Vanadium (V23) Discussion

Marc Mosko
marc.mosko@sri.com
ICNRG, IETF 119, Brisbane
Wednesday 20, 2024
What? Why?

- It is a secure, distributed, RPC system
- It has distributed naming and discovery
- It uses symmetrical authentication and encryption
- It may use private name discovery with IBE

- It seems interesting for ICN.
- We will do a quick intro to Vanadium, then discuss.
Vanadium has two parts

• Principals and Blessings and Caveats [Security]
  • Use a hierarchical name, e.g. alice:home:tv.
  • Certificate based
  • Blessings are scoped delegations from one principal to another for a namespace (e.g. alice grants Bob “watch” permissions to the TV)
  • Caveats are restrictions on delegations (e.g. Bob can only watch 6pm – 9pm).
  • 3rd party caveats must be discharged before authorization
    • E.g. revocations or auditing

• The RPC mount tables [Object Naming]
  • These describe how to locate RPC namespaces
  • They provide relative naming
How ICN-ish is Vanadium

• The security part is pretty ICN-ish
  • It uses prefix matching and encryption
  • Namespaces work like groups
  • The colon : separates the blesser from the blessed
  • Authorizations match extensions.
    • If Alice authorized “read” to alice:hometv to alice:houseguests, and if Bob has a blessing for alice:houseguests:bob, then Bob has “read” to alice:hometv.
  • A special terminator :$ only matches the exact prefix.
    • A blessing to alice:houseguest:$ only matches that exact prefix.

Examples from https://vanadium.github.io/concepts/security.html
Examples

• Alice authorizes her TV (Ptv) within her namespace.

\[ \text{P}_{\text{alice}} \text{ using name alice says that P}_{\text{tv}} \text{ can use the name alice:devices:hometv} \]

• This caveat is valid if server has a blessing as alice:devices:hometv.

\[ \text{P}_{\text{alice}} \text{ using name alice says that P}_{\text{bob}} \text{ can use the name alice:houseguest:bob as long as server matches alice:devices:hometv} \]

Examples from https://vanadium.github.io/concepts/security.html
Root of trust

• Vanadium skirts the issue of root of trust.

• They ran an identity service at https://dev.v.io/auth
  • Based on an initial OAuth2 IDP of an app for an email address
    • User-blessing: dev.v.io:u:<email>
    • App-blessing: dev.v.io:o:<appid>:<email>
    • Authorizes an app to work on behalf of a user within some context in the blessing

• They ran a Discharge service
  • The identity service offered a revocation caveat at dev.v.io/users/<email_address>
  • Chose a revocation rather than a short-lived authorization.

https://vanadium.github.io/design/docs/identity-service.html
Part 2: Object Naming

- An object name facilitates an RPC

If the name /host:8080/a/y/foo.jpg represents a JPEG file, then /host:8080/a/y/foo.jpg.Get() will return the contents of that file.

- Note that Object names are/

- A Mount Table maps an endpoint to object name components.

- Resolving an object name is a recursive walk.

- It is relative. Client 1 sees /ns1.v.io:8101/a/y, client 2 sees /ns2.v.io:8102/y

https://vanadium.github.io/concepts/naming.html
Entity resolution

• The devices that execute an RPC name are also listed in the mount table.
• E.g. /ns1.v.io:8101/a/srv.Get()
• Extensions are passed
• Cycles are permitted

https://vanadium.github.io/concepts/naming.html
More on names

• A mount table is a service that maps an object name to an (endpoint, suffix) pair.
• A mount table is itself an RPC, so you can mount tables on tables.
• A rooted name begins with an Endpoint
  • <ipv4:port>,
  • <ipv6:port>,
  • <dns:port>, or
  • <macaddr> (for BT)
Summary

• Vanadium is a permissioned RPC service

• A name encodes the endpoint plus name suffix

• The endpoint does not need to resolve to a single mount table server, it could be any server that possesses an appropriate blessing.

• Authentication is done via pair-wise key exchange and blessing validations. It can be private if using IBE, otherwise server name leaks.

• Authorizations and Blessings and Caveats use hierarchical, prefix-matching names.
What does it mean for ICN?

• The security piece is interesting.
  • Blessings and Caveats and discharges and namespaces as groups.
  • How does it differ from SDSI co-signings?

• Vanadium identity service
  • Interesting mapping of OAuth2 app:email tokens to PKI and blessings.

• The RPC piece needs a bit of finessing
  • Embedding an endpoint in the name is not ICN.
  • Embedding a root of trust could be.
  • CCNx [1] uses public-key scoped names.
  • NDN uses schematized-trust key anchors.