IBM Education Assistance for z/OS V2R2

Item: NFS Enhancements Part I

Element/Component: z/OS Network File System (NFS)

Material current as of May 2015

Agenda

- Trademarks
- Presentation Objectives
- For each of the 6 items covered in Part I, as necessary
  - Overview
  - Usage & Invocation
  - Interactions & Dependencies
  - Migration & Coexistence Considerations
- Installation
- Presentation Summary
- Appendix

Trademarks

- See url http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

Presentation Objectives

- Introduce the following enhancements:
-
  - Display network domain when nfsv4domain is not specified
  -
  - zNFSC mount option forcing UNSTABLE NFSv4 write like v3
  -
  - Cross check SiteAttr vs Checklist
  -
  - zNFSC 64bit PFS
  -
  - bpxmtext support
  -
  - Persistent Filehandle in USS

Overview – Display network domain when nfsv4domain attribute omitted

- **Problem Statement / Need Addressed**
  - Before V2R2, z/OS NFS Server displays "nfsv4domain(*my.domain*)" in the showattr utility output where *my.domain* has to be the specified value of "nfsv4domain()" in the site attribute. If “nfsv4domain()” attribute is omitted from the site attribute file, a blank value is returned which could cause confusion to system administrators.
- **Solution**
  - If the "nfsv4domain()" is not specified in the site attribute, z/OS NFSS will display "nfsv4domain*(default.tcpip.domain)" where the "default.tcpip.domain" is the returned domain from z/OS TCPIP.
- **Benefit / Value**
  - This item will increase the usability of NFS and will bring more awareness during the daily business. NFS should be able to point to the TCP/IP stack which is connected to NFS Server.

Usage & Invocation – Display network domain when nfsv4domain attribute omitted

- **Domain value provided**
  - bash-3.2$ showattr V2R1 or V2R2

    .....

    nfsv4domain(storage.sanjose.ibm.com)

    .....
- **Domain value not provided**
  - bash-3.2$ showattr V2R1

    .....

    nfsv4domain( )

    .....
- **Domain value not provided**
  - bash-3.2$ showattr V2R2

    .....

    nfsv4domain*(tuc.stglabs.ibm.com)

    .....

Interactions & Dependencies – Display network domain when nfsv4domain attribute omitted

- **Software Dependencies**
  - None
  -
- **Hardware Dependencies**
  - None
  -
- **Exploiters**
  - zNFSS users who want to know the default network domain used for NFSv4 name mapping

Overview – zNFS Client mount option for UNSTABLE NFSv4 write

- **Problem Statement / Need Addressed**
  - Before this item, z/OS NFS Client did not move written buffers to LRU until the buffers are write committed on the NFS Server side. This could lead to out of memory situation if the zNFSC was writing a huge file and there was no explicit COMMIT requests. The zNFSC could ask the server to commit the writes immediately by sending a STABLE NFSv4 write requests with implicit COMMIT. However, this would degrade server performance.
- **Solution**
  - This item introduces zNFSC mount option syncwrite(y/n), which could force zNFSC to move written buffers to LRU even with UNSTABLE (i.e. non-committed) NFSv4 writes sent to servers.
- **Benefit / Value**
  - Prevent zNFSC from out of memory situation without hurting server side performance.

Usage & Invocation – zNFS Client mount option for UNSTABLE NFSv4 write

- Specify syncwrite(n) as a mount option during mount time
  - e.g.

```
USER2:>mount -tnfs -w0 -f T3_FP0552 -o'mbull:/hfs/keith1,vers(4),syncwrite(n)' /mnt1

USER2:>nfsstat -m
mbull:/hfs/keith1,vers(4),syncwrite(n) is mounted on /mnt1 filesystem T3_FP0552
ipaddr=::ffff:x.x.x.x
:
:
```

- syncwrite(y) is the default value when this mount option is not specified. i.e. z/OS NFS Client writes the way it was as in V2R1

Interactions & Dependencies – zNFS Client mount option for UNSTABLE NFSv4 write

- **Software Dependencies**
  - None
  -
- **Hardware Dependencies**
  - None
  -
- **Exploiters**
  - All NFS users who use NFS version 4 mounts to the z/OS Unix file systems.

Migration & Coexistence Considerations – zNFS Client mount option for UNSTABLE NFSv4 write

- **Toleration/coexistence APARs/PTFs**
  - APAR OA45165 tolerates this new mount parameter syncwrite(y/n) in lower releases
  -
- **Migration**
  - None
  -
- **Coexistence**
  - None

Overview – Cross Check Site Attribute vs Checklist

- **Problem Statement / Need Addressed**
  - Before this item, z/OS NFS Server quietly ignored the directory_suffix (aka checklist entries) of each NFS exported entries asking for SAF checking, when the site attribute file specifies nochecklist.
  -
- **Solution**
  - Issue GFSA519I console message indicating conflict between site attribute and export entries' directory_suffix.
  -
- **Benefit / Value**
  - System programmer / administrators are aware of the reason of directory_suffix (particularly SAF / RACF permission) checking being dropped.

Usage & Invocation – Cross Check Site Attribute vs Checklist

Example 1 – SAF zNFSS with nochecklist

1)Server site attribute set to nochecklist, security(saf)

security(saf)

nochecklist

2)Export file set to

/hfs/vndrcv/d1<hosts=tinsun,nosaf>

/hfs/vndrcv/d2<hosts=tinsun,nosaf>

/hfs/vndrcv/d3<nosaf>

3)Issue “modify mvsnfs,exportfs” operator command

00- 04.44.35 SYSTEM1 f mvsnfs,exportfs

04.44.35 SYSTEM1 STC00087 GFSA519I (MVSNFS) Checklist entries are ignored because of **nochecklist**.
04.44.35 SYSTEM1 STC00087 GFSA796I (MVSNFS) EXPORTFS: completed successfully.

Usage & Invocation – Cross Check Site Attribute vs Checklist

Example 2 – Non-SAF zNFSS with checklist

1)Server site attribute set to checklist, security(exports)

- security(exports)
- checklist
-

2)Export file set to

- /hfs/vndrcv/d1<hosts=tinsun,nosaf>
- /hfs/vndrcv/d2<hosts=tinsun,nosaf>
- /hfs/vndrcv/d3<nosaf>
-

3)Expect "GFSA519I (MVSNFS) Checklist entries are ignored because of security(exports). "

- 03.48.51 SYSTEM1 STC00079 GFSA519I (MVSNFS) Checklist entries are ignored because of **security(exports)**.

Interactions & Dependencies – Cross Check Site Attribute vs Checklist

- **Software Dependencies**
  - None
  -
- **Hardware Dependencies**
  - None
  -
- **Exploiters**
  - System programmer / administrator who wants to be notified when checklist entries are conflicting between z/OS NFS server site attributes and exports data set.

Overview – z/OS NFS Client LP64 / Amode64

- **Problem Statement / Need Addressed**
  - z/OS Unix System Service (USS) moving to Amode64 in V2R2. If z/OS NFS Client were to stay in Amode31
    - numerous AMODE switching between Amode64 USS and Amode31 NFS PFS
    - USS would have to create numerous structures below 2GB for Amode31 PFS to access
- **Solution**
  - z/OS NFS Client also move up to Amode64
- **Benefit / Value**
  - Not only this eliminates the extra overhead between USS Amode64 and NFSC Amode31, this support greatly relieves the 2GB address space constraints, allows greater scalability, complies with the larger and wider data flow from end-to-end

Usage & Invocation – z/OS NFS Client LP64 / Amode64

- Startup JCL/Procedure with
  - REGION=0M, or =xxxM and
  - **MEMLIMIT=xxxxG** (memory above2GB)
  -
    - The REGION (existed before V2R2) specifies the total size of usable virtual storage below-the-bar of an address space
    -
    - The MEMLIMIT (new in V2R2) specifies the total size of usable virtual storage above-the-bar of an address space
    -
    - When REGION=0M, it implies MEMLIMIT=NOLIMIT (unlimited memory above2GB), so it is optional to add MEMLIMIT.
    -
    - When REGION=xxxM, it is better to add or specify MEMLIMIT=xxxG.

Migration & Coexistence Considerations – z/OS NFS Client LP64 / Amode64

- **Toleration/coexistence APARs/PTFs**
  - None
  -
- **Migration**
  - The support requires some modifications to the z/OS NFS Client Startup Procedure (MVSNFSC) to specify the MEMLIMIT (the virtual storage above2GB)
  -
- **Coexistence**
  - None

Overview – z/OS NFS bpxmtext support

- **Problem Statement / Need Addressed**
  - The BPXMTEXT tool handled USS, TCPIP, ZFS, C-Runtime/LE etc Reason Codes but not NFS's. Customer would like to see it expanded to also include NFS Reason Codes in order to improve diagnosing problems
  -
- **Solution**
  - The z/OS NFS Client is changed to support bpxmtext for all three (3) sub-components:
    - 0x6D – z/OS NFS Common or shared modules (ie., Ctrace, MSGNLS, etc...);
    - 0x6E – z/OS NFS Client;
    - 0x6F – z/OS NFS Server.
    -
- **Benefit / Value**
  - This enhances the BPXMTEXT tool which many customers rely on for diagnosing problems displaying a reason code with the error message.

Usage & Invocation – z/OS NFS bpxmtext support

- **Example 1 – Valid Client RSN**

USER2:/home/user2(8):>bpxmtext 6e050015

NFSERROR z/OS NFS Client

NFSERR_ISDIR: Is a directory.

A component of the specified path name exists, but it is a directory, when a non-directory object was expected.

Action: Ensure that the specified path name is correct.

From NFS Publication (Chapter 21 – Reason Codes)

05	NFS protocol error	0015	**EISDIR** Is a directory *Probable cause:* An attempt was made to open a directory with write mode specified. *Action:* Ensure that the access mode is correct.

Chapter 21. Reason codes 447

Usage & Invocation – z/OS NFS bpxmtext support

- **Example 2 – Undefined RSN**

  USER2:/home/user2(18):>bpxmtext 6f998088
  GFSAXMAI z/OS NFS Server
  GFSAXMAI+136

  Action: Consult your IBM service representative.

- **Example 3 – Invalid RSN**

  USER2:/home/user2(14):>bpxmtext 6fee0008
  unkn_mod z/OS NFS Server

- **V2R1 example – Without support**

  USER2:/home/user2(1):>bpxmtext 6e050015
  BPXMTEXT does not support reason code qualifier 6E05

Overview – Persistent Filehandle (Alias Feature)

- **Problem Statement / Need Addressed**
  - Although USS filesystems have saved filesystemID and fileID for nodes, at times z/OS NFS Server cannot maintain connection to these nodes, especially in certain customer environments (dual-copy filesystem maintenance with multiple filesystem IDs and/or physical paths to the filesystem).
  -
  - A recovery scenario is launched with the NFS4ERR_FHEXPIRED error, but some external NFS clients do not handle this scenario, forcing customer intervention upon z/OS NFS server restart/failover scenarios (unmount/remount).
  -
- **Solution**
  - Introduction of exports “alias” feature
  - Reduce the instance of FHEXPIRED error scenarios
  -
- **Benefit / Value**
  - Reduced instance of filehandle recovery instances requiring customer intervention, even when filesystem changes filesystemID (device number) or physical pathname.

Usage and Invocation – Persistent Filehandle (Alias Feature)

- **Server site attribute set to “alias”**
  -
  - Enables exports file parsing of exports aliases
  -
- **Export entry includes “alias” keyword**
  -
  - /hfs/path1/symlink2/path3 -alias=mypath3
  -
- **Client mounts to “mypath3”**
  -
  - Server sees mypath3 as an alias, interprets “/hfs/path1/symlink2/path3”
  - Offers simplicity of use and flexibility of interpretations, optionally including changeable symbolic link contents within the export entry pathname

Interactions & Dependencies – Persistent Filehandle (Alias Feature)

- **Software Dependencies**
  - None
  -
- **Hardware Dependencies**
  - None
  -
- **Exploiters**
  - System programmer / administrator who wants to use a non-IBM NFS client that does not recover from NFS4ERR_FHEXPIRED scenarios.
  -
  - System programmer / administrator who wants to use differing paths to a filesystem (dual copy maintenance, for instance) between NFS server instances in a multiplex environment

Migration & Coexistence Considerations – Persistent Filehandle (Alias Feature)

- **Toleration/coexistence APARs/PTFs**
  - In lower releases, the alias keyword is not supported in the site attributes file or the exports file. The message GFSN5032I is introduced to inform the user about it by toleration APAR OA45165.
  -
- **Migration**
  - None
  -
- **Coexistence**
  - None

Presentation Summary

- Various NFS enhancements new to V2R2 were covered:
  - Display network domain when nfsv4domain is not specified
  - zNFSC mount option forcing UNSTABLE NFSv4 write
  - Cross check SiteAttr vs Checklist
  - zNFSC 64bit PFS
  - bpxmtext support
  - Persistent Filehandle in USS

Appendix

- **SC26-7417** – *z/OS Network File System Guide and Reference*
- **RFC 3530** – *Network File System Version 4 Protocol*
- **RFC 1813** – *Network File System Version 3 Protocol*
- **RFC 1094** – *Network File System Version 2 Protocol*
- **RFC 1831** – *Remote Procedure Call (RPC) Protocol Specification Version 2*
- **RFC 1832** – *XDR: External Data Representation Standard*