

Morris Worm

Out of scope to explain about insects coming out of keyboards today,
sorry.

WORMS IN GENERAL

Definition of a worm

“A worm is a program that can run independently, will consume the resources of its host from within in order to maintain itself, and can propagate a complete working version of itself on to other machines.”

<http://tools.ietf.org/html/rfc1135>

How do worms work?

- A "pure" worm does not require human interaction.
 - Selfreplication
 - Scans for next victims
 - Copies itself to victims
 - Automated install and execution
 - Uses vulnerabilities

Mechanisms of the worm

1. Initial system infection
2. Scanning for other victims
 - And propagating
3. Maybe delivering a payload

Initial infection

- Exploits Operating System vulnerabilities
 - Very complex → lots of bugs!
 - The recent MS Server Service vulnerability
 - Vulnerabilities introduced by users
 - Weak passwords (or perhaps no password!)
 - Bad operating system settings

Propagation

- Uses internet connection to spread.
 - Which application is not using the internet these days?
 - E.g. Outlook's, email preview pane
 - Old insecure software

Scanning techniques

- Network scanning techniques:
 - Random
 - Nonrandom:
 - Localized
 - Hitlist
 - Importance
 - Sequential
 - Topological
 - ...and many more

Example

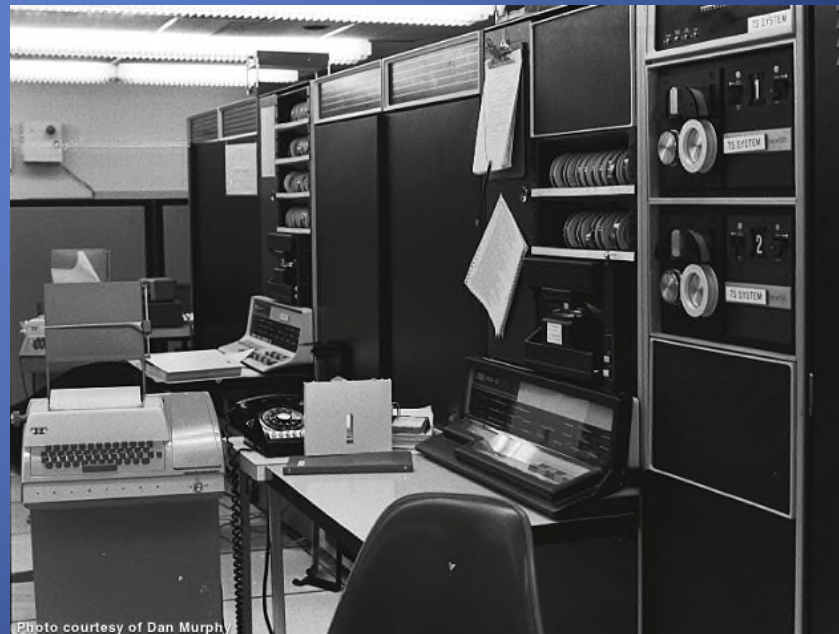
Proof of concept sql injection worm

- Scan a website and look for forms in the html
- Test each form for trivial SQL injection vulnerabilities
- If vulnerable, run following checks:
 - is the back end Microsoft SQL Server? (~33% chance)
 - are stored procedures executable?
 - Is the xp_cmdshell extended stored procedure executable?
- If above checks cleared, do:
 - Use xp_cmdshell to upload copy of itself.
 - Uses debug.exe to compile a normal ascii file, and execute it

HISTORY

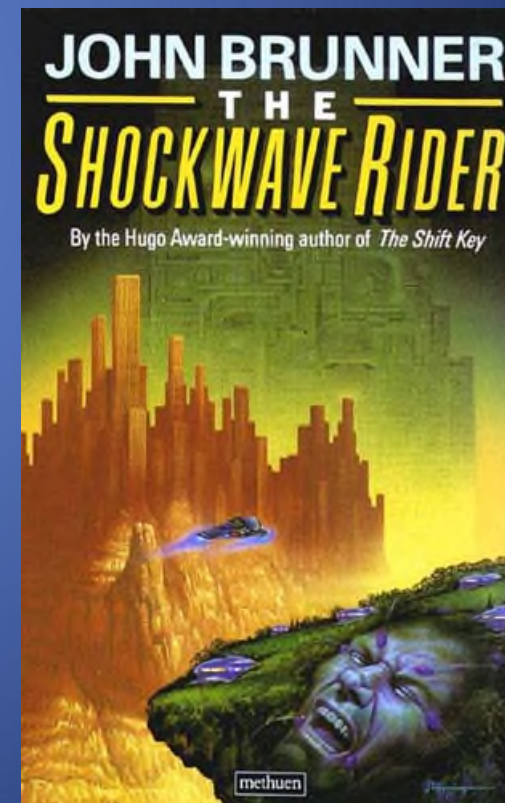
Before Morris

- 1971 – Creeper worm on the ARPA-Net
 - Written by Bob Thomas
 - Wrote the message I'm Creeper, catch me if you can on screen
 - No other effect



Before Morris

- Early 1980's – Tapeworm at Xerox Research Center
 - Written by John Shoch
 - Designed to do calculations on idle machines
 - Program got corrupted and crashed the infected computers



Before Morris

- Creeper worm on the ARPA-Net
- Tapeworm at Xerox Research Center
- Until now, none was made for malicious use

Morris

- 1988 – Morris Worm
 - Written by Robert Tappan Morris
 - About 6000 UNIX machines was infected
 - Est. damage: \$10 – 100 million



After Morries

- 1989 –WANK worm
 - Believed to be written by anti-nuclear activists
 - Spread through the DECnet protocol
 - It hid files from users, made new accounts on the system and broadcasted vulnerabilities found on the system

1999

- Melissa worm
 - Spread through the addressbook in Outlook
 - It would attach found word documents and send to random people
 - It would insert quotes from Simpsons into word documents
 - Est. damage: \$1.1 billion

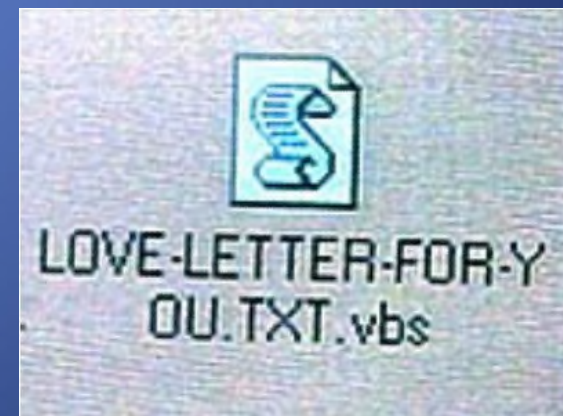


2000

- ILOVEYOU

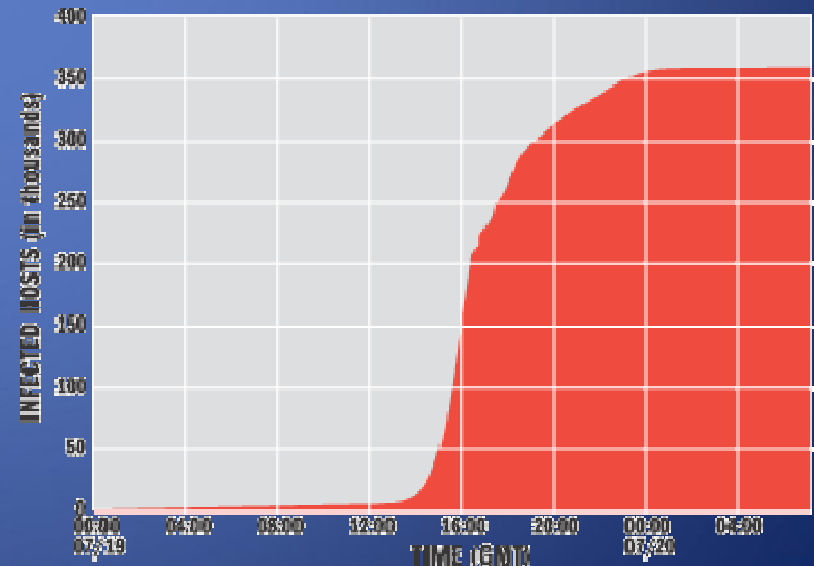
- Spread through Outlook as Melissa
- It overwrote system files with a copy of itself
- Marked mp3 files as hidden
- Installed a Trojan Horse that would steal usernames and passwords

- Est. damage: \$8.7 billion



2001

- Code Red
 - Infected more than 360.000 in 14 hours
 - Defaced the website on the server
 - DoS attack against White House IPs
 - Est. damage: \$2.6 billion



2003

- SQL Slammer Worm
 - Used a buffer overflow in MSSQL server database
 - The first Warhole worm
 - Est. damage: 36.1 billion
- Blaster
 - Used a buffer overflow in Microsofts DCOM RPC
 - Infected w2k and XP.
 - Programmed to make a DoS against windowsupdate.com
 - Est. damage: \$1.2 billion

2004

- MyDoom
 - Its attachment would resend itself to various mails
 - Install a backdoor on the infected system
 - Launch a DoS against The SCO Group
 - Est. damage: \$38.5 billion



Thank you for registering at WORLDXXXPASS.COM

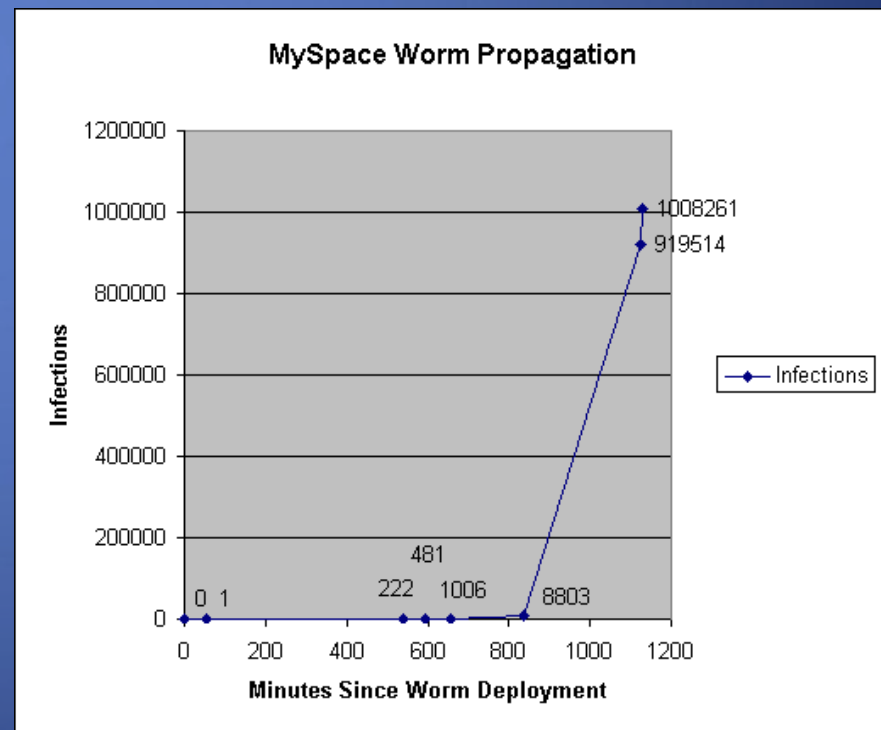
All your payment info, login and password you can find in the attachment file.

It's a real good choice to go to WORLDXXXPASS.COM



2005

- Samy
 - XSS worm that spread through Myspace
 - Altering user profiles and add them as friends to Samy Kamkar

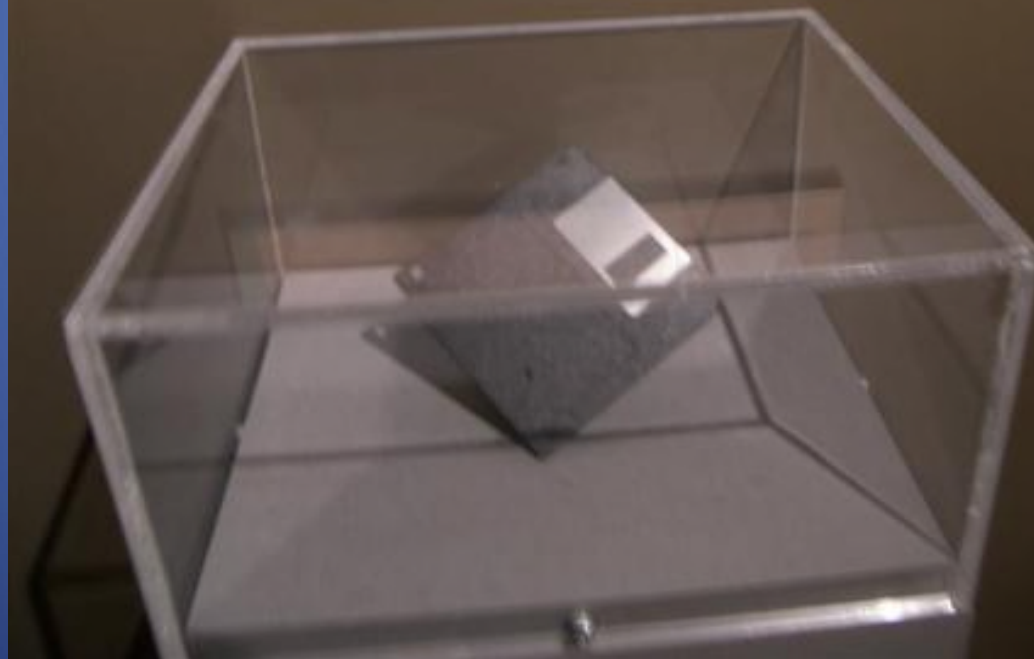


The Morris Internet Worm source code

This disk contains the complete source code of the Morris Internet worm program. This tiny, 99-line program brought large pieces of the Internet to a standstill on November 2nd, 1988.

The worm was the first of many intrusive programs that use the Internet to spread.

The Computer History Museum



TECHNICAL DETAILS

Outline

- How did the Morris Worm work?
- What were the vulnerabilities and how were they used?

High level structure

- Initialization – **main**
- Setup – **doit**
- States in the primary loop
 1. accumulate hosts and hashed passwords
 2. break passwords by guessing
 3. break passwords by own dictionary
 4. break passwords by found dictionary
 5. continuously try to infect

Initialization: Hiding

- Camouflage and covering tracks
 1. Change argv to look like shell
 2. Limit core dump
 3. Load object files into memory
 4. Remove traces
 5. Kill parent process

Initialization: Snooping

- Look around for possible network connections
 1. `loctl`
 2. `Netstat`
 3. System files

Doit

- Setup – seeding random number generator
- Infection attempts
 1. Check for service
 2. Exploit vulnerability
- Check for other worms
- Report to server at Berkeley

Primary loop / state machine

1. Accumulate hosts and hashed passwords
2. Break passwords by constructing guesses
3. Break passwords by own dictionary
4. Break passwords by found dictionary
5. Loop while trying to infect

The grappling hook

- Three ways to enter another system:
 1. Sendmail exploit
 2. Fingerd exploit
 3. Rsh/rexec vulnerability

And now for the exploits...

Fingerd attack: What is finger?

- Finger is a utility allowing a user to obtain information on other users.
- Uses finger protocol talking to a finger server.
- Not widely used anymore since shift in privacy views.

Fingerd attack: What was the vulnerability?

- The finger daemon used to contain calls to a standard library function - *gets*
- *gets* was vulnerable to a buffer overflow
- A 512 byte read buffer on the stack

– Fingerd attack:

– How was the vulnerability used?

- Overflow the message buffer
- Calculate offset to hit return address
- Replace return address with malicious code
- [bad stack drawing would be nice here]

– Fingerd attack:
– How did the worm enter?

- 1. /bin/sh command is executed instead of /usr/ucb/finger
- 2. stdin and stdout are now connected to network sockets
- 3. Bootstrap program is sent as text and written to a file
- 4. The bootstrap file is compiled to an executable
- 5. Bootstrap program is executed and fetches the worm files
- 6. The worm starts
- [nice network diagram to with pipes redirected to explain here]

Sendmail attack:

– What is sendmail?

- A daemon program responsible for mail routing
- Very commonly used program but very complex
- Listens on port 25 to see if SMTP messages arrive
- The versions used were normally compiled with a DEBUG option

– Sendmail attack:

– How was the vulnerability used?

- DEBUG option was used to test and develop the program
- Complexity and difficulty of configuration meant most versions used had the DEBUG option compiled in
- During debug it is allowed to pipe mail messages to another program
- instead of to a real mailbox

– Sendmail attack:

– How did the worm enter?

- 1. Call up Sendmail on port 25 using the DEBUG command
- 2. Specify that a shell should receive the “message” instead of an
 - actual mail address
- 3. Send the bootstrap program as text to the shell and pipe it into a
 - file
- 4. Compile the bootstrap program to an executable
- 5. Bootstrap program is executed and fetches the worm files
- 6. The worm starts

– Rsh/Rexec attack

- Not really an exploit/vulnerability
- Access to an account allowed access to other commonly used
- accounts for the particular user

Mommy, there are monsters under the bed!

SHIFTING FOCUS

Shifting Focus

- Old Skool
 - All about Respect
- Money
 - The birth of an industry
- Money-squared
 - The industrial Revolution, specialization
- The Future
 - Warfare and Propaganda

Cost - Benefit

- Pros
 - Make money
 - Whole world at your fingertips
 - Possible to hide
 - Not getting shot at
- Cons
 - Can be tracked
 - Fines
 - Prison sentence
 - Big crimes possible – big resources mobilised to catch you

The Old Skool



- Morris Worm
- Get Respect
- Get Fame
- Get Hired by FBI
- Internet one big 'hugge' community

Money

- All about a revenue stream
- An emerging market
- Hackers learning
- Security folk learning
- Policing and punishment lagging behind.....

Money²

- Industrial revolution
 - Factories appear (e.g. October 31st)
 - Mechanization of attacks!
 - Change in structure of Cyber-crime
 - Now they're Organised
 - Cyber-crime gangs get specialised
 - Specialists & recruitment
 - Attack programs are updated regularly
 - As shown by in 'storm botnet'
 - Multiple methods of attack

Money² - Weaponization

- 'Drive buy' infections, an army of zombies
- Removing each others worms - competition
- Catching organized 'cyber' gangs is like catching 'real world' organized crime gangs

Money² -The Gangs

- No longer lone pizza loving geek in their bed room
- Large Specialized gangs
- It's not the wild, wild west - it's the mob...

The Eternal Chase

- Cyber-crime units set up
- New laws enacted
- Cross border cooperation
- Harsher penalties
- Hide behind proxies
- Hide behind Fluxing
- Hide behind double fluxing
- Hide out in uncooperative countries

The Future

- Political Agenda
 - Warfare, e.g. Russia and Georgia
 - Propaganda
 - Spying on your own population – '1984'
- Staying under the radar
 - By filling the 'ether' with lots of noise from automated 'factory bots' the REAL crime goes undetected....
- Others
 - To be discovered...

Forewarned is Forearmed

- NATO established 'Cyber Warfare' base
- Cyber Warfare as bad as missiles?
- White House invests 6 Billion USD
 - DKK 35,021,383,276

Personal Reflection

- Worm uses and known exploit
- Trojan/Virus uses social engineering
- People are the weakest link and always will be...
- Secure the weakest link [VM]
- Low-hanging fruit is important – 'the long tail effect'[2]

The Final Word

- "The root cause of the issue is that the bad guys are better funded than we are," said Simmonds. "They have research and development programmes, they are putting people through university, they are calculating return on investment and they have better quality assurance. By comparison, the legitimate security industry is under-funded, under-resourced and constantly on the back foot." [6]

Bibliography

1. <http://www.root777.com/computer-security/important-computer-security-terms-and-terminology/>
2. http://en.wikipedia.org/wiki/The_Long_Tail
3. the changing face of cybercrime - the new internet threats create challenges to law enforcement, Terrence Berg, June 2007, Michigan Bar Journal
4. 'NATO says cyber warfare poses as great a threat as a missile attack' ,March 6 2008, <http://www.guardian.co.uk/technology/2008/mar/06/hitechcrime.uksecurity>
5. Finextra: Serious Organised Crime Agency warns on cyber gangs, 9 June 2008, <http://www.finextra.com/fullstory.asp?id=18560> {a news site for financial technology professionals.}
6. <http://cybercrimeupdates.blogspot.com/2008/06/organised-e-crime-targets-students-for.html>

LESSONS LEARNED

Lessons Learned

- Monoculture is rampant in IT
- Agriculture strives for monoculture as well
 - Ever heard of a fruit called Gros Michel?
- But monoculture is also good for software development
- Sandbox / mistrust programs. Case: RPC exploit
 - Windows monoculture
 - But why was the flaw less of a problem on Vista?

Current Protection

- Intrusion Detection
 - Alterations in system files
 - Fingerprint matches
 - Recognizable structure
- Behavioral Heuristics
 - Program agnostic
 - Watches for suspicious activity
 - Reactive, not proactive
- Firewalls
 - Simple - by IP and port
 - Intelligent - as above + packet contents
- Intelligent firewalls better or worse?

Future of worms

- Social
 - Improved masquerade
- Engineering
 - Continues to improve
 - Particularly stealth - use common, 'trusted', protocols such as HTTP, RSS
 - Increasingly modular and specialized
- Propagation
 - Cell phones
 - Other wireless devices
 - Ubiquitous computing: worms in your clothes
- Purpose

Future in defense

- Pure worms are now comparatively rare
- Intrusion Prevention at the network level
- Speculative defense
- Reaction time critical

- Far out?
 - Secure, authenticated networks?
 - Anti-malware worms?