be erased, formatting probably will not actually destroy all the data on the disk.
- When the iPhone first came out, there were some security concerns with the data that it might contain on its hard drive. Thankfully, Apple has since implemented a secure erase feature in the menu options so that if, for example, you need to trade in your phone because it's broken, you won't have to worry that you're handing all your e-mail and contact info to an Apple employee. Securely erasing your iPhone now takes a matter of seconds because it only involves deleting the key with which all of the data on the hard drive is encrypted.
- Question: using a magnet is not really a reliable way of erasing data on a hard drive since it may be reversible.
- If your hard drive is damaged, you face the problem of being unable to use software to reliably erase it. You may find yourself in a situation where the hardware manufacturer requests the original drive in order to replace it. But data can still be recovered from damaged hard drives. If you're paranoid enough, you might rather buy a new hard drive so that you can keep the old one and destroy it. Here again, the option of encrypting your hard drive would be a good solution as well, since you could trade in a damaged but encrypted hard drive knowing that even if data can be recovered from it, it won't be readable.
- As we mentioned earlier, the only way to ensure that physical access to your hard drive doesn't compromise your data is to encrypt the hard drive. Software like TrueCrypt enables you to cordon off a section of your hard drive to be encrypted. On the front end, you will see this section as a separate hard drive into which you can drag any files you want to be secure. Good encryption will make this data appear to be random if you read the actual bits that compose it.