

AFS on Windows



Jeffrey Altman
Your File System Inc.
2015 AFS and Kerberos Best Practices Workshop

The Explorer Shell Caching Bug



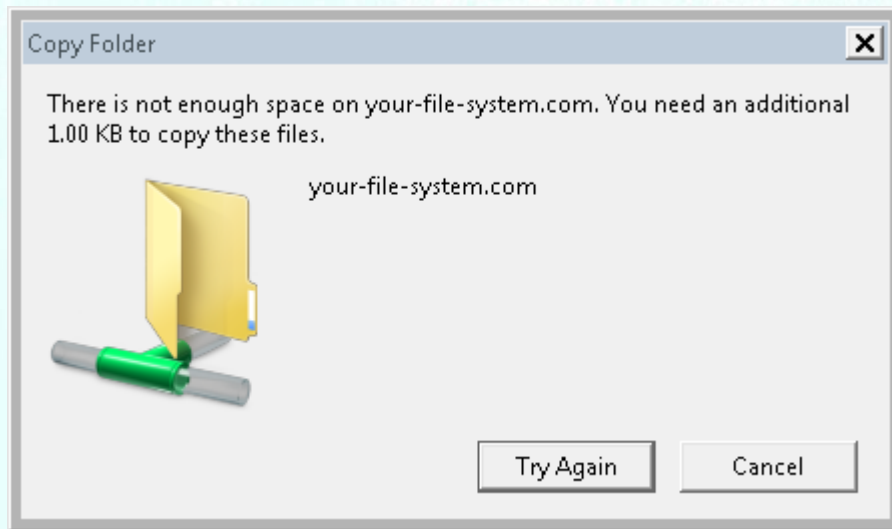
What is the Explorer Shell Caching Bug?

- ◎ The Shell caches for each directory object
 - Attributes
 - Volume Information → (AFS Volume ID = Device ID)
- ◎ The Shell caches entries for directories without attributes or volume information
- ◎ Shell believes that no Reparse Point has been crossed
- ◎ Volume Info unknown so queries info for \\afs\cell\



What are the Symptoms?

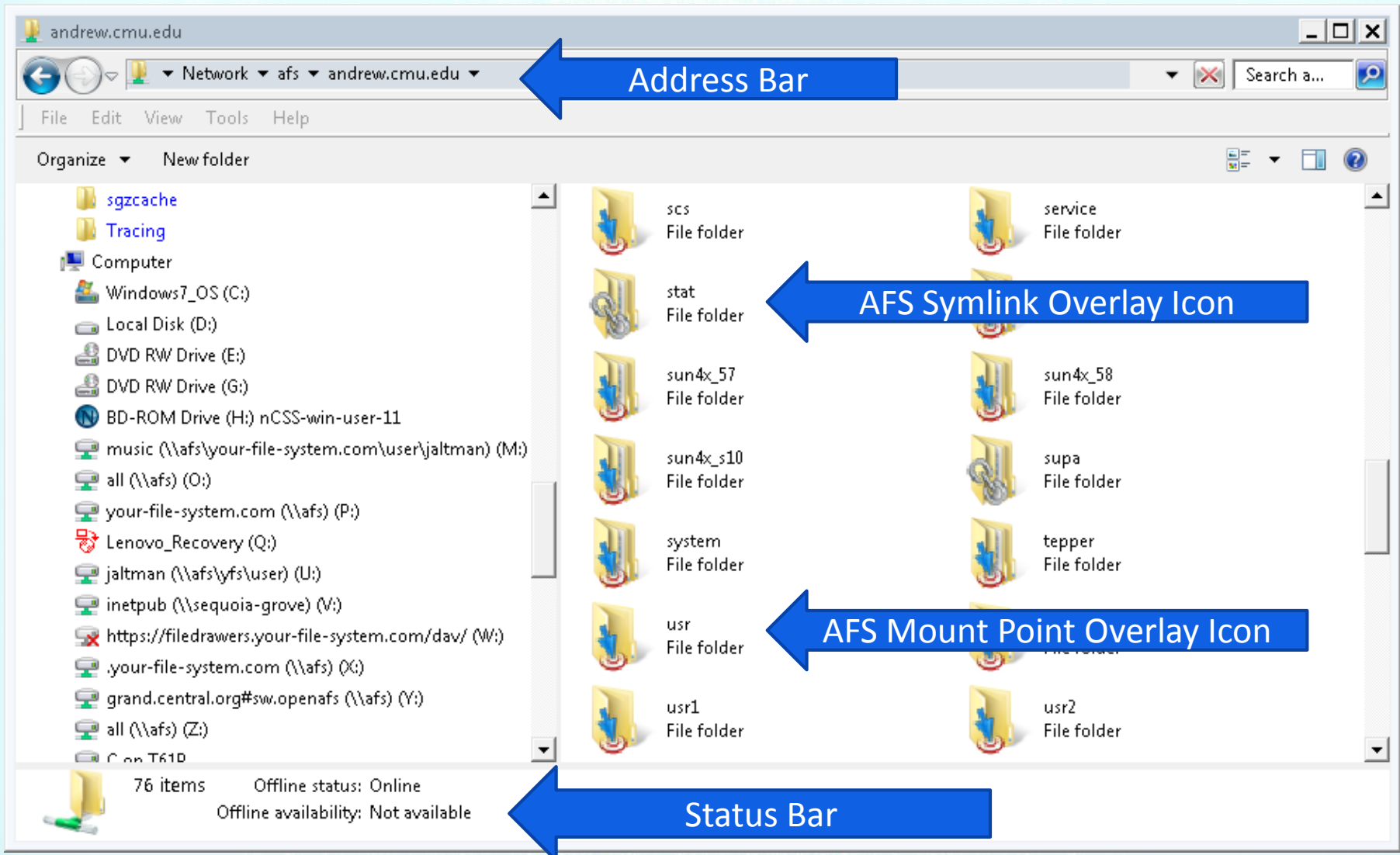
- If \\afs\cell\ refers to RO then attempt to write triggers READ ONLY VOLUME or 0 bytes free error
- If \\afs\cell refers to small RW volume, then insufficient space error is possible



Fixed in Windows 10

- ◎ This bug was fixed about a month before Windows 10 RTM
- ◎ This is one important reason for your users to upgrade to Windows 10
- ◎ BUT ... Overlay Icons are broken

The Explorer Shell



Known Bugs in 1.7.32

- ◎ Group Policy Service vs Mapped Drives [11909]
 - GPSVC issues drive mapping requests using restricted process handles
- ◎ WKS Pipe Service vs Explorer Shell [11924]
 - Shell API implementation does not check error codes
- ◎ VLDB Lookup Race [11919]
- ◎ Readonly volume failover bug [11920]



Ambiguous File Names

- ◎ FOO != foo != Foo != FoO, but
 - If the directory search is for “FOO” and there is only an entry for “foo”, return “foo”
 - If the search is for “FOO”, and “foo” and “Foo” exist, which should be returned?
 - There is no right answer – FAIL IT!!!!
- ◎ The Windows AFS SMB interface implemented this behavior
- ◎ The AFS redirector does not get it right



Explorer Shell vs `/afs/andrew.cmu.edu/usr/`



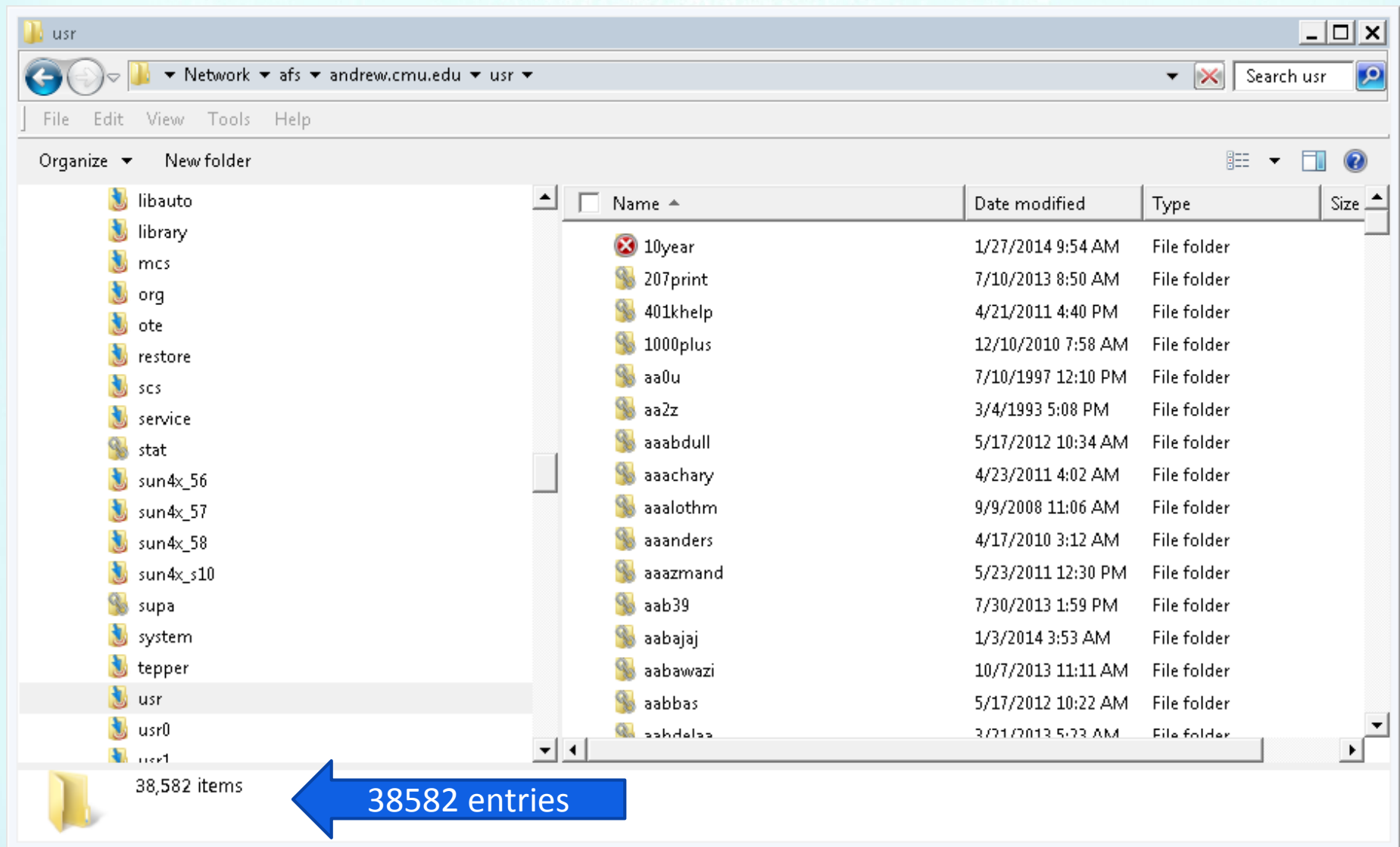
Why is `\\afs\andrew.cmu.edu\usr` a bad idea

- ◎ The `usr` directory contains more than 38,000 symlink entries
 - `/afs/andrew/usr/tequila -> (symlink)`
`/afs/andrew/usr11/tequila -> (mp)`
`#user.tequila`
- ◎ Windows requires that all directory entries be presented with full status info
- ◎ All symlinks must be evaluated at least until the mount point
 - Nearly 85,000 stat objects required



Your File System

</afs/andrew.cmu.edu/usr/>



Status Info and Callbacks

- ◎ The default stats cache on Windows is 10,000 entries
- ◎ 85,000 entries produces large amounts of stat cache thrashing
- ◎ This is exacerbated by the AFS Redirector design that requires whole directories including status to be present in kernel



FindFirst / FindNext vs Explorer Shell



AFS Redirector vs FindFirstFile / FindNextFile / FindClose

◎ OpenAFS 1.7 behavior

- Construct full directory in kernel plus status info before returning from FindFirstFile
- Optimized to return full buffers to application
- Results in application blocking on slow links with large directories



AFS Redirector vs FindFirstFile / FindNextFile / FindClose

◎ AuriStor[®] behavior

- Construct full directory listing in service in FindFirstFile
- Request entries plus status in FindNextFile
- Return as many entries as possible within 200ms
- Blocking
 - Waits for the directory enumeration in FindFirstFile
 - Waits for status info on first FindNextFile entry



Benefits of Directory Enum changes

- ⦿ Explorer Shell remains responsive
- ⦿ File Count increases as entries are added
- ⦿ Shell Extensions are more likely to access objects while their status is in the AFSCache



AFS Symlinks vs Microsoft Symlinks

- AFS Symlinks are POSIX
 - Target type is not encoded in the target path
 - Relative or absolute paths
 - Forward slash separators
- MSFT Symlinks are not POSIX
 - Target type is encoded in the symlink object
 - Either a directory or a file with RP Data attached
 - Relative or absolute paths
 - Backslash separators



Callback Processing Changes

OPENAFS 1.7

- ⦿ Callback Expiration processed in the service for the afs redirector

AURISTOR

- ⦿ AFS redirector processes its own callback expiration

Benefits:

1. File Status can be recycled in AFSCache without invalidating kernel data.
2. Fewer userland -> kernel IOCTL calls reduces CPU utilization.



Directory Enumeration Changes

OPENAFS 1.7

- ◉ All directories fully populated in kernel with complete status information

AURISTOR

- ◉ Sparse directory enumeration in kernel
- ◉ Entries cached as needed

Benefits:

1. Fewer directory entries allocated
 1. Smaller kernel memory footprint
 2. Less CPU spent on garbage collection
 3. Fewer MPs and Symlinks evaluated
 4. Fewer RPCs issued



The Results

- 10% to 15% reduction in wall clock time when building OpenAFS Windows in /afs over WAN.
- 30% reduction in AFS Service / kernel CPU time.



File System Requirements for Win10 and Server 2016

- ⦿ Microsoft must sign all drivers for Windows 10 and Server 2016
- ⦿ Microsoft is requiring Certification for all drivers to support Server 2016 before they will sign
- ⦿ Server 2016 certification adds a large number of requirements



UNC Hardening [MS15-011]

- ⦿ Group Policy Service reads new configuration from DCs at system boot
- ⦿ Must guarantee that mutual auth, integrity and encryption is used for the network path
- ⦿ Failure to do so opens a man-in-the-middle attack
- ⦿ MSFT solved this problem for SMB/CIFS by implementing a new Extended Create Parameter in the kernel.



UNC Hardening vs OpenAFS

- ◎ The guarantees required by UNC Hardening cannot be provided by the “rxkad” security class
 - “rxgk” is required
- ◎ AuriStor® integrates with the Multiple UNC Provider to enforce UNC Hardening policies
 - Supported on Vista and above with hot fixes applied



IPv6

- ◎ The IPv4 address space has been exhausted in many regions of the world
- ◎ The U.S. Federal Acquisition Record (FAR) requires IPv6 support for all software acquisition
- ◎ Microsoft is requiring support for IPv6 for all network file systems matching the FAR



Windows Server 2016 Nano and Server Core

◎ Server Core

- New default installation
- No GUI
- All installation and configuration must be performed via Power Shell

◎ Nano

- No console
- All installation, configuration, and administration must be Power Shell or Windows Management Instrumentation



One Final OpenAFS 1.7 release

- ⦿ Will fix outstanding bugs
- ⦿ Will use a non-Microsoft digital signature
- ⦿ Will be compatible with Windows 10 and Windows 7 but will not be compatible with Server 2016



Questions! Answers?





Your File System

255 W 94TH ST

New York NY 10025 USA

+1 212 769-9018

sales@your-file-system.com

<http://www.your-file-system.com>