NFSv4.2 Secure Inter-server
Server Side Copy Status

William A. (Andy) Adamson
andros@netapp.com
IETF88, Vancouver
Motivation

- IETF87: No progress on draft-ietf-nfsv4-rpcsec-gssv3
  - draft-ietf-nfsv4-minorversion2-20 removed the use of RPCSEC_GSSv3
- Discussion of draft-20 on list exposed several issues with non-GSSv3 secure inter-server server side copy
  - Several choices but no clear solution from list
- IETF88: Decipher and present choices to WG
  - Disclaimer: I hope I didn’t misrepresent 😊
Secure Inter-server SSC goals

1. Source server properly authenticates the destination server

2. Destination server READ is associated with the copy and is handled in a special manner by the source (see READ stateid issue slide)

3. Destination server is granted the privilege to act on behalf of the user-principal to READ and WRITE.

4. Limit the ability of the destination server to act as the user-principal (e.g. a single copy)
READ Stateid Issue: use of ca_src_stateid

- ca_src_stateid is from the Client OPEN verified against the Client clientid (NFSv4.1)
- Destination to perform ‘normal’ READs from the source
  - No OPEN from the destination server to avoid locking issues
  - Like to READ with ca_src_stateid and the COPY SAVE_FH
- Source server needs to know the READ stateid is special
  - Do not verify stateid against the destination server clientid
  - Map ca_src_stateid to another stateid for use with the READ
  - OR…
Secure Inter-server SSC Choices

- Mike Eisler’s random # in file handle (draft-20)
  - Plus new operation COPY_STATE_REGISTER stateid issue fix
- GSSv3 (draft-19)
- SSV Solution
- Shared secret in COPY_NOTIFY, COPY, and COPY_STATE_REGISTER
Section 3.4.1.2. Inter-Server Copy via ONC RPC

COPY_NOTIFY replies with a list of destination server target addresses
- <random number, source fh, user ID, destination address>

The list is sent to the destination in the COPY operation

Destination choses one to set up the copy READ:
- COMPOUND { PUTROOTFH, LOOKUP "_COPY" ; LOOKUP <random #>; LOOKUP "203.0.113.56"; LOOKUP "_FH" ; OPEN "0x12345" ; GETFH }
Random Number in File Handle (draft-20)

- Authenticatedates the destination server
  - YES, via the random # in the file handle

- Destination READ special handling at source
  - NO, (Yes with FH, but destination server does it’s own OPEN)
    - client may already have established an exclusive lock on that file

- Act on behalf of the user-principal
  - NO (Yes with FH)

- Limit the destination server
  - NO, trust the destination server not to continue use of FH
Tom Hayne’s Stateid Issue Solution

- We only allow NFSv4.2+ as the copy-engine.

- We provide a new procedure which is sent from the destination to the source which presents the ca_src_stateid and returns a CSR-stateid that is valid for the destination
  - COPY_STATE_REGISTER cfg, ca_src_stateid

- CSR-stateid is then used for destination READs from the source
Random Number and Stateid solution

- Authenticates the destination server
  - YES, via the random # in the file handle
    - Destination server does COPY_STATE_REGISTER instead of OPEN

- Destination READ special handling at source
  - YES, CSR-stateid derived from ca_src_stateid

- Act on behalf of the user-principal
  - YES, via READ using CSR-stateid

- Limit the destination server
  - YES, close of source file by client should destroy CSR-stateid
A user (or client) generated shared secret plus user-principal info is distributed between the source and destination via RPCSEC_GSS3_CREATE calls

- A copy_from_auth privilege GSS3 context is used to send the COPY_NOTIFY to the source
- A copy_to_auth privilege GSS3 context is used to send the COPY to the destination
- A copy_confirm_auth privilege plus compound_auth GSS3 context is used for the destination READs from the source
RPCSEC_GSS3 (draft-19)

- Authenticates the destination server
  - YES, via the shared secret distributed via GSS3

- Destination READ special handling at source
  - YES, using the copy_confirm_auth GSS3 handle for READs

- Act on behalf of the user-principal
  - YES, via the use of compound authentication for the copy_confirm_auth GSS3 context handle creation

- Limit the destination server
  - YES, client destroys the copy_from_auth and copy_to_auth GSS3 context handles
COPY_NOTIFY returns an SSV secret generated at the source, which is sent to the destination in COPY

- Insist on privacy

Using the source SSV secret, setup an SSV GSS context between the destination and the source

- Distribute any other info needed for SSV setup in COPY_NOTIFY response and COPY arguments
- Use COPY SSV (and info) to setup dest/source session
- May require TBD changes to the NFSv4.1 SSV
Trond Myklebust’s SSV Solution

- Authenticates the destination server
  - YES, via shared SSV secret

- Destination READ special handling at source
  - YES, use of SSV GSS context handle

- Act on behalf of the user-principal
  - YES, use of SSV GSS context handle which is created as user

- Limit the destination server
  - NO, trust the destination server (no client action can stop the use of the SSV handle)
Distribute shared secret and use CSR

- Add a shared secret and the user-principal info used in GSS3 to NFSv4.2 COPY_NOTIFY and COPY operations
  - Insist on GSSv1 privacy
  - Source and destination have shared secret

- Add the shared secret and user-principal info to COPY_STATE_REGISTER
  - CSR-stateid then represents the ‘privilege’ to copy the file
Distribute shared secret and use CSR

- Authenticates the destination server
  - YES, via shared secret

- Destination READ special handling at source
  - YES, CSR-stateid derived from ca_src_stateid

- Act on behalf of the user-principal
  - YES, via CSR-stateid tied to shared secret and user

- Limit the destination server
  - YES, closing the source file removes ca_src_stateid and CSR-stateid
Secure Inter-server SSC Choices

- Each choice distributes a secret to source and destination
  - GSSv3 uses NULLPROC
  - All others use COPY_NOTIFY and COPY

- Each choice passes secret from destination to the source to signal normal READ as ‘special’ to the source
  - GSSv3 context handle
  - SSV GSS context handle
  - CRS-stateid
Secure Inter-server SSC Choices

- Mike Eisler’s random # in file handle (draft-20)
  - Plus new operation COPY_STATE_REGISTER stateid issue fix
    - YES, YES, YES, YES : Uses CSR-stateid to represent copy privilege

- GSSv3 (draft-19)
  - YES, YES, YES, YES : Uses GSS3 to represent copy privilege

- SSV Solution
  - YES, YES, YES, NO : uses GSS-SSV to represent copy privilege

- Shared secret in COPY_NOTIFY, COPY, and CSR
  - YES, YES, YES, YES : uses CSR-stateid to represent copy privilege
GSSv3 Pros and Cons

- **Pros**
  - It allows for NFSv3 as well as NFSv4.x for destination READs from the source.
  - Also used for LNFS full mode or server-guest mode labels

- **Cons**
  - GSSv3 draft needs more attention
  - NFSv4.2 use of GSSv3 needs review
  - NFSv4.2 use of GSSv3 is more complicated than other solutions
Thank you