

Combating Malicious Software

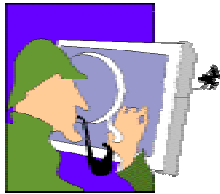
A “Defense-in-Depth” Approach

Presented By:
Ed Rodriguez

ACSAC '01
December 13, 2001

Booz | Allen | Hamilton
delivering results that endure

© 2001, Booz Allen & Hamilton Inc.



Outline

- Overview
- A Look Backward
 - Key Trends
 - Historic Treatment of Malware
- A Look Forward
 - Layered Defense
 - Future Outlook
- Final Words

Overview

- What is Malicious Software (Malware)?

Malware is any software added, changed, or removed from a software system in order to intentionally cause harm or subvert the intended function of the system.



- Back Orifice
- SubSeven



- Morris Worm
- Code Red



- Melissa Macro Virus
- ANNA Script Virus

Page 3

Overview

- What is not Malware?

- Easter Eggs
 - Hidden programs within COTS
 - Not intended to harm, only bring notoriety
- Software Defects
 - While not malware, may have exact same effects!
 - Somebody just has to find the security vulnerability
 - Buffer overflows to start, reverse engineering if serious



Develop By...
Design By...



Hello?XX...XXX



I'm yours to command

Page 4

Sun Tzu said...

If you know the enemy and know yourself,
you need not fear the result of a hundred battles

Briefing Approach

- Let's Look at Some Key Trends Based on Recent Events
- Let's Then Look How Malware has been Addressed in the Past
- Lastly Let's Look an Applicable Defense-in-Depth Strategy

Key Trends

- AV Software Vendors Continue to use Signatures
 - Perpetuates the “reactive” nature of AV products
 - Scalability considerations
- Continued success of Malware “Mutations”
 - Mutations are minor variants of existing malware that have been modified in a minor way
 - Continue to exploit known vulnerabilities
 - Require much less energy, creativeness, and technical skill than original author

Page 7

Key Trends

- Emergence of Script Viruses and "Trojan Horses"
 - VBA is easy to learn & widely used
 - SubSeven and Back Orifice are widely available
- Difficulty in Detecting "Trojan Horses"
 - It is a lot like automatically finding “bugs” in software ... and that is a *hard* problem!
- Continued “Social Engineering” Exploitation
 - For example, Sircam will not generate the same Subject Line all the time as it propagates itself (in contrast to the ILOVEYOU virus)
- Development of COTS SW by Foreign Labor
 - Skilled labor available overseas
 - Threat...yes Real Problem ??

Page 8

Historic View of Malware

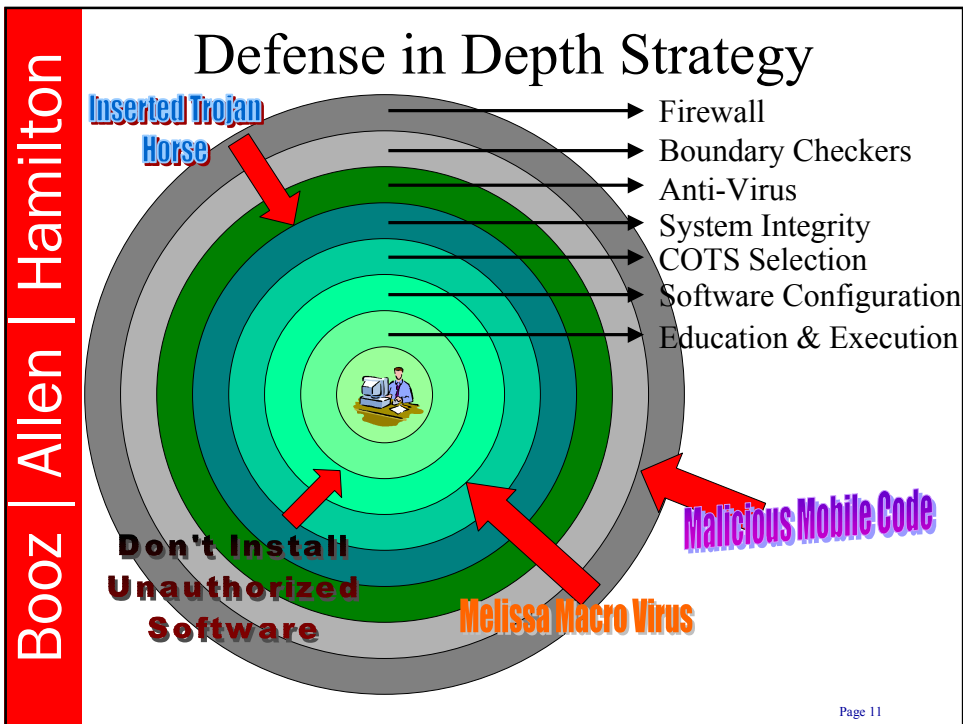
- It **was** about viruses!!
- Client-server model relied on either
 - Fat clients with much local functionality
 - Thin clients that relied on powerful servers
- User educated to
 - Update virus definition files
 - Not to download files off public sites
 - Accept attachments only from people they know

Page 9

Today's Situation

- It is **not just** about viruses!!
 - Trojan horses
 - Self executing emails from “friends”
 - Self propagating malware using “hacker” techniques
- Client-Server model has evolved
 - Mobile code use is increasing
 - While it supports powerful computing constructs, can be dangerous
- Rely **less on the user** to protect himself

Page 10



- Booz | Allen | Hamilton**
- ## Boundary Checker
- Software application typically used in conjunction with a firewall that allows for a granular security policy against mobile code
 - Relatively new class of products
 - Two Common Criteria Protection Profiles developed
 - Typical products include Finjin ..., eGate
- Page 12

Anti-Virus Software

- Effective against known and addressed viruses and other forms of malicious software
- However, you can only detect & protect against what you know about
- Oh... don't forget to update your signature database
- Inherently a non-real-time reactive approach despite attempts at proactive heuristic mechanisms

Software Configuration

- One of the most overlooked and neglected aspect of using COTS software components
- Can be viewed from two perspectives:
 - Initial: Traditionally software packages ship and are installed with default settings that are highly functional but have weak security properties
 - Operational: Changed based on operational needs and more importantly updated based vendor released service packs or patches
- Single most effective means to avoid being vulnerable to new malware

User Education

- Safety in education
- Traditional focus on informing users to avoid “unsafe computing practices”
 - This is generally ineffective and assumes that users know what these unsafe practices are
 - At odds with the axiom of making security as user transparent as possible
- Let’s not forget the most important soldier in the battle against malware...

Page 15

User Education

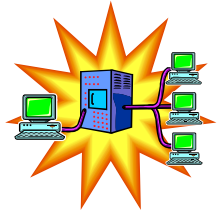


System/Security Administrator

- Tracks published COTS vulnerabilities and/or patches
- Assesses applicability to system configuration
- Ensure interoperability via regression testing
- Monitor current malware threats
- Update system/software parameters as needed
- Communicate with end users as needed

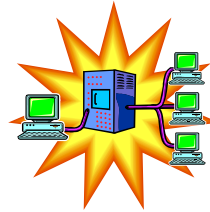
Page 16

System Integrity



Baseline System

?



Operational System

Question:

Has the operational system changed in an unauthorized manner?

- Manually inserted malware?
- Malware that infiltrated the other defenses?

Answer:

Comparison of key file hashes values between the baseline and operational systems. For example, Tripwire

COTS Analysis & Selection

Whether it is the intentional manual insertion of malicious software or the inadvertent creation of software “bugs” that create a security vulnerabilities,

does it really matter??



- Foreign agent
- Disgruntled employee
- Criminal type



- Poor quality software
- Widely used software
- Target of hacker community

COTS Analysis & Selection

- Consider the use of products that are not the focus of the hacking community
 - Beware of over reliance of this “security through obscurity” strategy
 - Recommendation by Gartner to not use Microsoft® IIS web server
- Development Company (Reputation and Location) might need to be factor in use of product
 - Would China have Russia develop its Air Defense system?

Page 19

Future Outlook

- “If it can happen it will happen”
- Recent Predictable Events
 - Continued Prevalence of Macro & Script Viruses
 - Continued Effective Use of Malware Toolkits
 - Emergence of the Worm
 - Detection of Malware Continues to be Difficult
- Future Predictable Events
 - Personal Digital Assistants (PDAs) Being Targeted
 - Extensible Markup Language (XML) Based Malware
 - Growing Prevalence of the Script & Hacker-Based Malware
 - Increasingly Sophisticated Social Engineering Techniques

Page 20

Key Observations

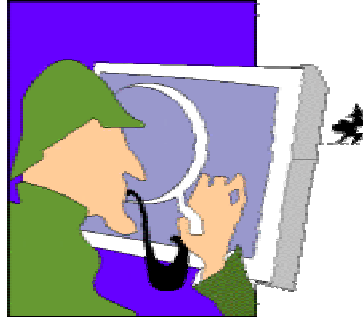
- It's About the People
 - The “insider threat” individual
 - The “socially engineered” individual
 - The “system administrator” individual
- Despite Industry Diligence Anti-Virus Software Has Diminishing Effectiveness
- Exploitable Vulnerabilities Are Everywhere
- New Technologies Introduce New Functionality Along With New Risks

Page 21

Final Thoughts

- Aggressively Pursue a Balanced Strategy Against Malicious Software
- Aggressively Pursue Policy to Ensure Consistent and Effective System Administration
- Development Guidance and/or Standards for Malicious Software Countermeasures

Page 22



Note: Information Assurance Technical Analysis Center (IATAC) scheduled to release a State-of-the-Art Report (SOAR) on Malicious Software in Early 2002

<http://iac.dtic.mil/iatac/>