Mal-ware (malicious software) =

=
- software with unwanted / undesirable behaviour
- software capable of replicating itself. (explains the generic term virus)

So, viruses = software, computer programs.

Undesirable behaviour
- annoying: playing sounds, displaying pop-up banners, etc.
- stealing resources: using the "infected" computer to launch attacks on other machines, using the computer to send SPAM, etc.
- stealing sensitive information: e-mail addresses, passwords, personal data for identity theft!

- to become active, all mal-ware need to be executed on the victim's computer. Once they are run, the programs make sure they will always be executed, & they keep running on the

(modify the boot sequence)

(Week 13) D2

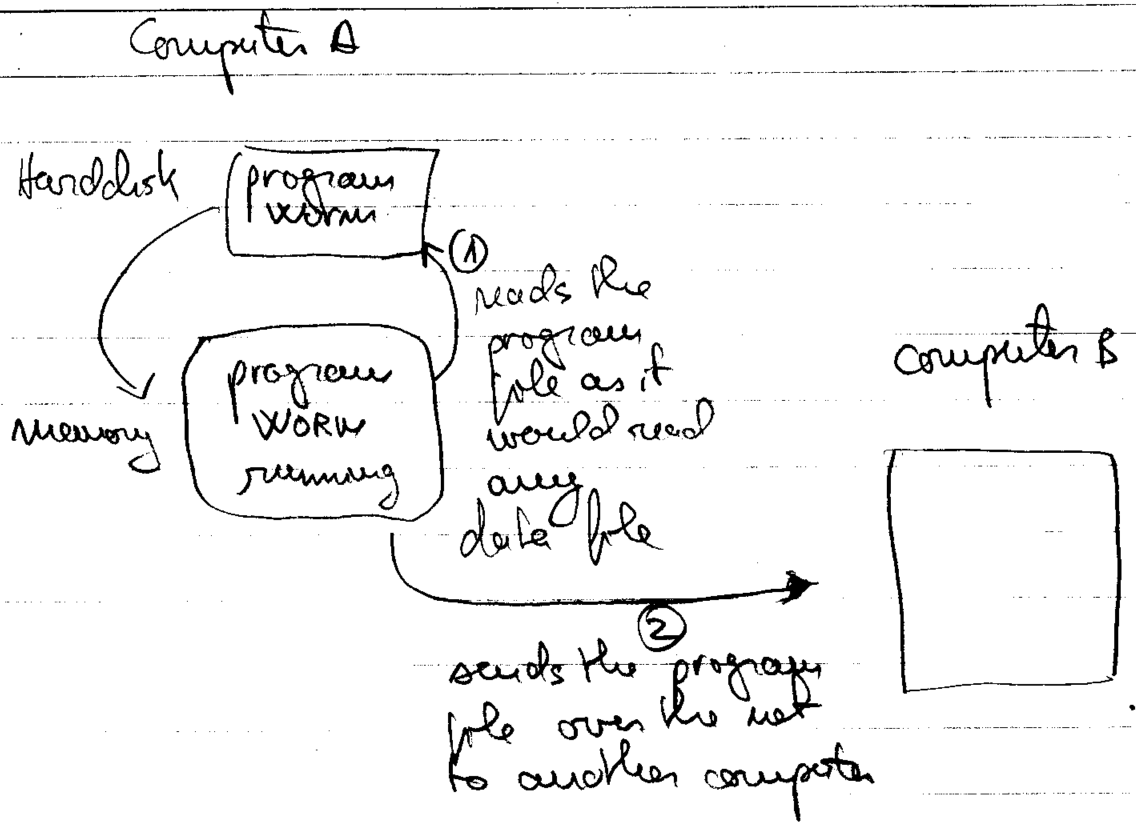
background (not noticed immediately by the user)
We say "the computer has been infected by a virus".

- viruses: computer programs that self replicate using other files that move from computer to computer.

- worms: similar to a virus, except it is a stand-alone program (doesn't attach itself to other computer files).

A self replicating program: when run, the program can copy its set of instructions (the program copies itself) to other files on the same computer (a virus) or to other computers over the network (a worm).

Self replication
- computer programs can manipulate data & files, for ex by copying data.
- self replication occurs when that data is the program itself.


Computer A
Harddisk
program WORM
①
reads the program file as it would read any data file
memory
program WORM running
Computer B
②
sends the program file over the net to another computer

- Trojan horses : mal-ware disguised as a useful program.

How does a computer get "infected" with mal-ware? The only way = the malicious code has to be run on the computer.

-> for VIRUS : user runs the executable files infected by the virus & which belong to legitimate computer applications (eg WORD, excel...)

-> for Worm : by exploiting vulnerabilities of computer programs & the operating system. The user may NOT be involved in this process!

→ for Trojan horses: user runs the Trojan horse thinking it is a useful program.

Very often malware has characteristics from all three categories, so the distinction is not always clear.

ex Storm Worm (or Nuwar virus).
(see PPT slides)
- characteristics of Storm Worm
  - → infects .exe & .scr files (virus behavior)
  - → downloads & executes files via a P2P network (also called bot net from "robot network")
  - → replicates via spam e-mail (social engineering)

• P2P network = collection of servers that communicate & transfer files between each other, creating a "virtual network" where users share files generally.

To have access to this network, a user needs to run the server on her machine

- typically P2P networks are used to distribute pirated copies of media, etc... (lack of central server & can't be detected & shutdown).

The Storm Worm -> uses P2P net to "control" the infected machines (the atacker can send & execute any program on the infected machine.
ex: • to harvest bank passwords, credit card info etc.
- • to send spam
- • etc.

- Vulneralbilities in OS:
  - bugs that allow an attacker to execute code on a remote computer without intervention of the user of that computer.

    (ex) Viruses, Trojans - need to be executed by the victim to get activated & infect a computer

    Worm - uses the OS to get executed & infect the computer.

Typical example of vulnerability:
- buffer overflow

ex: Any program reading data from a network connection, uses a variable to store the data. This variable = buffer:

```
make "a [1,2,3,4]      ← ex: buffer to
...                      read 4 # from
readData :a              a network connection
...
```

- sometimes, if the data coming from the network is more than a certain # of bytes (ex 4 numbers in our logo example) this data gets written over (in the memory of the computer) ~~over~~ the program instructions.

- if the data sent contains binary code of computer instructions, these will be executed, thus infecting the computer with the "virus" sent over the network connection.

How to protect ourselves ?

- General rule
  - like a tourist in a foreign city: keep your wallet close (don't jump to key in credit card # early), obey the law, use common sense.
- Specific rules
  - use antivirus software (at least AVG antivirus free version)
  - enable firewall (or install if using old versions of Windows eg NT, 95, ..)
  - update (or enable autoupdate functions) to protect against vulnerabilities
  - do not open e-mail attachments or click on URL from e-mail messages if you

are uncertain of the origin of e-mail or "news is too good" :).

<u>intellectual property</u>

-> information resulting from human creation process (artistic works, dramatic works, literary works, software).

- <u>software bought</u> = not owned, it is leased. Users buy the right to use the software (intended to fight piracy, originally).
  - -> license restricts how many copies of the program can be run, where it can be installed, etc
    (ex: Microsoft's EULA (end user license agreement) -> install 1 computer + 1 additional laptop

Types of software licences

- freeware: allowed to use, copy, redistribute @ no fee.
- shareware (try before you buy): license to use software for evaluation for a limited period; you pay author for use after trial period.

- GNU license (Free Software Foundation)
  -> promotes user's rights to use, study, copy, modify & redistribute modified computer programs
  Definition of Free Software
  (free as in "free speech" not "free beer")
  4 basic freedoms for software users:
  - freedom to run the program for any purpose
  - freedom to study & adapt the software to your needs (source code = available)
  - freedom to redistribute copies
  - freedom to improve the program & redistribute it

  Obs • when redistributed, these rights must follow the software
  (notion of copyleft, as a legal way to enforce this - same function as copyright).

  (Founder: Richard Stallman)

- Open Source License (Open Source Initiative)
  -> source code must be provided
  -> allows modifications & redistributions of modified versions; possible to restrict the modifications to clearly distinct version # of the original software