Intro to Microsoft Windows
OS Security
Microsoft Windows

• Major Versions
  – 3.x - 1990 - 1993
  – Windows 9.x
  – NT
    • 3.x - 1993
    • 4 - 1996
    • 5 (Windows 2000) - 2000
    • 5.1 (Windows XP) - 2001
    • 5.5 (Windows Server 2003)
Microsoft's 10 Immutable Laws of Security

Law #1: If a bad guy can persuade you to run his program on your computer, it's not your computer anymore.

Law #2: If a bad guy can alter the operating system on your computer, it's not your computer anymore.

Law #3: If a bad guy has unrestricted physical access to your computer, it's not your computer anymore.

Law #4: If you allow a bad guy to upload programs to your web site, it's not your web site any more.

Law #5: Weak passwords trump strong security.
Microsoft's 10 Immutable Laws of Security

Law #6: A machine is only as secure as the administrator is trustworthy.

Law #7: Encrypted data is only as secure as the decryption key.

Law #8: An out of date virus scanner is only marginally better than no virus scanner at all.

Law #9: Absolute anonymity isn't practical, in real life or on the web.

Law #10: Technology is not a panacea.
Windows Bootstrap Process

- **Hardware**
  - IA86 Power On Self Test (POST)
  - BIOS / CMOS
  - Boot Sector
    - MBR - fs type, physical & logical layout, bootstrap program
- OS kernel
  - ntdlr, boot.ini, ntdetect.com, ntoskrnl.exe, hal.dll
  - io.sys, msdos.sys, win.exe (DOS, 3x, 9x)
Windows Bootstrap -2

- `config.sys`, `autoexec.bat`, `system.ini`, `win.ini`, registry
- Drivers/Services
  - `exe`, `vxd`, `dll`
- Login
  - authentication, settings & preferences
- User startup programs
Windows Architecture

- API
- Registry
- Filesystem
- Discretionary Access Control
- Networking
Windows API

- Interprocess communication specification(s)
- API (function) calls
- vs Unix domain sockets, pipes, libc
- Available via PE export table:

exports table:
Name: KERNEL32.dll
Characteristics: 00000000
TimeDateStamp: 3B7DDFD8 -> Fri Aug 17 23:24:08 2001
Version: 0.00
Ordinal base: 00000001
# of functions: 000003A0
# of Names: 000003A0

Entry Pt Ordn  Name
00012ADA     1  ActivateActCtx
000082C2     2  AddAtomA

...remainder of exports omitted
Windows NT Architecture

- Two Distinct “Modes”
- User NEVER directly accesses hardware
- Win32 is Grand Central
- NT Executive only Talks to Win32 subsystem
- HAL makes all Hardware Calls
Windows Registry

- Hierarchical Database
- Settings & options
  - classes & associations
  - hidden elements
  - device settings
  - network options
  - passwords
  - startup processes
    - services, programs, libraries
9x Registry

• Win 9x
  – Registry hives
    • c:\windows\system.dat
    • c:\windows\user.dat
    • c:\windows\profiles\<username>\user.dat
NT Registry

- **Win NT hive files**
  - `%SystemRoot%\System32\Config`
  - `c:\winnt\profiles\<user>\ntuser.dat`
  - `c:\Documents and Settings\<user>\ntuser.dat`
  - system, software, default, security, SAM
  - *May be copied in c:\winnt\system32\repair!*
Registry
Regedit

- regedit <filename>

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Dhcp]
"Type"=dword:00000020
"Start"=dword:00000002
"ErrorControl"=dword:00000001
"ImagePath"=hex(2):
    25,00,53,00,79,00,73,00,74,00,65,00,6d,00,52,00,6f,00,6f,00,\
    74,00,25,00,5c,00,53,00,79,00,73,00,74,00,65,00,6d,00,33,00,32,00,5c,00,73,\n    00,65,00,72,00,76,00,69,00,63,00,65,00,73,00,2e,00,65,00,78,00,65,00,00,00
"DisplayName"="DHCP Client"
Registry

- Subtrees, keys, subkeys, entry name & data
- Local or remote administration
- BACKUP, BACKUP, BACKUP
- permissions
  - XP regedit, 2k regedt32
  - AEDebug, McAffee
Registry

• additional info
  – http://www.winguides.com/registry/
File Systems - FAT

- **fat16** - Win3x, Win9x, WinNT
  - 16 bit entries, max 2GB
- **fat32** - Win95 OSR2, 98, NT5+
  - 28 bit entries, max ~ 2TB
- File attributes - archive, system, hidden, read-only, volume label, directory
- Directory entries are used to logically order files and other directories
File Systems - NTFS

- NTFS 4/5
  - max $2^{64}$ Bytes
  - dacls, efs (service)
  - 16 bit Unicode
  - alternate data streams (nonresident data)
  - metadata
    - $MFT$ - Master File Table (FAT decendant)
    - $BADCLUS$
    - $SECURITY_DESCRIPTOR$
The Five NT Security Modules

- Local Security Authority - LSA
- Security Accounts Manager - SAM
- Security Reference Monitor - SRM
- Security Identifier - SID
- Access Control List - ACL
Access Tokens

- Created by the Local Security Authority (LSA)
  - Stays with user's session until log off
  - Token attaches to each process that the user creates.

- Token consists of
  - User’s SID
  - Group SID
  - User privileges
  - Owner assigned to new objects
  - Primary Group SID
  - Default ACL assigned to new objects

- Stored by Security Accounts Manager (SAM)
Access Control Lists

- Contains a list of users and groups assigned permission to access and/or operate on an object
  - Composed of Access Control Entries (ACE)
  - ACE defines the permissions given to each user or group for an object
Access Control Lists

- Each object contains a security descriptor
  - Security Identifier of the object owner
  - Discretionary ACL for access permissions
  - System ACL (SACL) which is used for auditing
  - Group security identifier.
Determining Object Access

Tokens (SIDs and GIDs)
- Security ID: Unique ID (i.e., S-1-5-21-146464331-...)
- Group IDs:
  - Employees (Unique SID)
  - Scientists (Unique SID)
  - EVERYONE (Unique SID)
  - LOCAL (Unique SID)
- Other Information:
  - Tokens (SIDs and GIDs)

Access Control Lists
- Owner Security ID: Unique Identifier
- Group SIDs: POSIX subsystem (only)
- Access Control List
- Access Control Entries
- System ACL: Auditing Messaging
- Object Type: Container/NonContainer

Comparison Process:
1. Compare Token SID’s to Access Control Listing?
   - Yes → Game Over
   - No → Process Access Denies
2. Process Access Denies
   - Yes → Game Over
   - No → Process Access Allows
3. Process Access Allows
   - Yes → Game Over
   - No → Allow Object Access
4. Allow Object Access
   - Yes
   - No → G’Bye
File Permissions
User Accounts

• Provide segmentation of resources based on need
• Provides accountability
  – supports auditing
• Superuser account - Administrator
• Local System
• Guest
Authentication

- Authentication is the process whereby you convince the authority to grant access to some resource by uniquely presenting one or more of:
  - Something you know
    - password, pin, mother's maiden name
  - Something you have
    - Magnetic card, identification card, key
  - Something you are
    - Retina scan, voiceprint, DNA, fingerprint, a friendly voice
Authentication

• To validate the credentials you present, an authentication system must compare them to a trusted store.

• Passwords should be:
  – Encrypted well
  – Kept confidential
  – Selected in accordance with reasonable policy
Password Storage

• Passwords stored as cryptographic hashes in SAM or Active Directory
  – SAM passwords hashed with:
    • Lan Manager Hash - Weak!
    • NTLM/NTLMv2 - better

• Other applications notorious for poor storage:
  – ws_ftp.ini, eudora.ini, username.pwl

• Active Directory authentication via Kerberos
LanMan Hash

• 14 Characters or less
  – foobar

• Split into two 7-byte fields, uppercase & padded with zeros
  – foobar0 0000000

• Create an 8-byte odd-party DES key from each, used to encrypt (0x4B47532140232425 encrypted with a key of all 1's).
  – d85774cf671a9947aad3b435b51404ee - foobar
  – aad3b435b51404eeaad3b435b51404ee - <null>
Steps in the Logon Process

- Two Logon Types
  - Local Machine Logon
  - Remote Domain Logon
- Logon consists of interaction between
  - User mode subsystems
  - Kernel mode subsystem
- Every Level of system involved
Steps in the Local Logon Process

1. User types name, password and domain

2. LSA creates an “MSV1_0 Authentication” package from #1
   a. MSV1_0 encrypts password using one way function (OWF)

   a. If no match, access denied, and user informed

Security Access Token

Win32 Subsystem

New Process

Security Accounts Database
Steps in the Local Logon Process

4. LSA creates Access Token (aka SID)
5. Access token moves to Application Layer
6. Win32 creates new user process
7. Access Token attached to user process
Steps in the Remote Logon Process

1. Netlogon Service establishes secure channel (NT Challenge-Response)
   a. Password moves over secure channel to server. **Domain Controller** SAM determines match.
   b. If no match, access denied, and user informed

2. **Domain Controller** LSA creates Access Token (aka SID)
   a. Domain Controller moves Access token to Client Application Layer
Groups

• Allow for simplified access control
• Many default groups preinstalled
  – Authenticated Users
  – Administrators
  – Everyone
• Create custom groups as needed
• http://windows.stanford.edu/docs/defaultgroups.htm
Defaults

- Windows 2000 Server - IIS enabled
- Everyone Group has read access to system
- SQL Server - null SA Password
- Profile directories on system volume
- Windows Networking Services
  - NetBIOS over TCP/IP
  - Null Sessions
Domains

- Networked computers share a common trust
- Domain Controller is the central security arbiter of trust within a "domain"
- Shares - SMB/CIFS
  - Server Message Block / Common Internet File System
By default, full control to "everyone" Sharing relies on the "server" service
Gathering information

- `c:\net use \target_ip\ipc$ "" /user:""`
Hotfixes, Rollups & Service Packs

- Vulnerabilities are disclosed only when fixed
- Vulnerabilities are disclosed when fixed
- Vulnerabilities exist due to lack of proper planning & understanding of more and more complex relationships
- Vulnerable software is most often targeted AFTER a patch is available
Hotfixes, Rollups & Service Packs

- **Baseline Security Analyzer**

- **Microsoft Security Bulletins**

- **NT Bugtraq**
  - [http://www.ntbugtraq.com](http://www.ntbugtraq.com)

- **Vulnerability Development Discussion Forum**
  - [http://online.securityfocus.com/archive/82](http://online.securityfocus.com/archive/82)
MS SQL Server Slammer

• The result of
  – poor planning & programming
  – default installation & lack of patching