

# Malware

Eamonn McGonigle

eamonn@lancomms.ie

<http://www.computing.dcu.ie/~eamonn/ca615>

# “Malware”

- Computer Viruses
- Worms
- Trojan Horses
- Spyware/Adware
- Browser Helper Objects (BHOs)/Hijacking
- Autodialers
- Keystroke Loggers
- Spam
- Root Kits

# Viruses

- **Characteristics of a virus are...**
  - **Self-replicating:** Originally via removable media such as floppy disks but increasingly now propagated through e-mail attachments (cf. “Worms”). Require some human action (e.g. sharing an infected disk, opening a file or attachment etc.) to facilitate replication
  - **Hidden:** The owner/user of an infected machine will usually be unaware of the virus’ presence (unless/until it accidentally or deliberately does some damage)
  - **Persistent:** Once infected, a victim machine will quietly reload the virus at each reboot. The virus may achieve this by several means, such as attaching itself to existing files, or surreptitiously inserting itself into the startup sequence
  - **Payload:** The virus will sometimes (but not always) have some purpose apart from its on self-perpetuation.

# Worms

- The characteristics of a worm are...
  - **Self-replicating:** ...but via network services rather than shared media. Will propagate without any human action
  - **Non-persistent:** “Pure” worms don’t attach themselves to files, so a reboot will eliminate them from the system (although prompt reinfection may follow unless some positive action is taken to prevent it)
- The first “major” worm was released on the 3<sup>rd</sup> of November 1988 by Robert Morris Jr. (commonly referred to as the **Morris Worm**)
- The most widely known contemporary worm is **Blaster**, which we will study later on

# Trojan Horses

- A program which ostensibly appears useful (e.g. a game or a utility) but which surreptitiously serves some other purpose (e.g. stealing or deleting files, removing security on a system)
- Relies on “social engineering” to entice the victim to activate it (sometimes employing reverse-psychology...the “AOL4Free” trojan)
- Not necessarily very sophisticated
- The distinction between a virus, a worm and a trojan is becoming increasingly grey (many current items of malware are referred to as “blended threats”, taking the best (?) features of all three)
- A recent variant is the “Backdoor Trojan”...NetBus, Back Oriface, SubSeven

# Spyware/Adware

- Software that monitors the activities of a user (or gathers information about a user) without the user's knowledge or permission (but see below).
- Typically relaying browsing habits or other information to a third party. A more active variant (Adware) will present unsolicited advertising to the user
- Sometimes delivered along with useful programs (cf. Trojan Horse). The program might be a conventional .EXE program or could be a Microsoft ActiveX control (e.g. a browser-based game)
- Users will sometimes “consent” to installation of Spyware/Adware by agreeing to the (usually long and opaque) license agreement for the useful software...the popular Kazaa filesharing (which we will examine) used this technique

# Browser Helper Objects/Hijacking

- Not actually malware but an enabling technology courtesy of Microsoft.
- This is the technology behind neat things like the Google search bar, the eBay toolbar
- It is also the technology behind malware which throws up popup ads unbidden, intercepts Internet searches, tracks Internet usage and lots more besides.

# Autodialers

- Often the payload of viruses or other malware, these will replace the user's dialup details (e.g. eircom.net...1893 150 150) with a premium-rate number (e.g. 1580 150 150). Will often then trigger a call to the new "service"
- Non tech-savvy users may not notice until the phone bill arrives

# Keystroke Loggers

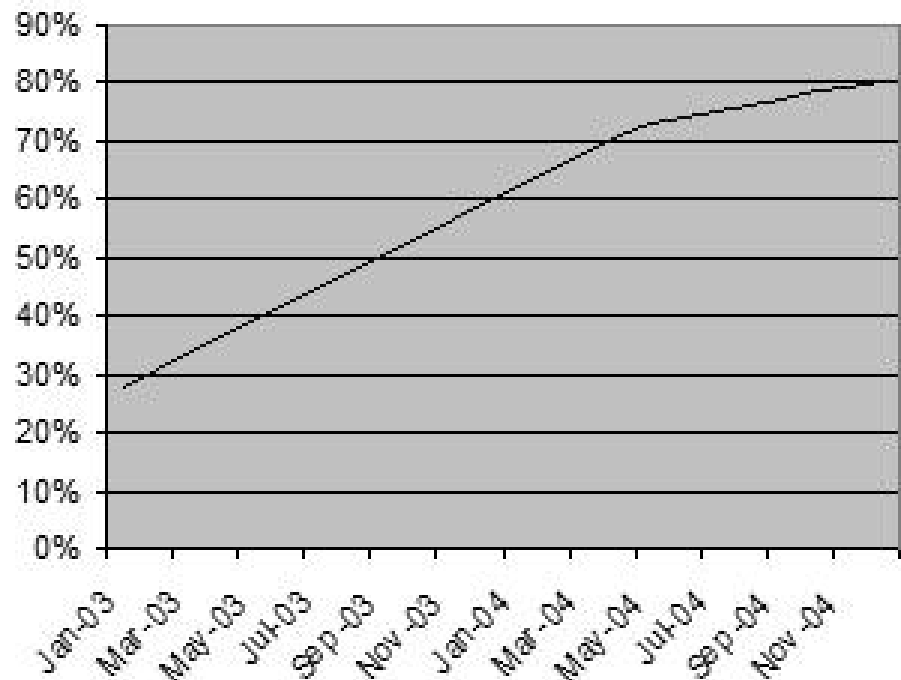
- Once again, delivered as the payload of some other piece of malware
- Intercepts keyboard device drivers and records every keystroke for transmission back to the attacker's location
- Obvious use is for capturing usernames/passwords, credit card and bank details and general spying.
- These are the main reason why you must use strong authentication (one-time passwords) for remote access from untrusted locations (e.g. Internet cafés, student labs 😊 etc.)
- Not always software...



# Spam

- Not exactly malware, but so pervasive that its causes, effects and remedies are worthy of study.
- Some estimates indicate that between 70% and 85% of the e-mail traffic on the Internet is spam.

Spam as a percentage of incoming email



# Root Kits

- Traditionally, a UNIX-specific term, but recent events (**SONY!**) have changed all that

# Viruses

- There are well over 60,000 known viruses
- Only a very small number of these (less than 1%) are actually “in the wild” at anytime. <http://www.wildlist.org> maintain a list of currently-active viruses (based on reported infections from many sources). The number of January 2006 is 804.

# Viruses

- The defining characteristic of a virus is that it attaches itself to “legitimate” programs
  - Boot sectors
    - These account for about 23% of known viruses (although the great majority are now “obsolete”
  - “Normal” executable files (parasitic viruses)
    - These account for about 13% of known viruses
  - Documents with scripting capability (macro viruses).
    - These account for about 65% of known viruses
    - Most new viruses are of this type

# Viruses – Milestones

- The first PC viruses originated in 1986.
- “Brain” was a boot-sector virus which
  - Changed the volume label of any floppy disk inserted into the machine (to “(c) Brain”) and copied itself into the executable area of the boot sector
- “Virdem” was a executable-file virus (infecting DOS .COM files). It was harmless...all it did was self-replicate.
- The first harmful virus was the Jerusalem virus (a.k.a. Suriv and the Isreali virus), released in October 1987
- Also in 1987, the Stoned virus was rampant in DCU (and almost everywhere else in the world). During 1 bootup in 8 it would display the message

“Your PC is now stoned! LEGALIZE MARIJUANA!”

...and the machine would freeze !

# Viruses – Milestones

- Tequila (1991) was the first *polymorphic* virus which would “mutate” in order to frustrate signature-based anti-virus software
- Michaelangelo (1992) was the first virus to prompt a media frenzy (and an anti-virus buying frenzy).
- 1992 was also the year when virus toolkits became popular making it much easier to produce full-featured polymorphic viruses
- Concept (1995) was the first (of many) Word Macro virus.
- Melissa (1999) was the first Macro virus to propagate via e-mail.
- ILOVEYOU (2000) became (at the time) the fastest-spreading virus ever, crippling many of the world’s mail servers
- 2003...annus horribilis – Slammer, Sobig, Blaster, Fizzer, Welchia, Nachi...and many more. Many of these are still in active circulation

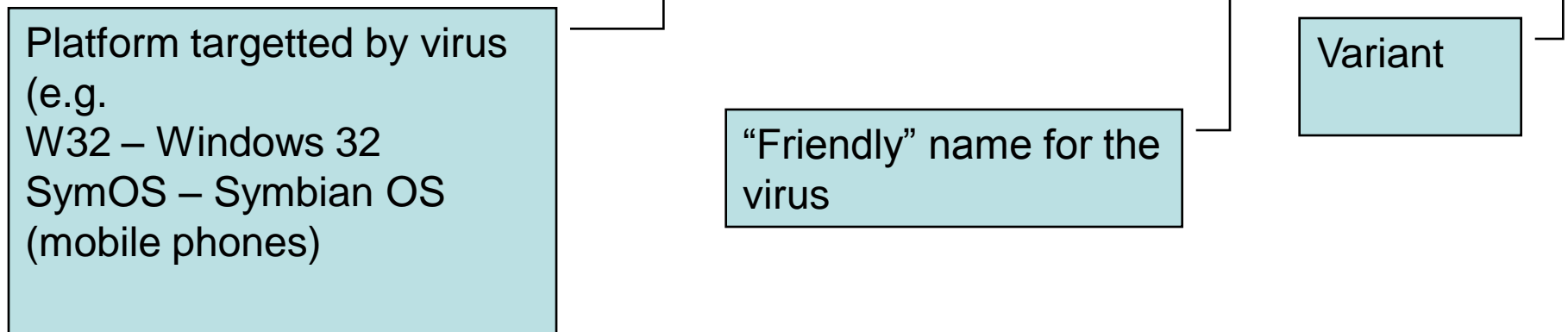
# Virus Naming – Conventions & Standards

- Problem: Multiple competing anti-virus researchers working in isolation, rapidly (we hope !) discovering and naming viruses independently
- No complete solution exists for this problem
- The first attempt was from the Computer Antivirus Research Organisation (CARO) in 1991. Problem was that CARO was a “closed-membership” organisation: non-members didn’t have access to their virus library and therefore couldn’t compare possibly-new samples to CARO’s reference collection

# Virus Naming – Conventions & Standards

- Like their biological counterparts (e.g. Retrovirus.Ebola.Zaire), viruses are now given “scientific” names

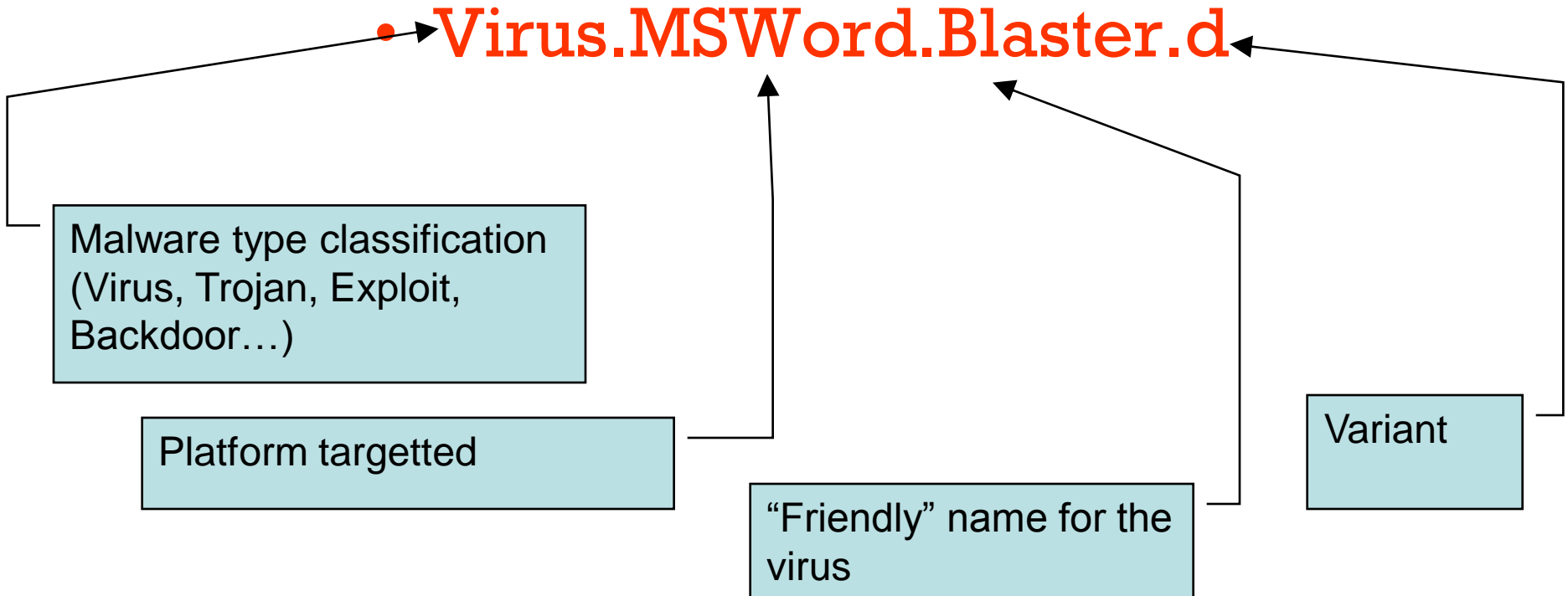
- **W32/Blaster.A**



# Virus Naming – Conventions & Standards

- Another common convention is...

• **Virus.MSWord.Blaster.d**



# Viruses – How They Spread

- The virus, once run, stays permanently resident in memory
- It intercepts appropriate system calls so that it “sees” attempts to access other system files. It “piggybacks” these accesses and spreads itself to any other suitable files which are accessed.
- The newly-infected files are then moved (e.g. by floppy, by e-mail etc.) to other systems where they are run and the cycle begins again
- Some modern viruses will actively e-mail themselves to (for example) everyone in the local Outlook address book. This greatly accelerates the rate at which they will spread (e.g. Melissa).

# Viruses – Evading Detection

- **Encryption**
  - Encryption key changes with each new generation
- **Polymorphism**
  - Encryption algorithm may change with each new generation
- **Metamorphism**
  - Source-code based. More of a threat to UNIX machines which typically have compilers/assemblers present by default
    - Register swapping
    - Constant changing
    - Subroutine rearranging
    - Null instruction insertion (NOOP and Jumps over garbage)
    - Equivalent code substitution
- **Stealth techniques**
  - Intercept calls to the OS, causing it to “lie”

# Viruses – Evading Detection

- Since 1992, *polymorphic* viruses have become widespread. The virus code is “self-modifying”, so that each generation is different to the previous one → simple byte-searches will fail.
- The virus payload is encrypted. The decryption code is randomly mutated by inserting random “null” instructions such as “nop” or “inc ax” followed immediately by “dec ax”
- Early polymorphic viruses were so effective that the “common” part could be as little as one or two bytes
- Ironically, the use of “virus-generators” to generate polymorphic viruses has limited the effectiveness of the technique: “smart” virus scanners can recognise *any* virus generated by such a generator.
- Macro-based viruses also use polymorphic techniques (random comments, random case changes, variable name changes)

# Viruses – Evading Detection

- A **stealth virus** goes further, taking steps to avoid detection. The kind of steps the virus might take include
  - Intercepting operating system calls (e.g. to the filesystem or process table) so as to “sanitise” the returned results
  - Detecting when a program is being run in “debug” mode (debug interrupt, timing) and behaving differently to frustrate analysis
  - Specifically targeting common anti-virus products so as to avoid detection by them
  - “Hiding” itself in the NTFS ADS (Alternate Data Stream)

# Viruses - Payloads

- Some viruses don't do anything other than spread. It is believed that many viruses are accidentally released before they are complete !
- Others will erase files or entire hard disks. One early virus (CIF ?) could actually destroy PC hardware by erasing the Flash BIOS. Another could damage the screen by stopping the clock which controlled the screen refresh
- Viruses which just erase files and disks are in some ways the least damaging...viruses with subtle effects or viruses which "steal" files (Melissa, SirCam etc.) are much worse

# Viruses – Detecting Them

- Three basic approaches
  - Checksum based
  - Signature-based
  - Emulation with heuristic analysis
- **Checksum-based** detection relies on the fact that the virus must modify the areas of the system (files or boot sectors) it attaches to. Given a database of checksums (CRC, MD5 etc.) of all executable files, it becomes relatively simple to detect any unauthorised modifications (TripWire)

# Viruses – Detecting Them

- **But...it is often (usually !) not practical to have a database of checksums**
  - How is the database maintained/updated when
    - New programs are installed
    - Legitimate changes (e.g. patches) are applied
    - “Unconventional” executable content (e.g. Word documents) which is routinely modified
  - How is the database (securely) stored and who gets to update it ?
  - How can you be certain that the baseline you are checking against actually is virus-free ?
  - This approach has its place on application-specific servers (e.g. database servers) where executable file changes are rare. It is rarely used on “user” machines.

# Viruses – Detecting Them

- **Signature-based** detection relies on the fact (?) that the virus must have a “fingerprint” which can be detected. Early implementations were simply a search for a sufficiently long (to avoid false alarms) sequence of bytes...this is no longer adequate
- **Emulation** techniques work by “running” the code in a virtual machine, looking for particular attributes (e.g. commonly-occurring instructions such as `CMP AX, "MZ"`)
  - This kind of technique can only really calculate a probability that a particular file contains a virus, but because of the proliferation of “engines”, it can do so with a fair degree of accuracy

# Viruses – Detecting Them

- A variation on the checksum approach is to apply a **digital signature** to the executable file. If the operating system can be configured to check for the digital signature, “unsigned” (and therefore untrusted) executable can be prevented from running
- Microsoft’s “Authenticode” technology implements this idea.
- But...
  - Who gets to decide who is trusted and who isn’t?
  - What is a sensible policy for unsigned code?

# Viruses – Detecting Them

- Writing a scanner to reliably detect viruses is challenging
- Alan Turing proved that it was mathematically that a number of problems were “undecidable” (the Halting Problem, the Busy Beaver)
- Fred Cohen used similar techniques (Turing Machines) in 1986 to prove that it was impossible to create a “universal” virus detector. A consequence of this is that virus-checkers will *always* be one step behind (reactive)

# Viruses – NetSky

- One of the most prevalent contemporary virus is NetSky.
- A mass-mailing virus, it contains its own SMTP engine and sits in the background of an infected machine quietly e-mailing itself. It searches through files on all mapped drives looking for e-mail addresses. It even tries to avoid sending itself to e-mail addresses which look like they belong to anti-virus organisations (Message Labs, McAfee, Symantec etc.)
- Presents as an attachment named to look like a document:-

```
{"document","msg","doc","talk","message","creditcard","details","attachment","me","stuff","posting","textfile","concert","information","note","bill","swimmingpool","product","topseller","ps","shower","aboutyou","nomoney","found","story","mails","website","friend","jokes","location","final","release","dinner","ranking","object","mail2","part2","disco","party","misc"}.{"txt","rtf","doc","htm"}.{"exe","scr","com","pif"}
```

# Viruses – NetSky

- Installs itself in the Windows folder as “winlogin.exe” and adds a registry entry so that it is started at bootup (the exact filename differs between variants)
- Drops copies of itself onto network shares with random filenames (e.g. “programming basics.doc.exe” ...some of the other possibilities are too crude to list !)
- Tries to delete registry entries created by the MyDoom virus (!)
- Tries to delete some of the common anti-virus products (Norton AV, Kaspersky)
- Beeps randomly on the 8<sup>th</sup> of March (its not clear why !)

# Viruses – Avoiding Them

- **Anti-virus software:** This is obvious.
- **Make sure that Windows is configured to display filenames in full and that hidden files are shown**
- **Keep up-to-date with system patches:** Most malware exploits vulnerabilities for which patches are already available !
- **Configure boot sequence to boot from hard disk:** This avoids accidentally running the boot sector from an infected floppy disk
- **Don't open attachments:** ...unless you are certain of their origin
- **Disable the Preview Pane:** Some viruses exploit vulnerabilities so that they run when they are simply displayed (e.g. JPEG vulnerability)

