Malware

What is malware

- Malware is
  - software intended to intercept or take partial control of a computer's operation without the user's informed consent.
  - It subverts the computer's operation for the benefit of a third party.
- Also called spyware.
  - The term “*spyware*” taken literally suggests software that surreptitiously monitors the user. But it has come to refer more broadly to any kind of malware,
- Malware covers all kinds of intruder software
  - including viruses, worms, backdoors, rootkits, Trojan horses, stealware etc. These terms have more specific meanings.

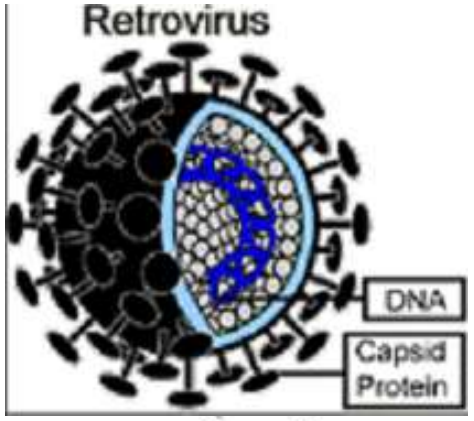
The purpose of malware

- To partially control the user's computer, for reasons such as:
  - To subject the user to advertising
  - To launch DDoS on another service
  - To spread spam
  - To track the user's activity ("spyware")
  - To commit fraud, such as identity theft and affiliate fraud
  - For kicks (vandalism), and to spread FUD (*fear, uncertainty, doubt*)
  - . . . *and perhaps other reasons*

How malware spreads

- Trojan horse
  - a malicious program that is disguised as useful and legitimate software. Can be part of, or bundled with, the carrier software.
- Virus
  - Self-replicating program that spreads by inserting copies of itself into other executable code or documents.
- Worm
  - Self-replicating program, similar to virus, but is self-contained (does not need to be part of another program). Spreads by exploiting service vulnerabilities.
- Drive-by
  - installs as side-effect of visiting a website; exploits browser vulnerability.

Detail from "The Procession of the Trojan Horse in Troy", Giovanni Domenico Tiepolo



Effects of spyware

- As well as the intended effects, malware has some side effects:
  - substantial loss of system performance, due to burden of dozens of parasitic processes
  - loss of system stability (→ crashes, hangs)
  - difficulty in connecting to Internet
  - unusability due to plethora of pop-ups
- Computers having malware generally have dozens of separate components.
  - Symptoms often due to cumulative effect: interactions between components
  - Some types of spyware disable software firewalls and anti-virus software, and reduce browser security settings, opening the system to further *opportunistic infections*.

Trojan horses

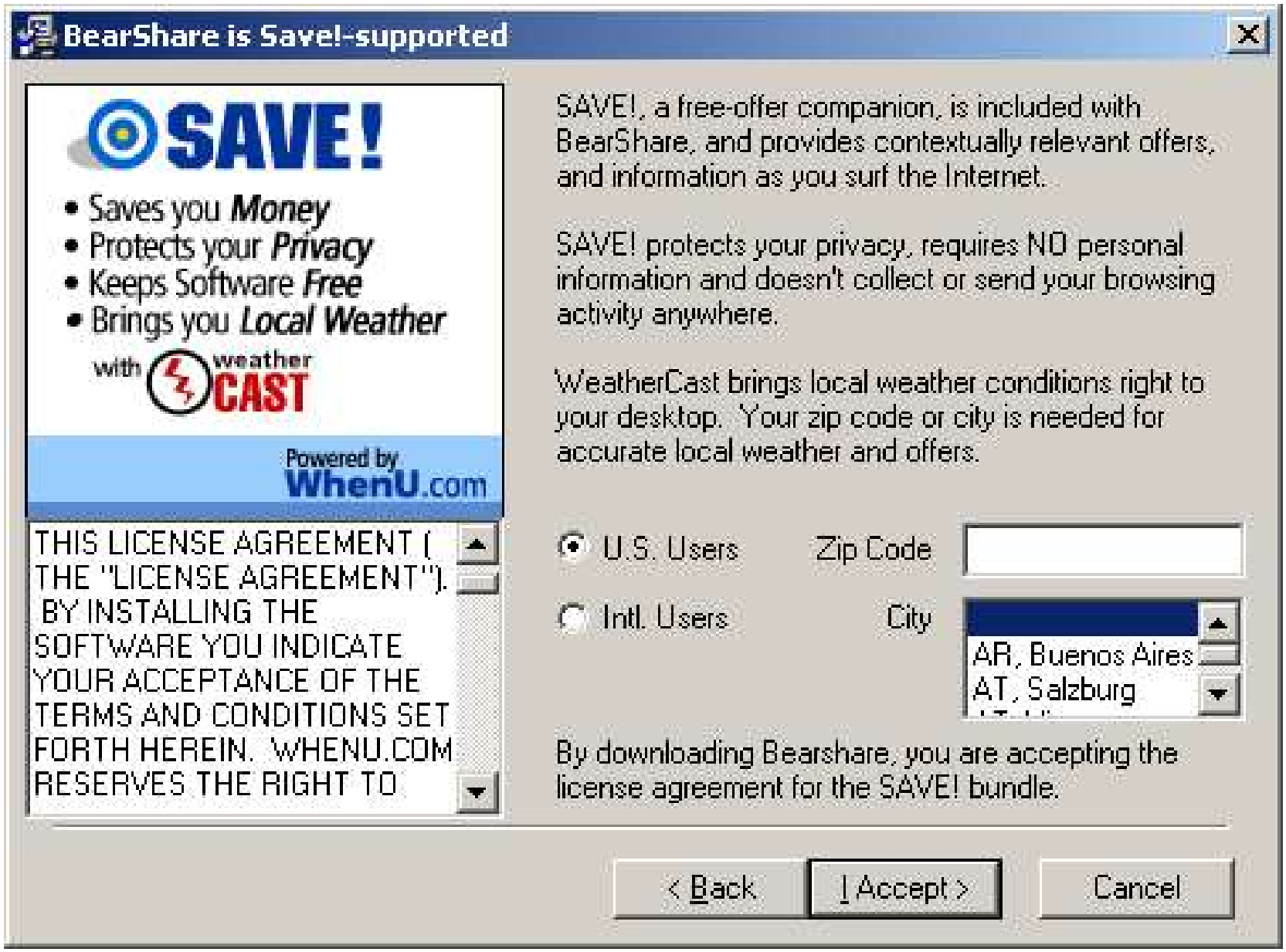
- Malware that the user installs inadvertently
  - not realising that they are installing anything
  - or thinking that it is useful/desirable software

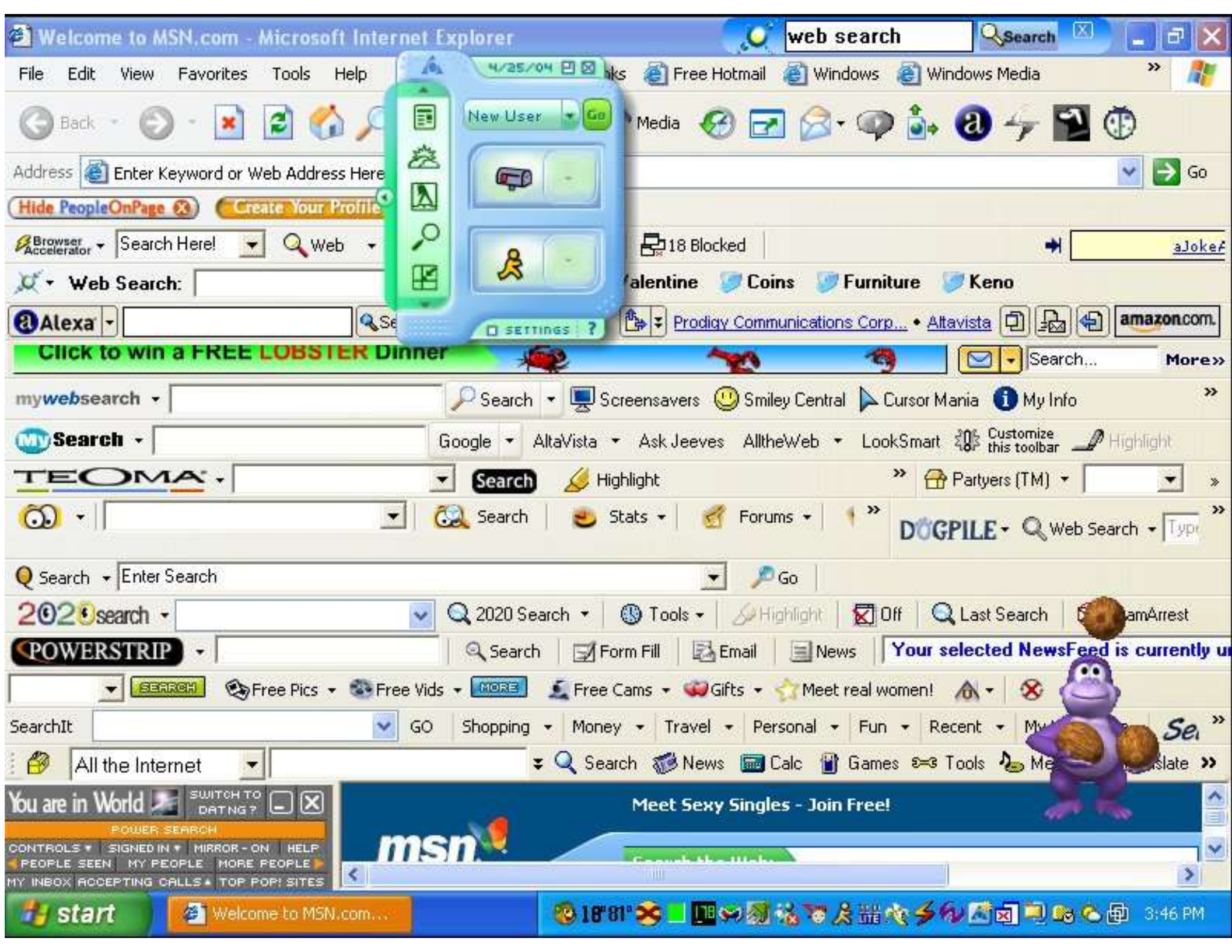
Trojan horses: appearing as desirable utilities

- Programs may be presented as a useful utility
  - for instance as a "Web accelerator" or as a helpful software agent.
- Users download and install the software, only to find out later that it can cause harm.
  - For example, Bonzi Buddy, a spyware program targeted at children, claims that:
    - *He will explore the Internet with you as your very own friend and sidekick! He can talk, walk, joke, browse, search, e-mail, and download like no other friend you've ever had! He even has the ability to compare prices on the products you love and help you save money! Best of all, he's FREE!*
  - constantly resets user's homepage to bonzi.com, tracks browsing habbits about the user, and serves advertisements.

Trojan horses: bundled with desirable utilities

- The BearShare file-trading program is bundled with WhenU spyware.
  - In order to install BearShare, users must agree to install "the SAVE! bundle" from WhenU, which is spyware. The installer provides only a tiny window in which to read the lengthy license agreement. Although the installer claims otherwise, the software transmits users' browsing activity to WhenU servers.


BearShare is Save!-supported
SAVE!
• Saves you Money
• Protects your Privacy
• Keeps Software Free
• Brings you Local Weather
with weather CAST
Powered by WhenU.com
THIS LICENSE AGREEMENT (THE "LICENSE AGREEMENT"). BY INSTALLING THE SOFTWARE YOU INDICATE YOUR ACCEPTANCE OF THE TERMS AND CONDITIONS SET FORTH HEREIN. WHENU.COM RESERVES THE RIGHT TO
SAVE!, a free-offer companion, is included with BearShare, and provides contextually relevant offers, and information as you surf the Internet.
SAVE! protects your privacy, requires NO personal information and doesn't collect or send your browsing activity anywhere.
WeatherCast brings local weather conditions right to your desktop. Your zip code or city is needed for accurate local weather and offers.
U.S. Users
Zip Code
Intl. Users
City
AR, Buenos Aires
AT, Salzburg
By downloading Bearshare, you are accepting the license agreement for the SAVE! bundle.
< Back
I Accept >
Cancel

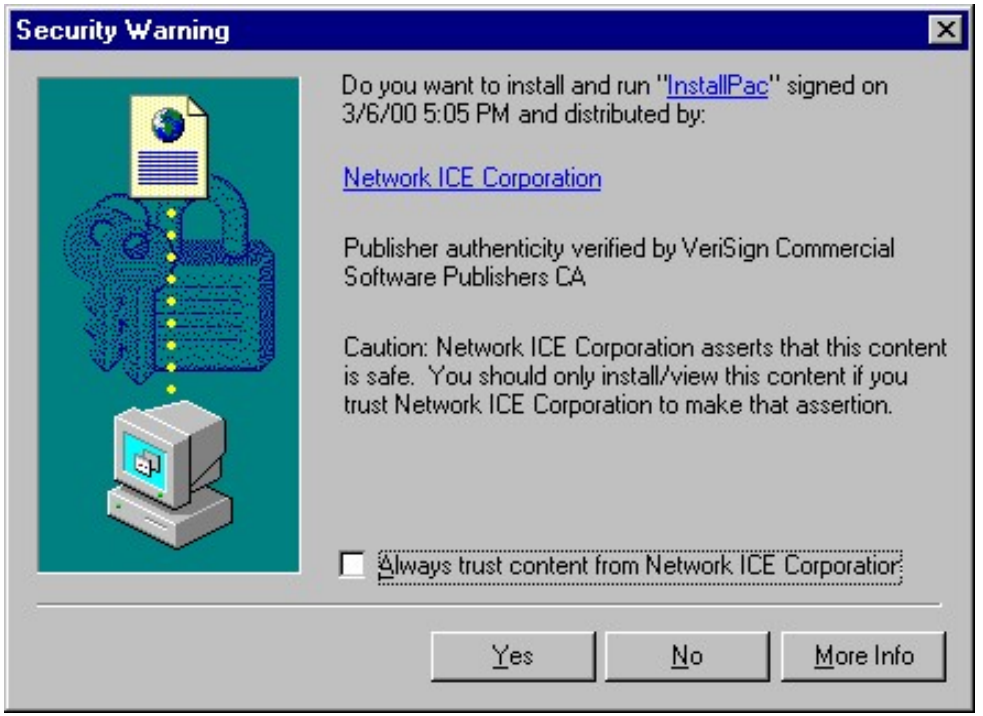

Welcome to MSN.com - Microsoft Internet Explorer
web search
Search
File Edit View Favorites Tools Help
Free Hotmail
Windows
Windows Media
Back
Address
Enter Keyword or Web Address Here
Go
Hide PeopleOnPage
Create Your Profile
Browser Accelerator
Search Here!
Web
18 Blocked
Web Search:
Valentine
Coins
Furniture
Keno
Alexa
Prodigy Communications Corp...
Altavista
amazon.com
Click to win a FREE LOBSTER Dinner
Search...
More
mywebsearch
Search
Screensavers
Smiley Central
Cursor Mania
My Info
MySearch
Google
AltaVista
Ask Jeeves
AlltheWeb
LookSmart
Customize this toolbar
Highlight
TEOMA
Search
Highlight
Partyers (TM)
Search
Stats
Forums
DOGPILE
Web Search
Search
Enter Search
Go
2020search
2020 Search
Tools
Highlight
Off
Last Search
POWERSTRIP
Search
Form Fill
Email
News
Your selected NewsFeed is currently u
SEARCH
Free Pics
Free Vids
MORE
Free Cams
Gifts
Meet real women!
SearchIt
GO
Shopping
Money
Travel
Personal
Fun
Recent
All the Internet
Search
News
Calc
Games
Tools
You are in World
SWITCH TO DATING?
POWER SEARCH
CONTROLS
SIGNED IN
MIRROR - ON
HELP
PEOPLE SEEN
MY PEOPLE
MORE PEOPLE
MY INBOX
ACCEPTING CALLS
TOP POP! SITES
Meet Sexy Singles - Join Free!
msn
start
Welcome to MSN.com...
3:46 PM

Trojan horses: tricking the user into installation

- Web browsers are designed *not* to allow Web sites to initiate a download without explicit user consent (to prevent drive-by Trojans).
  - Instead, a user action, such as clicking on a link, has to trigger a download.
- However, links can prove deceptive for naïve users.
  - a browser pop-up may appear like a standard Windows dialog box. The box contains a message such as "Would you like to optimize your Internet access?" with links which look like buttons reading *Yes* and *No*. No matter which "button" the user presses, a download starts, placing the spyware on the user's system.

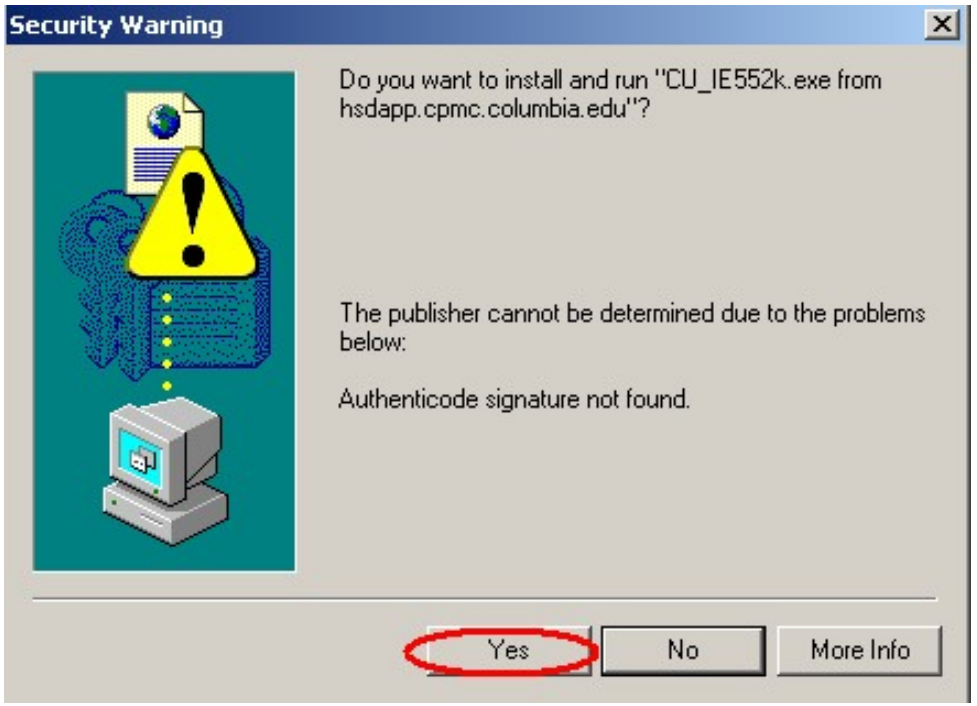
Authenticode

- Authenticode is a mechanism designed to verify whether downloadable code has been digitally signed by a reputable distributor.
  - *Do you want to install and run XXXXX signed on dd/mm/yy and distributed by ABC inc*
  - *ABC asserts that this content is safe. You should only install/view this content if you trust ABC Inc to make that assertion.*

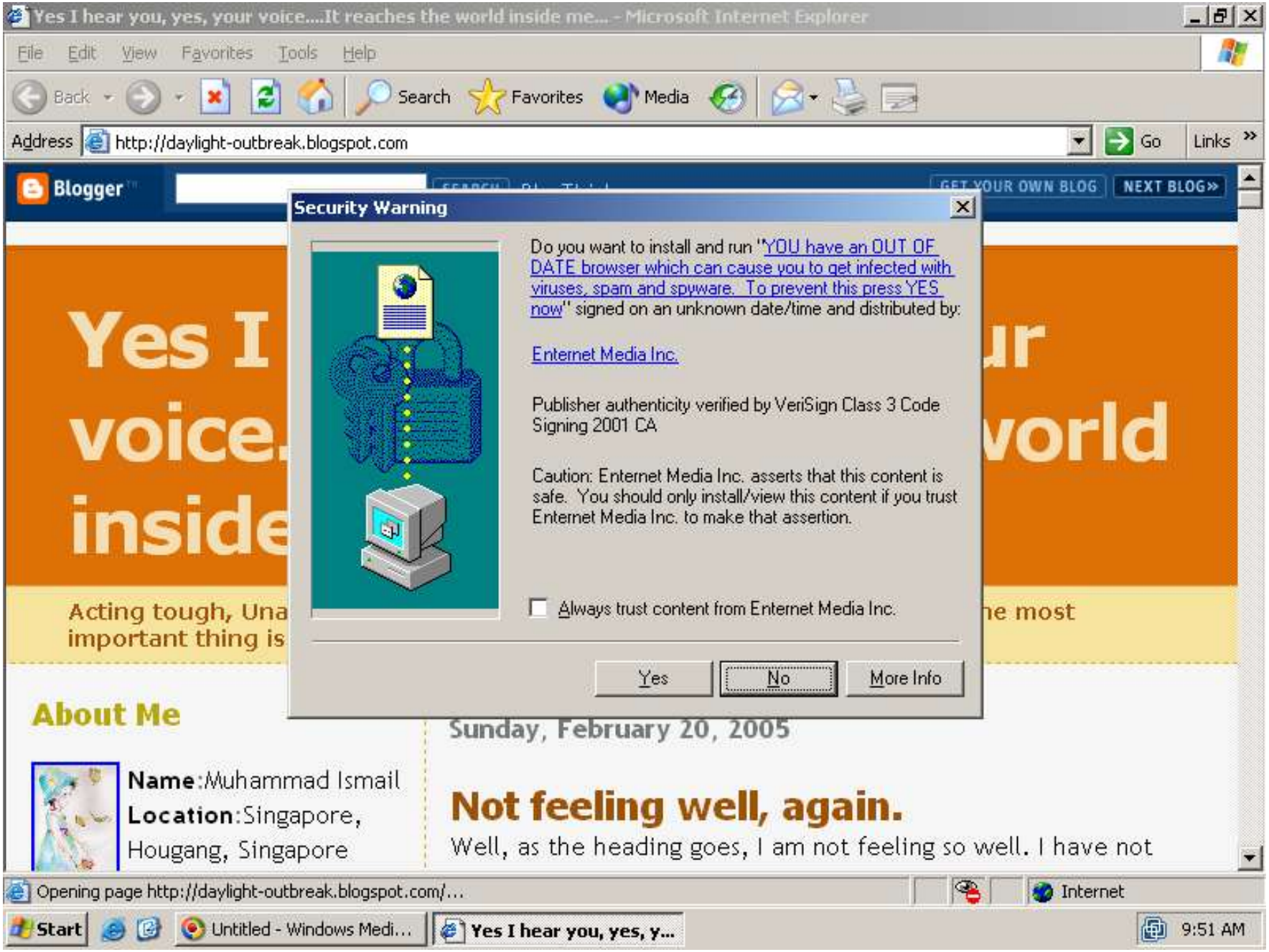


Authenticode

- Authenticode is poorly understood by users, and often it is not clear what is being said.



- Spyware merchants can try to confuse users by corrupting the already confusing sentence


Yes I hear you, yes, your voice....It reaches the world inside me... - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://daylight-outbreak.blogspot.com
Go Links
Blogger
GET YOUR OWN BLOG
NEXT BLOG»
Security Warning
Do you want to install and run "YOU have an OUT OF DATE browser which can cause you to get infected with viruses, spam and spyware. To prevent this press YES now" signed on an unknown date/time and distributed by:
Enternet Media Inc.
Publisher authenticity verified by VeriSign Class 3 Code Signing 2001 CA
Caution: Enternet Media Inc. asserts that this content is safe. You should only install/view this content if you trust Enternet Media Inc. to make that assertion.
Always trust content from Enternet Media Inc.
Yes
No
More Info
Yes I
voice.
inside
ur
world
Acting tough, Una
important thing is
he most
About Me
Name:Muhammad Ismail
Location:Singapore, Hougang, Singapore
Sunday, February 20, 2005
Not feeling well, again.
Well, as the heading goes, I am not feeling so well. I have not
Opening page http://daylight-outbreak.blogspot.com/...
Internet
Start
Untitled - Windows Medi...
Yes I hear you, yes, y...
9:51 AM

Viruses

- Replication by attaching to *hosts*, including binary executables files, disk boot sectors, documents that contain macros.
- Often evade detection by *self-modification*, which defeats *signature scanners*, because each infected file contains a different variant of the virus.
  - *Simple self-modifications*: exchanging subroutines in the code.
  - *Encryption*: virus consists of a small decrypting module and an encrypted copy of the virus code. Key randomly generated each time. Encryption can be simple XOR with random one-time-pad. But scanner might still detect the decrypting module.
  - *Polymorphic code*: the decryption module is also modified on each infection. No parts stay the same. Detection (by using an emulator, or by statistical pattern analysis of virus body) much more difficult.
  - *Metamorphic code*: virus rewrites itself completely each time. A *metamorphic engine* is needed. Virus very large and complex. W32/Simile >14000 loc; 90% of it is the metamorphic engine.

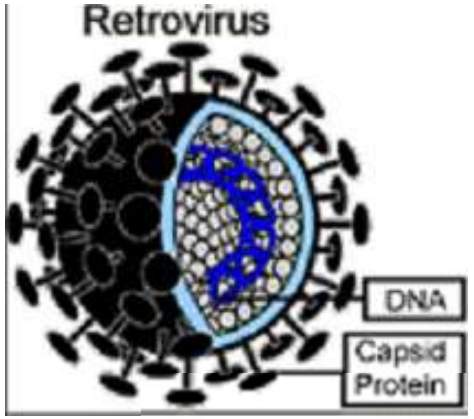
Worms

- Morris worm (Nov 1988)
  - Written by a student at Cornell University, Robert Tappan Morris.
  - Propagated through vulnerabilities in BSD Unix (sendmail, fingerd, rsh/rexec and weak passwords). Estimated 6000 computers infected. Effects intended to be benign, but caused huge overloading.
  - Morris convicted under the US Computer Crime and Abuse Act and received three years probation, community service and a fine in excess of $10,000. He is now an associate professor at MIT, as well as an entrepreneur.

Other notable viruses / worms

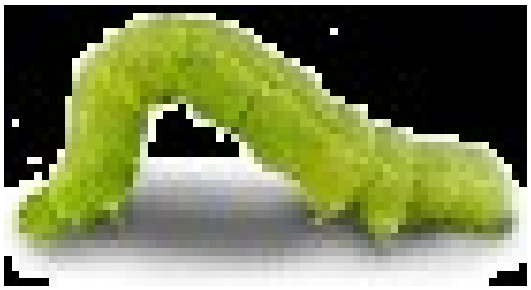
- VBS/Loveletter worm, also known as the "I love you" virus, May 2000
  - As of 2004, the most costly virus to business, causing estimated $10bn damage.
  - DDoS on Whitehouse website
  - Reasons for success: VB script; love
- Code Red worm, July 2001.
  - attacked MS IIS webservers.
- Sobig, August 2003
- Mydoom, January 2004.

The Sobig virus/worm



- Computer worm that infected millions of Internet-connected, Microsoft Windows computers in August 2003.
- Varieties Sobig.A through Sobig.F, which was the most successful.
- Spread by email with subjects “Re: Approved”, “Re: Details”, “That movie”, etc. Body is "See the attached file for details“, with attachment “details.pif”, “your_details.pif” etc.
- Replicate by using own SMTP agent engine. Email addresses gathered from files on the host computer.

Sobig continued



- Sobig.F programmed to contact 20 IP addresses on UDP port 8998 on August 26, 2003 to install some program or update itself. Unclear what this program was, but earlier versions of the virus had installed the Wingate proxy server, which provides a backdoor that can be used by spammers to distribute email.
- Spam distribution thought to be primary purpose.
- Microsoft has announced will pay $250K for information leading to arrest of creator Sobig.

Mydoom (Jan 2004)

- Spreads in several ways including as email attachment claiming to be transmission error, and copies itself to KaZaA's shared folder.
- Sets new record, exceeding record set by Sobig.
- Appears to have been commissioned by e-mail spammers. Author unknown.
- Payloads:
  - backdoor on port 3127/tcp to allow remote control of the subverted PC
  - A DoS attack against SCO Group and Microsoft
  - blocks HTTP access to Microsoft and about 60 antivirus sites, thus blocking virus removal tools and updates.

Mydoom timeline

- *26 January 2004*: Whole internet slowed by about 10%, web access by 50%. About 10% of all emails are by Mydoom.
- *1 February*: An estimated one million computers begin DDoS. SCO removes **`www.sco.com`** from DNS, moves its website to **`www.thescogroup.com`**.
- *3 February*: DDoS attack on Microsoft begins. MS prepared.
- *9 February*: Doomjuice, "parasitic" worm, begins spreading using MyDoom backdoor
- Mydoom spread slows due to programmer bugs; programmed to stop spreading on 12 February (A) and 1 March (B), but backdoors remain open
- *26 July*: A variant of Mydoom attacks *Google*, *AltaVista* and *Lycos*, completely stopping Google several hours.

Backdoor

- Software that allows access to a computer system bypassing the normal authentication procedures. For example
  - A special username and password hard-coded into the login program
- Such backdoors may be inserted by viruses, worms, Trojan horses or spyware.
  - A service listening on a particular IP port for remote instructions (e.g., Back Orifice)

Rootkit

- After installing the backdoor, the cracker wishes to avoid being undetected or removed by routine maintainance of the system. For that, she uses a rootkit.
- A rootkit is a set of modified versions of the usual utilities for administering the system, such as:
  - List all processes (unix: ps)
  - List logged–in users (unix: w, who)
  - List files (unix: ls)
  - Change passwords (unix: passwd)
  - Logging utilities

Combating malware

- Antivirus software
- Detecting Rootkits
- Fundamental limitations

Anti-virus software

- Initially: signature detection.
- But signatures are not enough!
  - Pattern matching
  - Automatic learning
  - Environment emulation
  - Neural networks
  - Data mining
  - Bayes networks
  - Hidden Markov models

Anti-virus software: TbScan

- TbScan looks at the following characteristics:
  - F = Suspicious file access. Might be able to infect a file.
  - R = Relocator. Program code will be relocated in a suspicious way.
  - A = Suspicious Memory Allocation. The program uses a non-standard way to search for, and/or allocate memory.
  - N = Wrong name extension. Extension conflicts with program structure.
  - S = Contains a routine to search for executable (.COM or .EXE) files.
  - # = Found an instruction decryption routine. This is common for viruses but also for some protected software.
  - E = Flexible Entry-point. The code seems to be designed to be linked on any location within an executable file. Common for viruses.
  - L = The program traps the loading of software. Might be a virus that intercepts program load to infect the software.
  - D = Disk write access. The program writes to disk without using DOS.
  - M = Memory resident code. This program is designed to stay in memory.
  - ! = Invalid opcode (non-8088 instructions) or out-of-range branch.
  - T = Incorrect timestamp. Some viruses use this to mark infected files.

TbScan (continued)

- TbScan (continued)
  - J = Suspicious jump construct. Entry point via chained or indirect jumps. This is unusual for normal software but common for viruses.
  - ? = Inconsistent exe-header. Might be a virus but can also be a bug.
  - G = Garbage instructions. Contains code that seems to have no purpose other than encryption or avoiding recognition by virus scanners.
  - U = Undocumented interrupt/DOS call. The program might be just tricky but can also be a virus using a non-standard way to detect itself.
  - Z = EXE/COM determination. The program tries to check whether a file is a COM or EXE file. Viruses need to do this to infect a program.
  - O = Found code that can be used to overwrite/move a program in memory.
  - B = Back to entry point. Contains code to re-start the program after modifications at the entry-point are made. Very usual for viruses.
  - K = Unusual stack. The program has a suspicious stack or an odd stack.

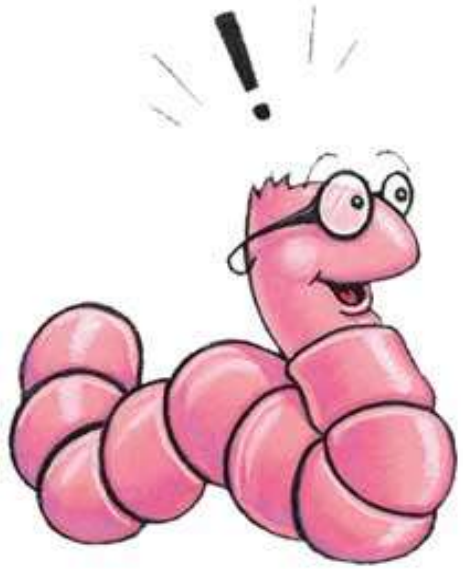
Sandboxes

- A sandbox is a security mechanism for safely running untrusted programs
  - Provides a tightly-controlled set of resources for guest programs to run in, such as space on disk and memory. Network access, the ability to inspect the host system or read from input devices is usually disallowed or heavily restricted. Cf. virtual machine.
- Examples of sandboxes are:
  - Applets are self-contained programs that run in a virtual machine or scripting language interpreter that does the sandboxing, for example in the browser.
  - Jails are a special kind of resource limit imposed on programs by the operating system.
  - Virtual machines emulate a complete host computer, on which an entire operating system can run.

Detecting rootkits

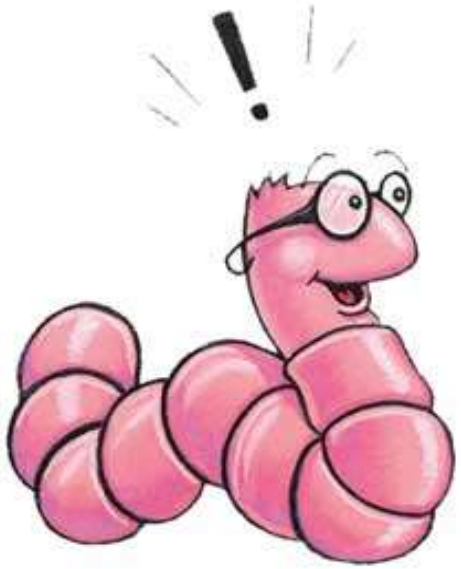
- Because they often hook into the operating system at the kernel level to hide their presence, rootkits can be very hard to detect.
  - There are inherent limitations to any program which attempts to detect root kits while those programs are running under the suspect system.
  - As with virus detection, the rootkit detection and elimination is an ongoing struggle between perpetrators and defenders. Examples of current tools (unix): chkrootkit and rkhunter
- Probably best to reinstall the operating system from scratch.

“Good” viruses/worms?



- A family of worms known as *Nachi* tried to download and install patches from Microsoft's website to fix various vulnerabilities in the host system — the same vulnerabilities that they exploited.
- It eventually made the systems affected more secure, but generated considerable network traffic (often more than the worms they were protecting against), and rebooted the machine in the course of patching it
- Worked without the consent of the computer's owner or user.
- Most security experts deprecate worms, whatever their payload.

Outlook



- Why we have so much malware
- Trends in malware
  - Crime
  - Mobile?
- The future

Why we have so much malware

- Users are ill-educated, resulting in distribution as Trojans and viruses
  - Because computers are fast-changing and still relatively new
- Software has vulnerabilities, resulting in distribution of worms and viruses
  - Because it is badly written or badly designed
  - Because the designers have historically favoured user convenience over security
- The PC is an open platform
  - Users can install software, in contrast with (*old fashioned*) mobile phones, mp3 players, set-top boxes, embedded computers, etc.

The threat of monopoly

- Another reason for the prevalence of malware is the *homogeneity of software*
  - Most computers run Windows, MS Office, MS Outlook Express, MS Internet Explorer
  - This makes the attacker's job very easy.
- In contrast, in the linux world, there is a plethora of rival distributions, office suites, email clients, browsers.
  - Makes the attacker's job much, much harder!

Open-source vs closed source

- It is often argued that
  - OS more secure because vulnerabilities have a much higher chance of being spotted, since hundreds of people around the world are scrutinising the source code.
  - CS less secure because very few people have access to the source code.
- But one can also argue that
  - OS *less* secure because attackers can see the code and find vulnerabilities to exploit.
  - CS *more* secure because attacker doesn't have access to the source code.
- However, this argument is "*security through obscurity*" and should be rejected.

Open-source phenomenon

- An attempt to plant a backdoor in the Linux kernel, exposed in November 2003, showed how subtle such a code change could be.
  - In this case a two-line change took the form of an apparent typographical error, which in practice gave the caller to the sys_wait4 function root access to the machine (see the external link below).
- The attack was detected well before the code was released

Trend towards crime

- Mikko Hypponen is chief research officer at F-secure, and worked on detection of Sobig, Sasser, etc.
  - “Worms aren't making the news these days, because they are not the right tool to use if you want to become rich by writing malware... Modern bots and trojans spread more stealthily, remaining below the detection radar... They infect your PC and wait for instructions.”

Mobile phone malware

- "Malware goes Mobile", Mikko Hypponnen, *Scientific American* 2006.
  - Reports on Cabir, a proof of concept virus.
  - Original did nothing (bragware), but variants dialled 1-900 premium numbers.
  - As of 2006, >300 different viruses (compared with 200,000 for the PC)
  - Spread by bluetooth (pestering user to accept)
  - As of 2006, all mobile malware exploits user naivity, not software vulnerabilities (why?)

The future

- The tension between *flexibility* and *security* will continue to introduce vulnerabilities
  - especially in emerging domains, such as wearables and multi-functional devices
- But users will continue to become better educated, and established software will continue to mature and become less vulnerable
- And PCs continue to become “locked-down”
  - draconian firewalls, sandboxes, fewer user priviliges
  - Trusted computer platforms
- Thus, things will slowly get better, but at a high price, and we will still see some spectacular attacks