

Windows Enterprise OU Administrator Tips Imaging Systems That Use OpenAFS

August 9, 2007

IT Administrators should be aware of special steps that need to be done during the deployment of an image that uses OpenAFS. Make sure during your SysPrep/imaging process you have used one of the methods below. If you do not, all systems will have the same AFS UUID. This has the potential for causing a disruption of AFS file servers (which will probably result in all your imaged systems being blocked from AFS file usage until the problem is fixed).

Both of the following methods are from the OpenAFS “Release Notes”:

Method 1 – AFSCache and SysPrep

3.38. AFS Client Universally Unique Identifiers vs. System Cloning

The OpenAFS Client implements Universally Unique Identifiers (UUIDs). They are used to provide the AFS file server with a method of identifying the client that is independent of IP address. This permits the AFS file server to track mobile clients or those behind Network Address Translators when they move from address to address or port to port. Tracking the client improves client performance by permitting callback state to be maintained across location changes. The UUID is generated when the AFSCache file is created and is maintained as long as the contents of the AFSCache file are valid. The UUID is stored in the AFSCache file.

When cloning machines that have Windows AFS client installed it is necessary to generate a new UUID for each client. This will be done automatically if the Windows Machine SID is re-generated using Microsoft SysPrep. If the SID is not being re-generated either the AFSCache file should be deleted or the command “fs uuid –generate” must be executed as part of the cloning process. Multiple AFS clients reporting the same UUID will not only result in horrible AFS client performance and cache inconsistencies, but they will also put a tremendous strain on the AFS file servers.

Method 2 – NonPersistent Caching

Alternately, cloned machines can have the registry key “NonPersistentCaching” turned on. This creates a new cache each time the service starts. As a side effect the uuid is generated each time.

(“Appendix A” of the Release Notes)

Value: NonPersistentCaching

Type: DWORD [0..1]

Default: 0

Variable: buf_CacheType

When this registry value is set to a non-zero value, the CachePath value is ignored and the cache data is stored in the windows paging file. This prevents the use of persistent caching (when available) as well as the ability to alter the size of the cache at runtime using the "fs setcachesize" command.