



What's out there

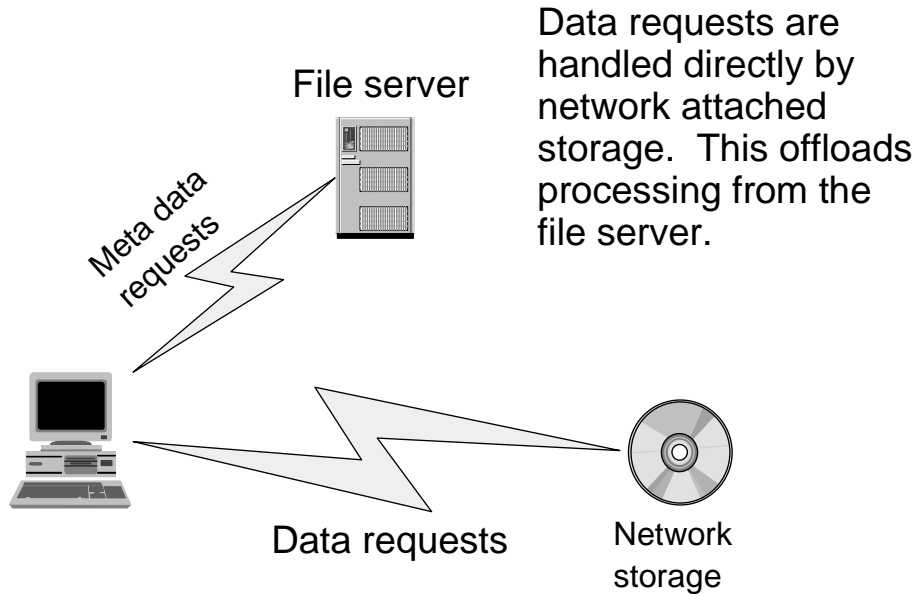
- NFS
 - Cannot span servers (automounters)
 - No security (secure NFS)
- CIFS (Windows file sharing)
 - similar to NFS
- AFS
 - Good scalability and security
 - Complex infrastructure
- HTTP
 - Basically read-only
 - No referential integrity



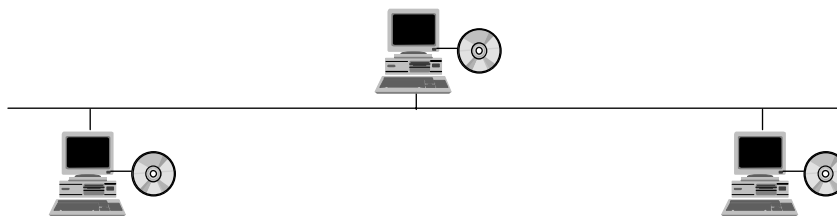
What is needed

- Referential integrity
- Security
- Unified name space
- Scalability
- Manageability

What has been done? (NASD)



What has been done? (XFS)

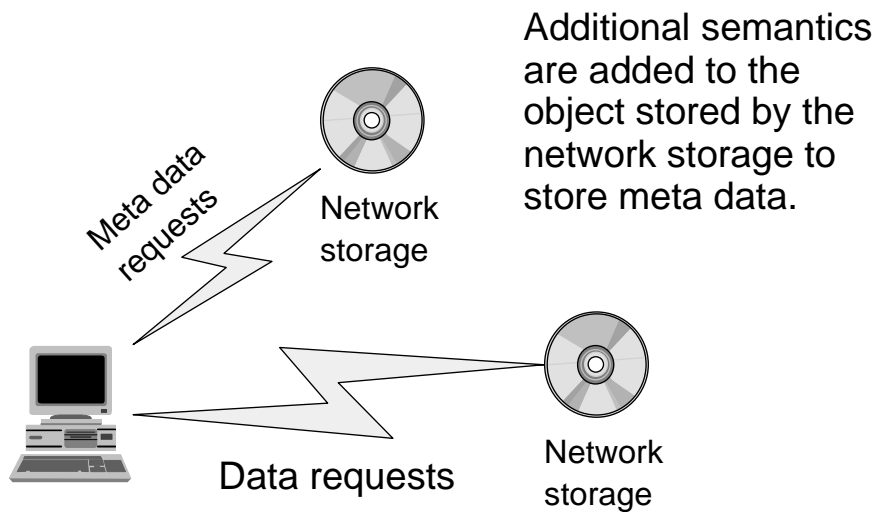


Clients expose local storage to the other clients. The file system is striped across the client storage. All clients are effectively servers for part of the file system

Our approach

- Allow the clients more participation in the meta data management of the file system. (Like XFS.)
- Add meta data semantics to the object model used by NASD.
- Use a well defined key derivation scheme to simplify key management and eliminate the need for encryption.

Our approach





Object semantics

- Attributes common to all objects
 - Each object has a unique id. The namespace is flat.
 - Each object has an information block and possibly an ACL associated with it.
- Data objects
 - Data is accessed as a contiguous stream of bytes.
 - Read/Write operations supported.
- Meta data objects
 - Data is organized into an array of entries.
 - Entries can be accessed using a lookup tag.
 - Read/Lookup/Delete entry/Change entry/Insert entry operations supported.



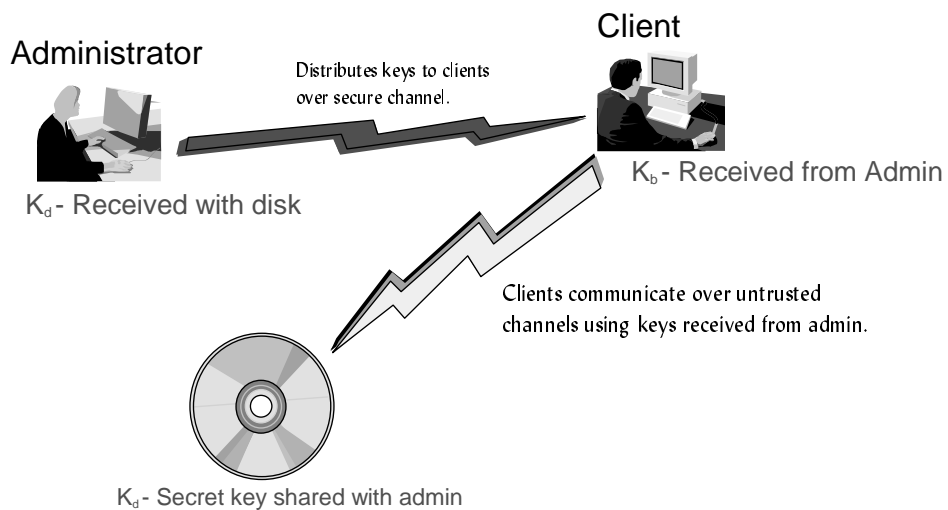
Scared security requirements

- Integrity – packets cannot be modified while in transit without detection.
- Identity – the receiver of a message can verify the key that was used by the originator of the message.
- Freshness – the receiver can know that the message received was not a replay of an earlier message.
- Confidentiality - an eves dropper cannot know the content of a message.

Assumptions

- Clients and administrators exchange keys using existing infrastructures (SSL, Kerberos, etc.).
- Clients communicate with storage over untrusted network.
 - Packets can be modified and addresses spoofed.
 - Both server and client must authenticate.
- Administrator is not assumed to always be online.
- No globally synchronized clocks. Local timers.
- Encryption is *not* done on the storage device.

Roles in a distributed file system



Restricting access

- Using identities
 - Clients prove their identity to the network storage.
 - Network storage grants access based on ACL.
- Using capabilities
 - Clients prove their access rights by presenting credentials received by someone in possession of access rights.
 - No need for ACLs.

Message Authentication Codes

- Rely on shared MAC key (k)
- Provide identity and integrity
- The function $MAC_k(\text{message})$ produces a fixed length string based on k and the message
- No Freshness Guarantee

Calculates $M = MAC_k(\text{message})$



message, M



Calculates $MAC_k(\text{message})$
 $MAC_k(\text{message}) = M ?$

If equal, the sender was is possession of k.



Key derivation not key exchange

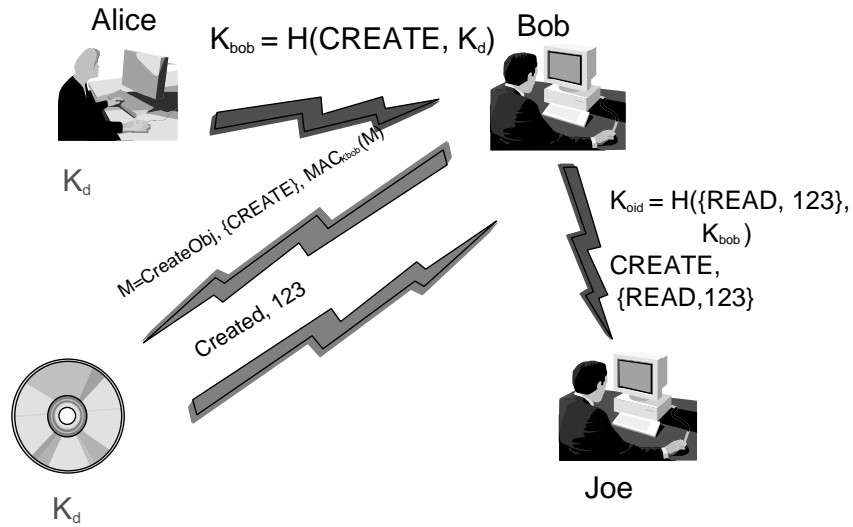
- No encryption on the storage device rules out key exchange.
- Based on keyed one-way hashes (psuedo-random functions) $H(D,K)$ where D is public and K is secret.
- Derivation method is well known.
 - $K' = H(D,K)$
 - D is the public key data associated with the new key and K is the key from which K' is derived.



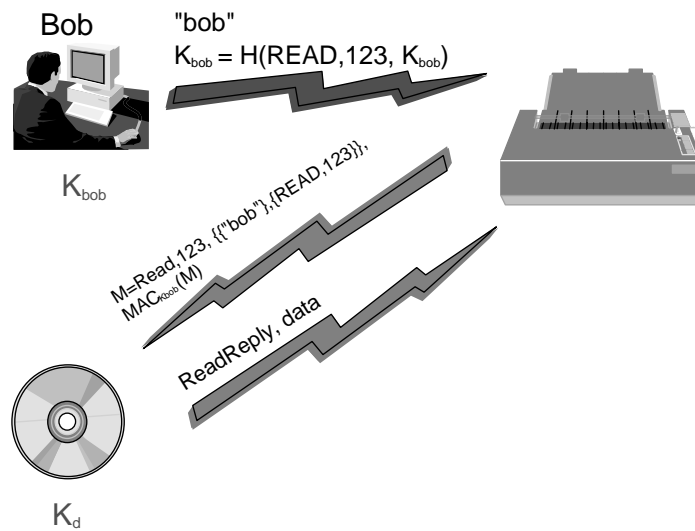
Uses of the key data

- The data associated with the key is bound to it.
 - The key can only be used with its associated data. (Changing the data changes the key.)
- SCARED structures the data to contain information about the key.
 - Expiration time is used to limit the life of the key.
 - Capability information is used to restrict what the key can do.
 - Identity information is used to identify client (group or user identification)

Example key derivation



Example mixed key derivation





Client freshness guarantee

- A nonce is a 128-bit number generated by the client.
- Does not need to be random, just unique.
- A new nonce is generated for each request and checked in each response.
- The storage device copies the nonce from the request to the reply.
- The presence of the nonce binds the request to response (avoids replays of earlier responses).



Server freshness guarantees

- When using session oriented communication
- The device gives the client an initial counter.
- With each request, the client increments the counter and includes it in the request.
- The device checks that the counter is monotonically increasing.



What to do about encryption

- SCARED addresses authentication, not confidentiality. Confidentiality when needed can be addressed two ways.
- Encrypt at the client.
 - Would work well with CFS.
 - Encryption key distribution/revocation needed.
 - Some meta data not encrypted.
- Derive a session encryption key based on the shared key.
 - Increases the processing at the device.
 - Requires encryption.



Summary

- SCARED is simple to administer.
- Provides strong authentication guarantees.
- Doesn't require encryption at the device.
- Key exchange among clients and administrators can be done using existing infrastructure.