Drinking from the Firehose: Multicast USENET News

Kurt J. Lidl
Josh Osborne
Joseph Malcolm

Email: musemasters@uunet.uu.net

UUNET Technologies, Inc.
3110 Fairview Park Drive, Suite 570
Falls Church, VA 22042 USA

Voice: +1 703 204 8000
Fax: +1 703 204 8001

Overview

- *Muse*
- **Background and History of USENET**
- **Motivations**
- **Technologies Used**
- **Overview of Operation**
- **Traffic Analysis**
- **Future Directions**
- **Conclusions**

Muse

Muse **runs on top of the Mbone**

Muse **is a unreliable article transfer system**

Muse **must be used with a traditional news system to provide reliable news service**

Muse **detects the injection of forged articles into the multicast news stream and rejects them**

Muse **provides a way of matching the flooding model of USENET article propagation to a datagram network that has similar characteristics**

Background and History of USENET

Basic Concepts:

- **Articles**
  - Fundamental unit of transmission on USENET
- **Newsgroups**
  - Grouping mechanism for news articles
- **Distributions**
  - A way of limiting the scope of an article
- **Transmission**
  - News-name of a Site
  - Flooding of Articles
  - Duplicate Suppression

Background and History of USENET (cont.)

History:

- **Transport Agents**
  - UUCP Feeds/Batches
  - NNTP
- **News Systems**
  - B news
  - C news
  - INN

Motivations

- **Reduce Article Propagation Times**
  - 120ms coast-to-coast
  - Direct transmission, not store and forward

- **Allow for an Unlimited Number of Listeners**
  - Scale as well as Mbone
  - Centralized resource usage scales sub-linearly

- **Reduce Network Traffic**
  - More efficient article flood-fill mechanism
  - Avoid replicated data streams

- **Encourage the Use of the Mbone**
  - Popularity brings stability to the Mbone

Technologies Used

- **The Mbone**
  - Experimental Multicast Backbone
  - Transport mechanism for IP Multicast packets
  - *RFC 1112, RFC 1075*

- **NNTP**
  - Network News Transfer Protocol *(RFC 977)*
  - Already widely deployed on Internet
  - Little modification necessary for listeners

Technologies Used (cont.)

- **RSA** *rsaref* **library**
  - Public Key Cryptography System
  - RSA has slower signing than DSS
  - RSA has faster verify than DSS
  - *Muse* uses a 512 bit key, could use longer keys
  - Already twice as fast as distributed code
  - Signing entire article directly impractical

- **MD5**
  - Message Digest Checksum (*RFC 1321*)
  - Fast, cryptographically secure checksum
  - O(2ˆ128) to forge an article
  - O(2ˆ63) to create two messages that produce the same hash

Overview of Operation

- **On the Transmitter:**
  - Accept article via NNTP
  - Timestamp article (*t1*)
  - Calculate MD5 checksum
  - RSA sign checksum
  - Multicast signed article
  - Log *t1*, message-id and current time (*t2*)

Overview of Operation (cont.)

- **On the Listener:**
  - Receive multicast article
  - Get timestamp (*t3*)
  - Calculate local checksum
  - Decode transmitter’s checksum
  - Compare checksums:
    - Don’t match – discard article
    - Match – offer article via NNTP
  - Log timestamps (*t1*, *t3*), message-id, and NNTP reply code

Future Directions

- **Faster RSA Signing**
  - Investigate hand-coded assembly loop
  - RSA hardware for PCs
  - Would allow for full newsfeed to be multicast

- **Increased Multicast Selectivity**
  - All news travels directed at a single address
  - Multicast news hierarchies on different addresses, e.g. comp, rec
  - Cross-posted articles
  - Allows pruning on Mbone to limit scope of packets

- **Richer Control Messages**
  - Currently can only send ‘version’, ‘news’ and ‘NOP’ messages
  - Measure average propagation through network

Future Directions (cont.)

- **Better Receiver**
  - Receiver cannot deal with many control messages
  - Cleaner mechanism to feedback control results

- **Add Compression**
  - Currently limited to about 8900 bytes of payload
  - Approximately 40Kbytes of raw article with ‘gzip’ compression
  - Allows larger percentage of articles to be eligible for multicast

- **Batches**
  - Multiple articles in each transmitted packet
  - Only need to RSA sign the entire ‘batch’
  - Helps work around the RSA signing bottleneck

Conclusions

- Concept Works
- Scales as Well as the Mbone Scales
- Handles Very Large Percentage of Articles
- Can be Extended to Handle Even More Articles
- Reduces Load on Backbone Links

Acknowledgments

– UUNET Technologies
    - For allowing us to develop this on their machines
– Dennis Ferguson
    - NTP timestamp code
– Mbone Implementors
– RSA Laboratories
    - rsaref library of functions
    - MD5
– David Hsu
    - Title of this paper
– Debbie Greenberg
    - Proofreading
– Nat Howard
    - Cryptographic Q & A

Muse – Something you *need* to have!

Availability:

Anonymous FTP from host ftp.uu.net

/networking/news/muse/muse-client.tar.Z

Email:

musemasters@uunet.uu.net