A Status Report on USEC:
The User-based Security Model for SNMPv2

Glenn Waters

Bell-Northern Research Ltd.

gwaters@bnr.ca

A Brief History

- Posted to SNMP (not SNMPv2) list
- Provides same security properties as SNMPv2 (“classic”)
  - but with a considerably reduced “footprint”
- Simplification has its advantages:
  - 90% of the functionality at half the cost
  - no longer any requirement for an NV-clock
  - single naming model (user names)
- Simplicity has its disadvantages:
  - (local) strings used to identify users, contexts
  - relationships are symmetric

USEC and the SNMPv2 Working Group

- Admin Model re-written to:
    - define contexts, access rights and views
    - remove parties
    - refer to “security models”
- USEC document revised for SNMPv2
- USEC remote configuration MIB created:
    - replaces party MIB
    - although important and needing timely resolution, not lock-stepped with other documents
- Non-admin documents revised to genericize admin references
- Two openly-available implementations published
- Although the SNMPv2 WG didn’t agree on security, the work on USEC continues...

The Original USEC Idea

- Separate the party concept into users and agents:
  - users have secrets and access rights
  - agents have transport information and clocks
- The dichotomy vastly simplifies implementation:
  - one clock per agent, etc.
- Easy, intuitive configuration based on user-identities

USEC Re-design

- Technical design team formed to address comments from WG (and others):
  - Bert, Glenn, Keith, Marshall, and Shawn
- Design team format chosen instead of “open” forum:
  - “design by committee” doesn’t work very well; and,
  - USEC already close to complete
- Now doing implementation testing to examine some issues in revised documents

Clarifications to the Administrative Framework

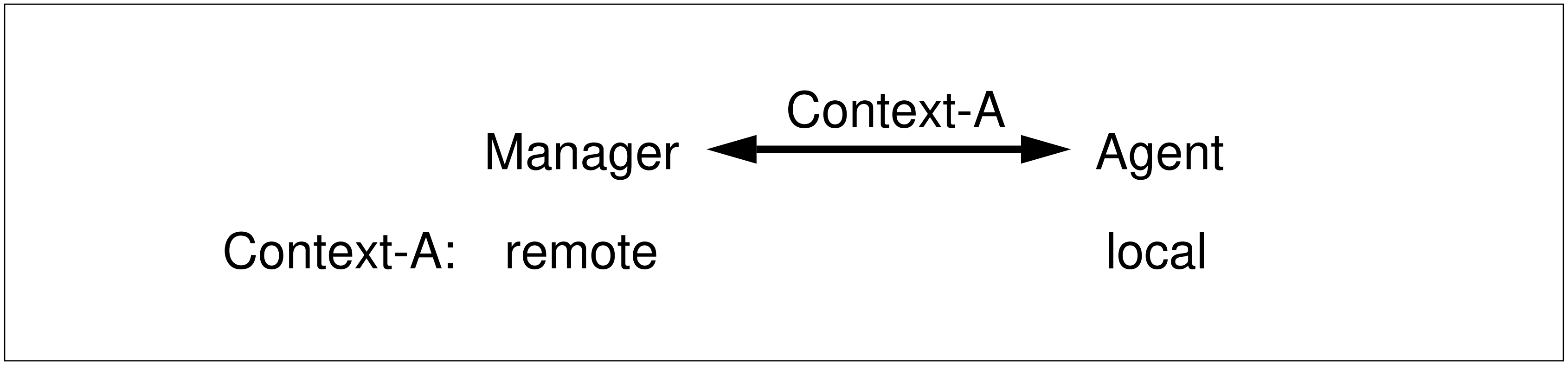
(Informs, Agents, and Managers)

- SNMPv2 is ambiguous in this area
- Since only agents can have managed objects,
  - the information in an inform <u>must</u> come from the agent portion of a dual-role entity
- Corollary: if you generate an inform with data,
  - you have to be able to respond to requests for data, since, informs, like traps, are unreliable
  - upstream manager may need to retrieve values later
- This is consistent with the notion that the owner of the data also owns the clock

Changes to the User-based Security Model: Contexts

- Contexts are uniquely identified by one (or more) *agentID* and *contextSelector* pair(s), hence, *unknownContexts* reports when agentID wrong
- Four kinds of contexts:
  - remote: used by a manager
  - local: used by an agent
  - local-proxy: received by a proxy-agent
  - remote-proxy: forwarded by a proxy-agent
- Rules added for proxy error propagation and timeouts

Non-Proxy Context Scenario



Proxy Context Scenario

Manager $\xleftrightarrow{\text{Context-B}}$ Proxy $\xleftrightarrow{\text{Context-C}}$ Agent

Context-B:	remote	local-proxy	- - -
Context-C:	- - -	remote-proxy	local

- B and C are two identities for the same context

Changes to the User-based Security Model: Error Reports

- QoS bit indicates whether reports may be generated; however:
  - reports may never be generated on bad ASN.1 or parameters
- *notInWindow* reports have same authentication QoS as requests but are never encrypted
- To discover information about an agent:
  - send a noAuth/noPriv packet for user “public” with a zero-valued *agentID* and an empty *contextSelector*

  you’ll get back a *unknownContexts* report,
  - and the parameters tell you about *agent**
  - optionally, you can then perform authenticated clock sync
- Password to Key algorithm now “localized” by *agentID*
- Keyed-MD5 strategy used for digest calculation

Where Next?

- Look for USEC revision by Spring, 1996
  - based on SNMPv2 WG's current document set
  - new Admin and USEC documents
  - Agent and manager code, commercial and otherwise
- Hope for IWL testing event thereafter