
# RPCSEC_GSSv3 and NFSv4.2 ID Update

Andy Adamson
andros@netapp.com
IETF 90, Toronto Canada
Presenter Title
Date

# Topics

- GSS3: NFSv4.2 Inter SSC “Three Legged Stool”
  - Issue from IETF 89 NFSv4 WG meeting
- GSS3: NFSv4.2 Inter SSC with pNFS
  - Question from IETF 89 NFSv4 WG meeting
- GSS3: NFSv4.2 use of server side security labels
  - Scalability question

# GSS3: Inter SSC “Three Legged Stool”

- Question at IETF 89 WG meeting:
  - This system of GSS3 handles is like a three legged stool. If I break one leg due to a valid, poor behaving server can the client detect this and stop the copy?
- The answer is YES. If one handle is destroyed, or is invalidated during the secure NFSv4.2 inter server copy then the copy will not proceed
  - I submitted patches to the list to update draft-ietf-nfsv4-minorversion2-26 to address this issue.

# GSS3: Inter SSC “Three Legged Stool”

- Three GSS3 handles established using a shared secret – e.g. the “three legged stool”
- client has:
  - copy_from_auth GSS3 handle with source server
  - copy_to_auth GSS3 handle with destination server
- destination server (acting as a client) has:
  - copy_confirm_auth GSS3 handle with source server
- All three GSS3 handles need to be valid for the secure NFSv4.2 inter server copy to proceed

# Maintaining a Secure Inter SSC

- A GSS handle's validity is determined by using it.
- During a secure inter server copy, the client SHOULD use the copy_from_auth and the copy_to_auth GSS3 context handles for the NFSv4.2 lease renewing operations to the source and destination servers respectively to periodically check the validity of the handles.
  - If lease renewal fails with the GSSv3 privilege handle (RPC AUTH_ERROR), the client SHOULD retry with GSSv3 parent before expiring the lease on the client

# Maintaining a Secure Inter SSC

- An NFS NULL procedure ping can also be used for the purpose of determining a handles validity.
- If the client determines that either handle becomes invalid during the copy, then the copy MUST be aborted by the client sending an OFFLOAD_CANCEL to both the source and destination servers and destroying the respective copy related GSS3 context handles.

# Maintaining a Secure Inter SSC

- On the source server:
- The copy_confirm_auth GSS3 handle is associated with a copy_from_auth GSS3 handle on the source server via the shared secret and MUST be locally destroyed when the copy_from_auth GSSv3 handle is destroyed

# Maintaining a Secure Inter SSC

- On the destination server:
- The copy_confirm_auth GSS3 handle is constructed from information held by the copy_to_auth privilege, and MUST be destroyed by the destination server (via an RPCSEC_GSS3_DESTROY call) when the copy_to_auth GSS3 handle is destroyed.

# Maintaining a Secure Inter SSC

- The source server has the filehandle, stateid, and copy_from_auth assertion data. If a READ is attempted by the destination server using the file handle and stateid without a valid copy_confirm_auth privilege, the source server MUST deny or abort the READ and locally destroy both the copy_to_auth and copy_from_auth handles.

# Maintaining a Secure Inter SSC

- Each of the three participants (client, src and dst servers) sees 2 of the three “legs” – e.g GSSv3 handles
- For each participant, both “legs” must be valid for the participant to allow the copy to proceed.
- As long as 2 of the three participants is not compromised, the copy is secure.
- Some of above text has yet to be submitted ☺
  - I’ll submit a new patch set for this issue soon

# GSS3: Inter SSC with pNFS

- Can pNFS be used by the destination server acting as a client (D-client) with secure Inter SSC?
  - The D-client mounts the source server with pNFS enabled and with all GSS3 secure Inter SSC privileges established.
  - On Linux, the D-client READ request triggers a LAYOUTGET which triggers a GETDEVINFO
  - D-client using krb5(i:p) connects to each Data server as required using normal GSS3
  - D-client will then use the ca_src_stateid with the appropriate file handle from the layout to READ from each DS

# GSS3: Inter SSC with pNFS

- No GSS3 assertions required on DS connections
- All of the “Maintaining a Secure Connection” checks still apply to the MDS (source server) which has the responsibility to deny or abort the DS READs if needed.
- No additional GSSv3 assertions need be obtained.

# GSS3 and NFSv4.2 Server Labels

- The client sends an RPCSEC_GSS_LIST request to the server to obtain supported label types
- The client forwards a subject label to the server in an RPCSEC_GSS_CREATE message with a label assertion payload
- If granted, the resultant GSS3 handle is used for all NFS traffic asserting the server side label.

# GSS3 and NFSv4.2 Server Labels

- This means one GSS3 child context per client side security label
- Is this manageable?
- How many subject labels are typically enforced?


# Thank you