CI583 Data Structures and Operating Systems
Security

Just because you're paranoid, doesn't meant they're not out to get you

Security affects all aspects of computing at technical, business and social levels.

Database servers, web servers, mobile devices, etc., all have their own security and privacy implications.

The most common cause of security breaches is social engineering and inadequate password protection.

Just because you're paranoid, doesn't meant they're not out to get you

Passwords have been used to log in to computer systems since the early 1960s, probably beginning with MIT's timesharing system `CTSS`.

Security was not a big concern for the designers though and passwords were stored in plain text, in a file that anyone could print...

Authentication

UNIX was the first system to introduce one-way hashing of passwords.

For a password, p, only the hashed version of the password, $f(p)$, is stored on the system.

Authentication

f is a hash function that produces (mostly) unique values and whose inverse, f^{-1} is extremely difficult to compute.

Thus, losing the contents of the password file is not a big problem.

(See `/etc/passwd` and `/etc/shadow`.)

Authentication

This approach did not guard against a dictionary attack.

If most passwords are based on common words then we can compute $f(w)$ for every w in the dictionary, compare this against the password file, and discover most passwords.

This is made harder to do by adding a *salt*, or random value, to the password.

Now the function is $f(p + salt, salt)$ and the crackers' job is made harder but still feasible, given time.

Longer passwords are stronger

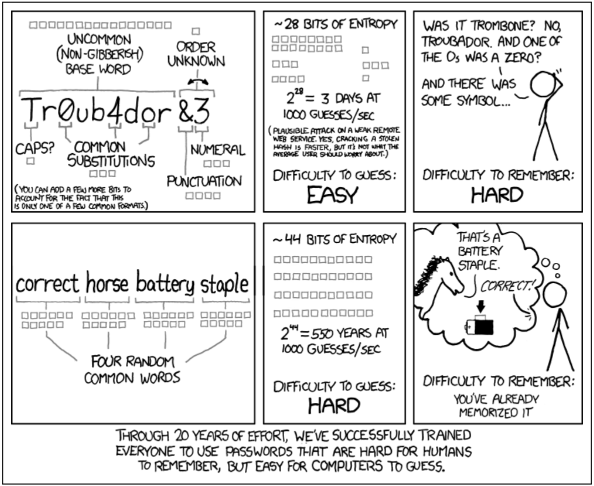
UNCOMMON (NON-GIBBERISH) BASE WORD
ORDER UNKNOWN
Tr0ub4dor&3
CAPS?
COMMON SUBSTITUTIONS
NUMERAL
PUNCTUATION
(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)
~28 BITS OF ENTROPY
2^28 = 3 DAYS AT 1000 GUESSES/SEC
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)
DIFFICULTY TO GUESS: EASY
WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?
AND THERE WAS SOME SYMBOL...
DIFFICULTY TO REMEMBER: HARD
correct horse battery staple
FOUR RANDOM COMMON WORDS
~44 BITS OF ENTROPY
2^44 = 550 YEARS AT 1000 GUESSES/SEC
DIFFICULTY TO GUESS: HARD
THAT'S A BATTERY STAPLE.
CORRECT!
DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT
THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

Image ©`http://xkcd.com/936/`

Authentication

New forms of authentication are rapidly emerging. Some of them are intended to be more convenient than passwords (e.g. smartphone unlock patterns) and some are intended to be more secure (e.g., online banking card readers, biometrics). We are not likely to discover a silver bullet.

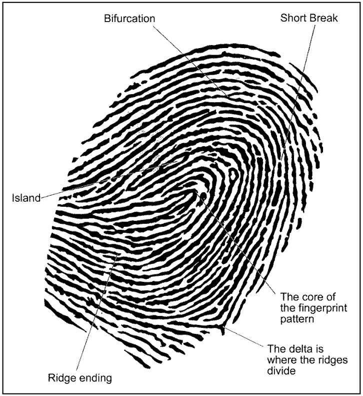
Bifurcation
Short Break
Island
The core of the fingerprint pattern
The delta is where the ridges divide
Ridge ending

Image ©http://www.zdnet.com/

What is security from the OS point of view?

From the point of view of the OS, security has a more narrow focus and the impact of a security breach is more critical – it may affect the whole system.

We have covered some of these means already: *keeping the OS safe from user processes* and *keeping processes safe from each other*.

What is security from the OS point of view?

The means we have talked about ways for doing this – e.g. access control at the file system level, base and bounds registers, the use of zeroed-page lists.

These are sufficient, most of the time, for preventing accidental security breaches.

What is security from the OS point of view?

Malicious security breaches come in a number of forms:

1. Trojan horse: a user runs a malware program that appears to do one thing but instead, or additionally, does something else. E.g. an update to IE that installs a rootkit, enabling a remote user to control the victim.
2. Virus: a piece of malware that arrives attached to a host, such as an email attachment with an enticing name like ILOVEYOU. Running the host cause the virus to run, possibly causing damage but also attempting to spread the virus to other computers, such as by emailing all contacts in the user's address book.

What is security from the OS point of view?

3. Worm: similar to a virus but does not need the user to run a program in order to become attached. Often spread from server to server by techniques such as buffer overflow. This occurs when the sender overwrites a fixed-size buffer with executable code that rewrites the return address of the current stack frame, so that the next instruction is now the code that runs the worm.
4. Denial of Service (DoS): disabling a computer by flooding it with requests, occupying so many resources that the computer is unable to do anything else.
5. Trap doors (or *back doors*): functionality hidden within a program that can be used to gain admin rights, for instance by entering a secret key sequence.

What is security from the OS point of view?

The idea of the trap door was first discussed by Ken Thompson in his Turing Prize acceptance speech, Trusting Trust: `http://cm.bell-labs.com/who/ken/trust.html`.

Thompson produced a version of the `C` compiler which would insert special code whenever the `login` function was compiled, and which would also insert this adaptation whenever the compiler was used to compile a new version of the compiler.