

# Improving Resiliency

[steriley@microsoft.com](mailto:steriley@microsoft.com)

**Steve**

**Riley**



Microsoft®

**Windows<sup>®</sup> xp**

**Service Pack**

# What is SP2?

- All the usual fixes of course
- New security technologies
  - Network protection
  - Memory protection
  - Safer e-mail handling
  - More secure browsing
  - Improved computer maintenance

# Security goals

**Increase the security resiliency  
of Windows XP**

**Make attackers work harder**

**Reduce damage of worms and viruses  
even if updates are not installed**



# Scope

- Information today reflects beta version as of 12 December 2003
- Will include info today for developers
  - RPC
  - DCOM
  - ICF
  - NX (execution protection)

# Defense in depth

## Networks

- Routers
- Firewalls
- VLANs
- Subnetting

## Hosts

- IPsec
- Access control lists

## Applications and data

- Authentication
- Authorization
- Rights management
- Access control lists
- Execution partitions

## Users

- Uhh...

# Network protection

- Internet Connection Firewall
- RPC interface restrictions
- DCOM security enhancements

# ICF—new features

- On by default
- Boot time security
- Global and per-interface configurations
- Local subnet restriction
- Command-line support
- Shielded operational mode
- ICF permissions list
- Multiple profiles
- RPC support

## *Internet Connection Firewall*

# On by default

**What is it?**

- ICF on by default on all interfaces
- New installations and upgrades
- Enabled when new interfaces are added

**Why do it?**

- Configuring ICF proved to be too difficult
- Default configuration provides good protection against worms (eg., Blaster)

**What's different?**

- Certain applications might require special ICF settings

**How do I fix it?**

- Developer documentation ICF API

## *Internet Connection Firewall*

# Boot time security

**What is it?**

- New static filtering policy at boot time
- Permits DNS, DHCP, Netlogon
- ICF policy applied after logon

**Why do it?**

- Closes hole that existed after boot but before policy application

**What's different?**

- Nothing

**How do I fix it?**

- No need

## *Internet Connection Firewall*

# Global configuration

**What is it?**

- Configuration changes apply to all interfaces (including new interfaces)
- Per-interface configuration still possible

**Why do it?**

- Easier to synchronize policy across multiple interfaces
- New interfaces get a policy when created

**What's different?**

- Global plus local configs

**How do I fix it?**

- Developer documentation ICF API

## *Internet Connection Firewall*

# Local subnet restriction

**What is it?**

- Can restrict port opening to local subnet address range
- Is the default for file sharing ports

**Why do it?**

- More granularity—allows local subnet communication but not to/from Internet

**What's different?**

- Enabling “file and printer sharing” applies restriction to 137/udp, 138/udp, 139/tcp, 445/udp, 445/tcp

**How do I fix it?**

- Developer documentation ICF API if application can't work with restriction

# Command-line support

**What is it?**

- Add ICF configuration to NETSH utility
- Default state, open ports, global or per-interface, subnet restrictions, logging options, ICMP handling, application permissions

**Why do it?**

- Best method for logon scripts and group policy

**What's different?**

- Nothing—new functionality

**How do I fix it?**

- No need

# Shielded operational mode

**What is it?**

- A UI button that closes all static openings for inbound traffic

**Why do it?**

- Easy way for user to stop all incoming unsolicited traffic
- Useful if a virus or worm is suspected

**What's different?**

- When enabled, computer won't respond to incoming requests
- API calls to create static openings will be stored but executed only when operational mode is returned to normal

**How do I fix it?**

- Restore normal operational mode

## *Internet Connection Firewall*

# Permissions list

**What is it?**

- Applications that need to opening listening ports

**Why do it?**

- Allows application to run in lower security context
- Only local administrator can add to list
- Ports remain open only while application is running

**What's different?**

- Any app that listens must be on the list

**How do I fix it?**

- No need

# Multiple profiles

**What is it?**

- Location-based profiles: one when connected to a corporate network, another when connected to the Internet

**Why do it?**

- Can have a more relaxed profile when corp-attached and a more restrictive profile when traveling

**What's different?**

- Computer must be domain-joined
- Listening applications might need to be on both profiles

**How do I fix it?**

- No need

## *Internet Connection Firewall*

# RPC support

**What is it?**

- ICF watches as RPC apps register ports
- Allows incoming requests only if service is running as Local System, Network Service, or Local Service

**Why do it?**

- Can control which RPC services are exposed to the network
- Better than granting permissions to SVCHOST.EXE

**What's different?**

- Must do this for RPC—ICF blocks all RPC by default

**How do I fix it?**

- Developer documentation ICF API to automate

# ICF—changes

- **Enhanced multicast and broadcast support**
- **Updated NETSH helper for IPv6 ICF**
- **Updated user interface**
- **New group policy support**

## *Internet Connection Firewall*

# Enhanced m'cast and b'cast

**What is it?**

- If ICF receives incoming m'cast or b'cast traffic, it allows for three seconds a response from any source address to the originating port

**Why do it?**

- Allows responses without adding client applications to permissions lists

**What's different?**

- Incoming b'cast and m'cast traffic now passes through ICF without manual configuration

**How do I fix it?**

- No need

## *Internet Connection Firewall*

# Updated user interface

**What  
is it?**

- New dialogs and settings
- Final UI still under design

**Why  
do it?**

- Necessary for new configuration options

**What's  
different?**

- Now a control panel applet

**How do  
I fix it?**

- No need

## *Internet Connection Firewall*

# New group policy support

**What is it?**

- More objects for better control
- Operational mode, allowed programs, opened ports (static), ICMP settings, enable RPC

**Why do it?**

- Better management between corporate and standard profiles

**What's different?**

- IPv4 only (IPv6 still just on/off)
- Final GPOs might change

**How do I fix it?**

- No need

# ICF— Inbound APIs

- IPv4 inbound connections for applications and services
- IPv4 inbound connections on RPC and DCOM ports

# Inbound applications (IPv4)

## Issue

- Application needs to bind to a socket and accept inbound requests

## Do this

- Call `INetFwV4AuthorizedApplication` as either enabled or disabled
- Provide image file name, friendly name, and whether all traffic or local subnet

## Notes

- When application starts, ICF dynamically opens ports
- App must run as local admin to add to list, but can run in any context later
- Apps should get user consent
- Cannot add `SVCHOST.EXE`

# Inbound services (IPv4)

## Issue

- Service ports usually need to remain open always

## Do this

- Call `INetFwV4OpenPort` as either enabled or disabled
- Provide port number, protocol, friendly name, and whether all traffic or local subnet

## Notes

- When service starts, ICF opens ports
- Service must run as local admin
- Limit to local subnet whenever possible
- Service should get user consent
- Service should close ports if disabled

## *Internet Connection Firewall*

# Inbound RPC/DCOM (IPv4)

### Issue

- RPC handled by ICF's new RPC awareness

### Do this

- Call `INetFwV4Profile`
- Set `AllowRpcPorts` to "true"

### Notes

- App or service must run as local admin to enable RPC, but can run as admin, network service, or local service later
- App or service should get user consent
- Service should close ports if disabled

# RPC restrictions

- Restrict remote clients
- Require authentication to endpoint mapper (135/tcp)
- New interface registration flags

# Restricting remote clients

**What is it?**

- `RestrictRemoteClients` registry key to enforce authentication
- Remote anonymous calls to RPC interfaces now rejected by default

**Why do it?**

- Useful mitigation against worms that rely on exploitable buffer overruns invoked through anonymous connections

**What's different?**

- Apps that expect anonymous calls might be affected

**How do I fix it?**

- Require clients to use RPC security
- Exempt interface from authentication using exemption flag

## *RPC restrictions*

# Endpoint mapper authN

**What is it?**

- Clients always contact EP mapper anonymously
- If client restrictions are set, clients also won't be able to contact EP mapper

**Why do it?**

- Setting `EnableAuthEpResolution` key tells RPC client to use NTLM authentication to EP mapper

**What's different?**

- Both peers will need XP SP2

**How do I fix it?**

- No need

## *RPC restrictions*

# New i/f registration flags

**What is it?**

- Three new flags for developers to use in applications

**Why do it?**

- Provide additional security tools to make RPC better

**What's different?**

- No affect on existing RPC applications

**How do I fix it?**

- No need

## *RPC restrictions*

# New i/f registration flags

- **RPC\_IF\_ALLOW\_CALLBACKS\_WITH\_NO\_AUTH**
  - RPC runtime invokes registered security callback for all calls
  - Without: RPC rejects all unauthenticated calls before reaching security callback
- **RPC\_IF\_SEC\_NO\_CACHE**
  - Disables security callback caching
- **RPC\_IF\_LOCAL\_ONLY**
  - Reject remote client calls
  - Reject local calls over all ncadg\_\* protocols
  - Reject all calls over ncacn\_\* protocols (except...)
  - Reject all calls over ncacn\_np if not from SVR
  - Allow ncalrpc calls

# DCOM enhancements

- Computer-wide restrictions
- More specific COM permissions

# DCOM enhancements

- Don't apply to in-process COM
- Apply if your DCOM server meets any:
  - Access permission for app is less stringent than permission necessary to run it
  - App is usually activated on a Windows XP computer by a remote COM client not using administrative account
  - App uses unauthenticated remote callbacks
  - App is meant to be used locally



# Computer-wide restrictions

**What is it?**

- Computer-wide access controls that govern access to all DCOM requests on the computer
- An additional `AccessCheck` against the ACL for on each call, activation, or launch of any COM server

**Why do it?**

- Minimum authorization bar that must be passed to access COM servers
- Allows administrators to override weak security settings in an application's `CoInitializeSecurity`
- ACLs checked when interfaces exposed by RPCSS are accessed

*DCOM enhancements*

# Computer-wide restrictions

<b>Permission</b>	<b>Administrator</b>	<b>Everyone</b>	<b>Anonymous</b>
<b>Launch</b>	<b>Local launch</b>	<b>Local launch</b>	
	<b>Local activate</b>	<b>Local activate</b>	
	<b>Remote launch</b>		
	<b>Remote activate</b>		
<b>Access</b>		<b>Local call</b>	<b>Local call</b>
		<b>Remote call</b>	

# Computer-wide restrictions

**What's different?**

- Local scenarios will continue to work
- Most COM client scenarios will continue to work
- Unauthenticated remote calls will break
- Only administrators can remotely activate and launch

**How do I fix it?**

- Don't write apps that require remote activation by non-admin client or remote unauthenticated calls!
- Can change new defaults with registry keys

## *DCOM enhancements*

# More specific COM perms

**What is it?**

- Distinguish COM access rights based on distance: local (LRPC), remote (eg., RPC over TCP)

**Why do it?**

- Create precise COM permission policy
- Restrict app so it can only be used locally

**What's different?**

- Launch/activate ACEs: LL, RL, LA, RA
- Access (call) ACEs: LC, RC
- Generally backward-compatible, some specific ACL alterations might be needed

**How do I fix it?**

- Search MSDN on "LaunchPermission"

# Memory protection

- Execution protection (NX)

## *Memory protection*

# **NX—“no execute”**

- Prevents code execution in data pages:
  - Default heap
  - Various stacks
  - Memory pools
- Both user and kernel modes
- Requires developers to explicitly mark pages as executable

## *Memory protection*

# **NX—“no execute”**

- OS feature that relies on processor hardware to mark memory
- Functions on a per-VM page basis
- Common: change a bit in the page table entry to mark the page
- Affects apps that:
  - Perform just-in-time code generation
  - Execute memory from default process stack or heap

## *Memory protection*

# **NX—“no execute”**

- **Hardware implementation varies by processor**
- **Processor must raise exception when code executes from disallowed page**
- **Current processor support**
  - **AMD K8 (32-bit Windows)**
  - **Intel Itanium (64-bit Windows)**

*Memory protection*

# 64-bit Windows

What is it?

- Applications *expected* to function with NX enabled by default!
- Protected areas
  - Stack
  - Paged pool
  - Session pool
  - Default process heap
- Can't be disabled
- To allocate virtual memory—
  - Call `VirtualAlloc()` with one of the `PAGE_EXECUTE_*` attributes

*Memory protection*

# 32-bit Windows

What  
is it?

- **User mode**
  - AMD processors with “physical address extension” mode enabled
  - Investigating per-application methods to disable or enable NX
  - Result: unhandled exception (blue screen)  
`STATUS_ACCESS_VIOLATION (0xc0000005)`
- **Kernel mode**
  - Only to the stack by default
  - Can't be enabled/disabled on per-driver basis
  - Result: bugcheck `0xFC: ATTEMPTED_EXECUTE_OF_NOEXECUTE_MEMORY`

## *Memory protection*

# All versions

**Why do it?**

- Many worms and viruses execute code from data pages
- NX reduces impact—can't spread now
- Encourages good software engineering

**What's different?**

- Apps that perform dynamic code execution might break
- Drivers that expect 64-bit addressing or >4 GB RAM in PAE mode might break
- Drivers that do DMA transfers

**How do I fix it?**

- Mark generated code with an execute permission
- Update apps that execute from stack, default process heap, or dedicated heap
- DMA transfers are double-buffered

# Safer e-mail handling

- Not done yet! ☹️

# More secure browsing

- Add-on management and crash detection
- Binary behaviors security settings
- BindToObject mitigation
- MSJVM security setting
- Local machine zone lockdown



# More secure browsing

- **MIME handling enforcement**
- **Object caching**
- **Pop-up manager**
- **Untrusted publishers mitigations**
- **Window restrictions**
- **Zone elevation blocks**

*More secure browsing*

# Add-on management

**What is it?**

- View and control all IE add-ons, including ones previously difficult to detect
  - Browser helper objects
  - ActiveX controls
  - Toolbar extensions
  - Browser extensions
- Status bar and balloon notifications

**Why do it?**

- Error reporting data shows add-ons create significant instability
- Many pose security risks

*More secure browsing*

# Add-on management

**What's different?**

- Disabled add-ons not removed; IE simply won't instantiate them
- Applies only to IEXPLORE.EXE and EXPLORER.EXE
- Other programs based on IE components won't respect disabled state

**How do I fix it?**

- Use "Manage Add-ons" to restore broken functionality
- Restart IE

*More secure browsing*

# Add-on admin control

- Can alter user control of add-ons through registry key (apply with GPO)
  - Normal: user has full control (default)
  - AllowList: admin specifies which add-ons are allowed; users can't change
  - DenyList: admin specifies which add-ons are denied; users can run others

*More secure browsing*

# Add-on crash detection

- Crash detection program launches when IE crashes; collects:
  - List of DLLs that are loaded
  - Value of instruction pointer (EIP)
- Finds DLL whose memory range the EIP lies within; DLL must be:
  - Non-system
  - A COM server for an IE add-on
- Displays dialog to manage
  - Disable from here



*More secure browsing*

# Binary behaviors setting

**What is it?**

- Components, attached to HTML, that encapsulate specific functionality
- New “URL Action” setting in each zone

**Why do it?**

- Unrestricted binary behaviors could be exploited
- Allow users to control binary behaviors

**What's different?**

- Disallowed in restricted sites zone

**How do I fix it?**

- Custom security manager for apps that need to run in restricted sites zone
- <http://go.microsoft.com/fwlink/?linkid=21863>

*More secure browsing*

# BindToObject mitigation

**What is it?**

- Apply security policies consistently at source of URL binding: URLMON

**Why do it?**

- Uniformly enforce ActiveX security model rather than relying on calling code
- Eliminates exploits that use IE to compromise vulns in calling code

**What's different?**

- Any component that wants to resolve a URL and get back a stream or object

**How do I fix it?**

- <http://go.microsoft.com/fwlink/?linkid=21814>

*More secure browsing*

# MSJVM security setting

**What is it?**

- Separate setting to control MSJVM
- Existing JVM setting renamed

**Why do it?**

- No known threats to MSJVM

**What's different?**

- Clean installs of these will lack MSJVM:
  - Windows XP SP 2 full OS
  - Windows Server 2003
  - Windows 2000 SP 4 full OS
- Upgrading won't remove MSJVM

**How do I fix it?**

- Need to transition away from MSJVM
- <http://go.microsoft.com/fwlink/?linkid=21850>

## *More secure browsing*

# Local machine zone lockdown

**What is it?**

- A non-displayed security zone that runs all local HTML pages on a computer

**Why do it?**

- Helps stop malicious local code from elevating privilege

**What's different?**

- Enabled for IE processes
- Not enabled for non-IE processes

**How do I fix it?**

- Can save HTML as .HTA (dangerous: full privileges)
- Use “mark of the web” comments to load file into another security zone

*More secure browsing*

# Local machine zone lockdown

- **Overridden URL actions**
  - **Run ActiveX: disallow**
  - **Override ActiveX safety: disallow**
  - **Run scripts: prompt**
  - **Cross domain data: prompt**
  - **Block binary behaviors: disallow**
  - **Java permissions: disallow**

*More secure browsing*

# MIME handling enforcement

**What is it?**

- IE checks received files in four ways:
  - File name extension
  - Content-Type from HTTP header (MIME type)
  - Content-Disposition from HTTP header
  - MIME sniff

**Why do it?**

- Eliminates improper handling of mis-reported files (eg., .EXE assumed as text)

**What's different?**

- If MIME sniff results in different type, IE changes file extension in cache
- Never elevates to a more dangerous type

**How do I fix it?**

- Report your MIME types correctly!

*More secure browsing*

# Object caching

**What is it?**

- New security context on all scriptable objects
- Access blocked when navigating away from current FQDN

**Why do it?**

- Single MSHTML instance across navigations; cached objects available
- Eliminate current cross-domain hole exploitable by frames

**What's different?**

- Four more bytes added to cached markup

**How do I fix it?**

- Probably nothing here

*More secure browsing*

# Pop-up manager

**What is it?**

- **Blocks automatic and background pop-up windows activated by:**
  - `window.open()`
  - `window.external.navigateAndFind()`
  - `showHelp()`
- **Doesn't affect windows opened by:**
  - Mouse click
  - Locally-running software
  - ActiveX controls on a web site
  - Trusted sites or local intranet zones

**Why do it?**

- **Pop-ups suck!**

*More secure browsing*

# Pop-up manager

What's different?

- Allowed windows that open outside viewable screen are positioned onto viewable area
- Allowed windows that open larger than the viewable screen are resized to the viewable area

How do I fix it?

- No need

*More secure browsing*

# Pop-up manager

- **Notification and sound, with choices:**
  - Show blocked pop-up
  - Allow pop-ups from this site
  - Block pop-ups
  - Open pop-up management options
- **Configuration choices**
  - Allow list
  - Block all, including clicked pop-ups
  - Override key for above
  - Sound
  - Zones

*More secure browsing*

# Untrusted publishers mitigations

**What is it?**

- Block all signed content from a publisher
- One prompt per control per page
- Block invalid signatures
- Display ellipsis if text is longer than box

**Why do it?**

- Eliminate repeated prompts
- Stop modified code

**What's different?**

- New functionality
- Reduces social engineering tricks

**How do I fix it?**

- Not needed

*More secure browsing*

# Window restrictions

**What is it?**

- Scripts can't position or resize windows with title and status bars offscreen
- Scripts can't turn off status bar

**Why do it?**

- Eliminates windows that try to spoof desktop objects
- Allows users to always see security zone

**What's different?**

- Title and status bars will always be visible to users

**How do I fix it?**

- Must change code that will break

*More secure browsing*

# Pop-up window restrictions

- Unrestricted “chromeless” windows can cover important UI elements and deceive users
- Script-initiated pop-ups are constrained
  - Appear between top and bottom of parent window “chrome”
  - Must overlap some part of parent window
  - Must stay immediately on top of parent (eg., can't be placed over dialogs)



*More secure browsing*

# Zone elevation blocks

**What is it?**

- IE prevents the security context for any link from being higher than the context of the current page

**Why do it?**

- Stop scripts from navigating to higher security zone

**What's different?**

- Web pages that try to call more privileged pages will fail
- Only a user-clicked link can go to higher privilege

**How do I fix it?**

- Fix apps to require user initiation

# Improved computer maintenance

- Not done yet! ☹️

**OK, what's next?**



# More resiliency

- **Increase protection and security of Windows XP**
  - Even if updates haven't been installed
- **Implications for users and developers**
- **The next step of trustworthy computing**



# Updates

- “New security technologies in Windows XP Service Pack 2”
- <http://go.microsoft.com/fwlink/?linkid=20969>



**Steve Riley**  
**steriley@microsoft.com**

***Microsoft***<sup>®</sup>