NC STATE UNIVERSITY Computer Science

Iolus: A Framework for Scalable Secure Multicasting

Vladica Stanisic

Multicast

- Efficient means of distributing data to a group of participants
- Supports one to many and many to many service
- Needs to transmit just one copy of data
- Supports dynamic membership

Multicast Vulnerabilities

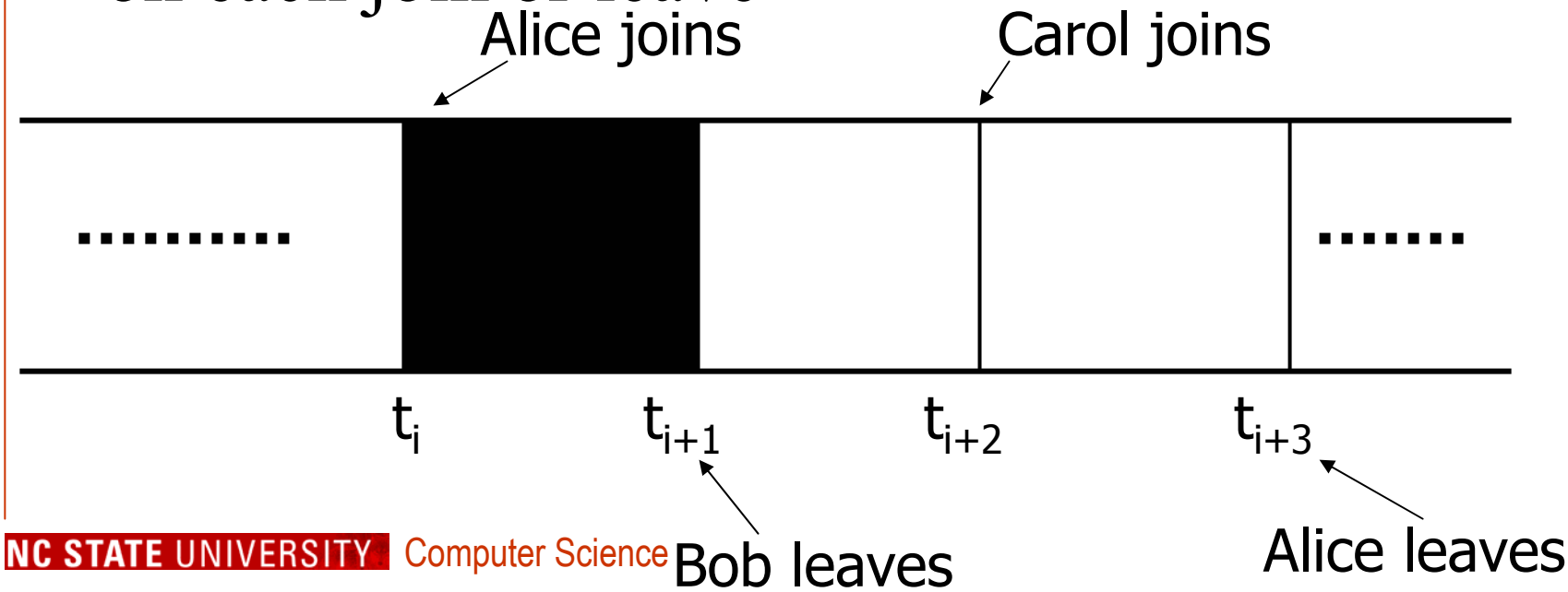
- More opportunities for interception of traffic
- Sessions are frequently advertised
- Attack affects a broader base of people
- Attacker can pose as a legitimate user easier (larger “crowd” of principals)

The Iolus Objectives

- It can secure arbitrary multicast transmissions
- It can be used to implement common key management service
- It can be used to provide a group key management service to unicast appliactions

Network Security: Adding Multicast

- Treat a multicast session as blocks of time
- A new block begins when someone joins/leaves the group
- The security associations need to be changed on each join or leave



The Scalability Problem

- "1 affects n" type failure
  - the actions of one member affects the entire group
  - A new source-based delivery tree rooted at the sender needs to be created whenever a new sender joins the multicast group
- "1 does not equal n" type failure
  - occurs when protocol cannot deal with the group as a whole and must consider the conflicting demands of members on an individual basis

“1 affects n” Example

- When a new user joins the group, if the entire group shares a single group key (K_{grp}), a new key must be generated ($K`_{grp}$) and distributedto the current members as well as the joining member.

“1 does not equal n” Example

- When user leaves, we need to replace K_{grp} with $K`_{grp}$, but we cannot distribute using K_{grp}
- The new key needs to be transmitted to each user individually
- Extremely inefficient in large groups or highly dynamic memberships

Additional Notes

- All receivers must get a K_{grp} update, or they will not be able to decrypt later communication, and may also accept communications from members that have been removed from the group
- Senders failing to receive a K_{grp} update will continue to encrypt using an outdated key resulting in legitimate users that are unable to decrypt the transmissions and allowing former group members to decrypt the transmissions

Additional Notes (cont.)

- If the underlying multicast system is unreliable, additional processing will be required through the use of a reliable multicast protocol
- In highly dynamic memberships due to flurries in key update messages, the probability of confusion and security breaches increase

Design Features and Requirements

- Scalability
- Robustness
- Security Objective Independence
- Security Technology Independence
- Communication Protocol Independence
- Protocol Layer Independence

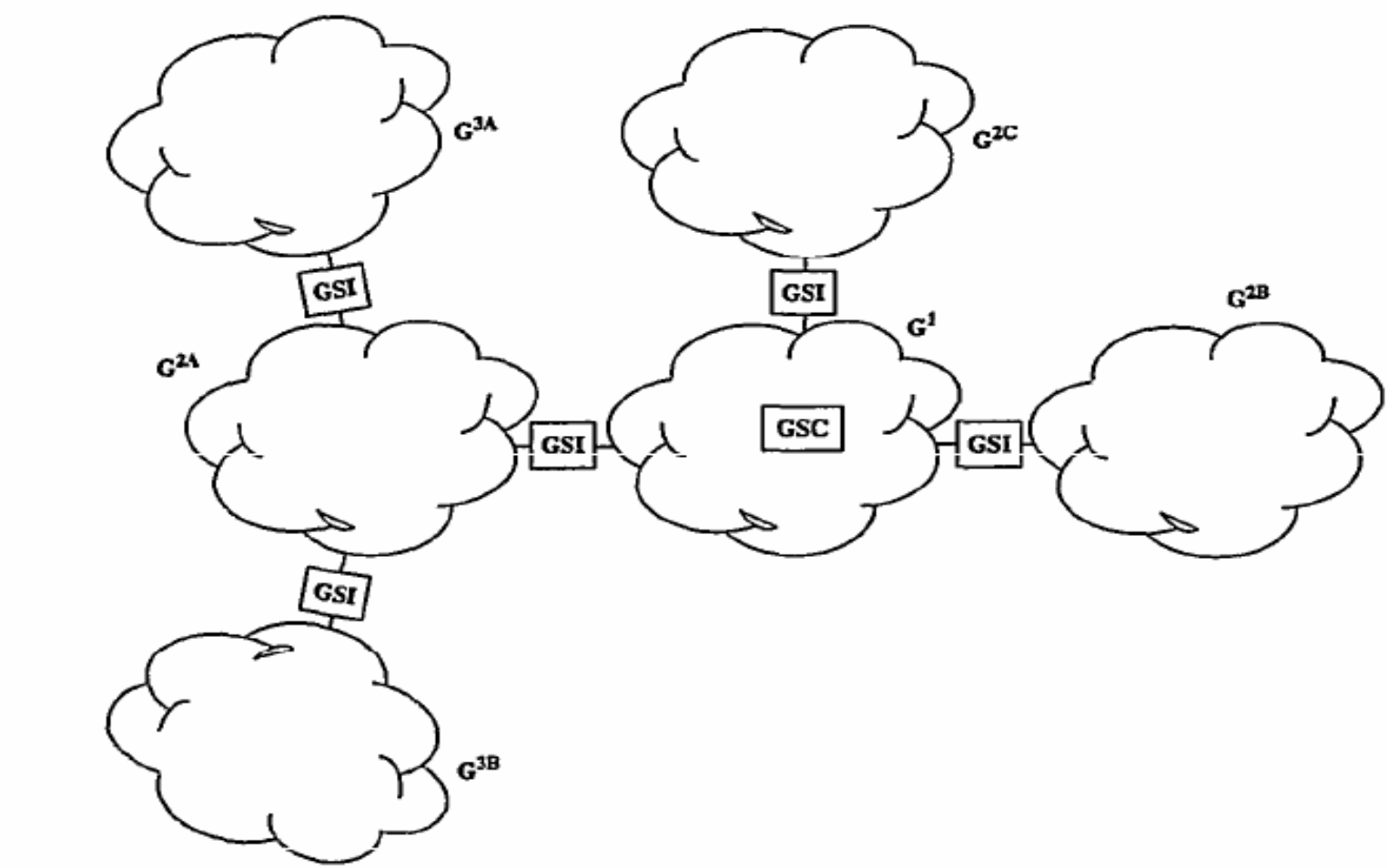
The Iolus Framework

- Removes the concept of a flat secure multicast group
- A secure distribution tree is created composed of subgroups
- Each subgroup has its own multicast group (with its own address)
- Each group has its own subgroup keying material (K_{sgrp})
- Only the local K_{sgrp} needs to be changed on a join or leave

Iolus Entities

- Group Security Controller (GSCs)
  - Manages the top level subgroup
- Group Security Intermediaries (GSIs)
  - Manage each of the subgroups
- GSCs and GSIs are called Group Security Agents (GSAs)

Iolus Secure Distribution Tree



Functions of the GSAs

- The GSC maintains control of the top-level subgroup at the root of the secure distribution tree
  - Ultimately responsible for the security of the group
- GSIs are special trusted servers that are authorized to act as proxies of the GSC or their parent GSIs and control their local subgroup

GSIs

- Grouped according to levels within the secure distribution tree
- They form a bridge between subgroups by receive data multicast in their parent or child subgroups and re-multicast to their child or parent subgroups respectively

Iolus Operational Overview

- Startup
- Joins
- Refreshes
- Leaves
- Data Transmissions
- Re-keying

Startup

- GSC is supplied with an access control list (ACL), which is used to set the security policy concerning user access
- GSIs and other members apply to join its subgroup

Joins

- A sender or receiver locates its designated GSA and issues a JOIN request using a secure unicast channel.
- GSA checks its database and decides whether to approve or deny request
- GSA generates a secret key ($K_{GSA\text{-}MBR}$) to be shared only with the new member
- Stores secret key along with other relevant information in a private database

Joins (cont.)

- Sends $K_{GSA\text{-}MBR}$ to member using the secure channel
- GSA multicasts a GRP_KEY_UPDATE message containing $K`_{sgrp}$ encrypted with K_{sgrp} to current multicast subgroup
- Sends $K`_{sgrp}$ to the joining member using the separate secure channel
- If the GSA is not currently part of the secure multicast group, it follows a similar procedure to join its parent subgroup

Refreshes

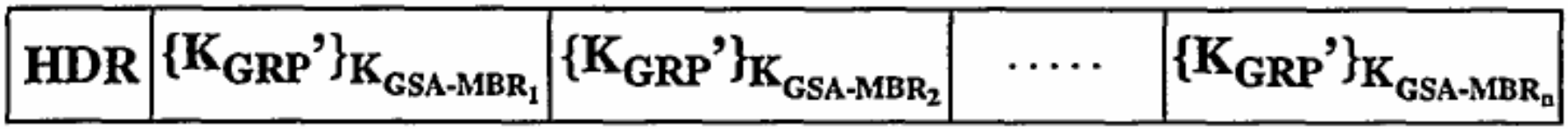
- Each JOIN has an expiration time associated with it
- If the member wishes to remain in the group, it sends a REFRESH message
- Allows the framework to gracefully handle network partitions
- Subgroups that have become partitioned from the top-level subgroup are implicitly removed from the group after some threshold time has elapsed

Leaves

- Occur under two conditions
  - A member wishes to voluntary leave (sends a LEAVE request to the GSA)
  - GSA wishes to expel member (sends a notification to expelled member)
- Either way, K_{sgrp} needs to be changed
- GSA multicast new $K`_{sgrp}$ to the remaining group members

Leaves (cont.)

- Multicast one message containing n copies of $K`_{sgrp}$
- Use separate $K_{GSA\text{-}MBR}$ to encrypt $K`_{sgrp}$



Data Transmission

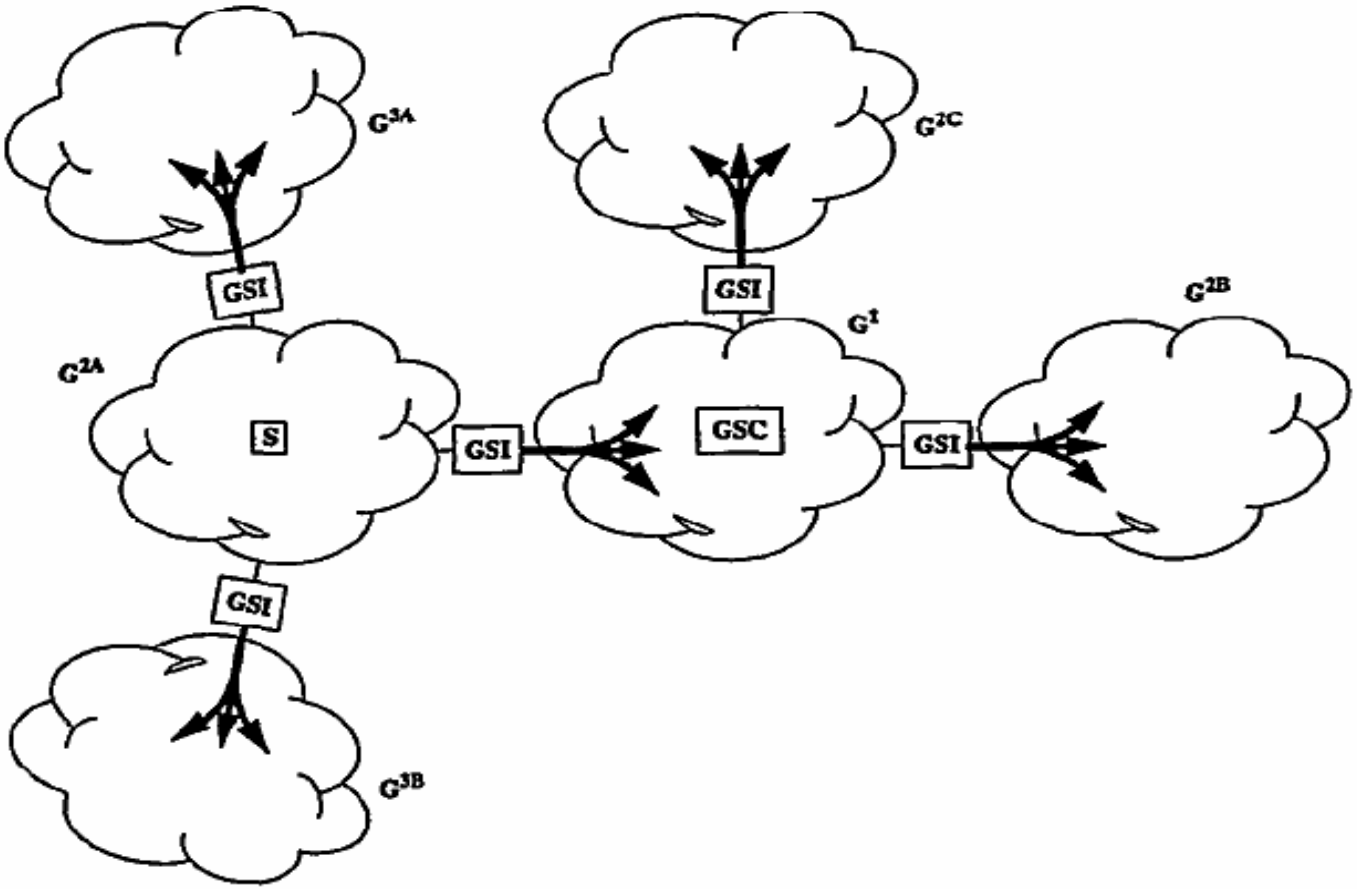


Figure 5: Data multicast in group G^{2A} is re-multicast throughout the secure distribution tree

Data Transmission (First Method)

- Instead of encrypting message with K_{sgrp}, sender generates a one-time random key and encrypts data with this key
- Encrypts key with K_{sgrp} and includes it with data
- GSAs only need to decrypt and re-encrypt the random key.

Data Transmission (Second Method)

- Sender *unicasts* message to GSA using $K_{GSA\text{-}MBR}$, GSA decrypts, re-encrypts with K_{sgrp} and signs packet, then multicasts to group and parent subgroup
- Eliminates the possibility of senders having an outdated K_{sgrp}
- Receivers have the assurance of GSAs signature to verify message is from a valid source

Data Transmission Comparison

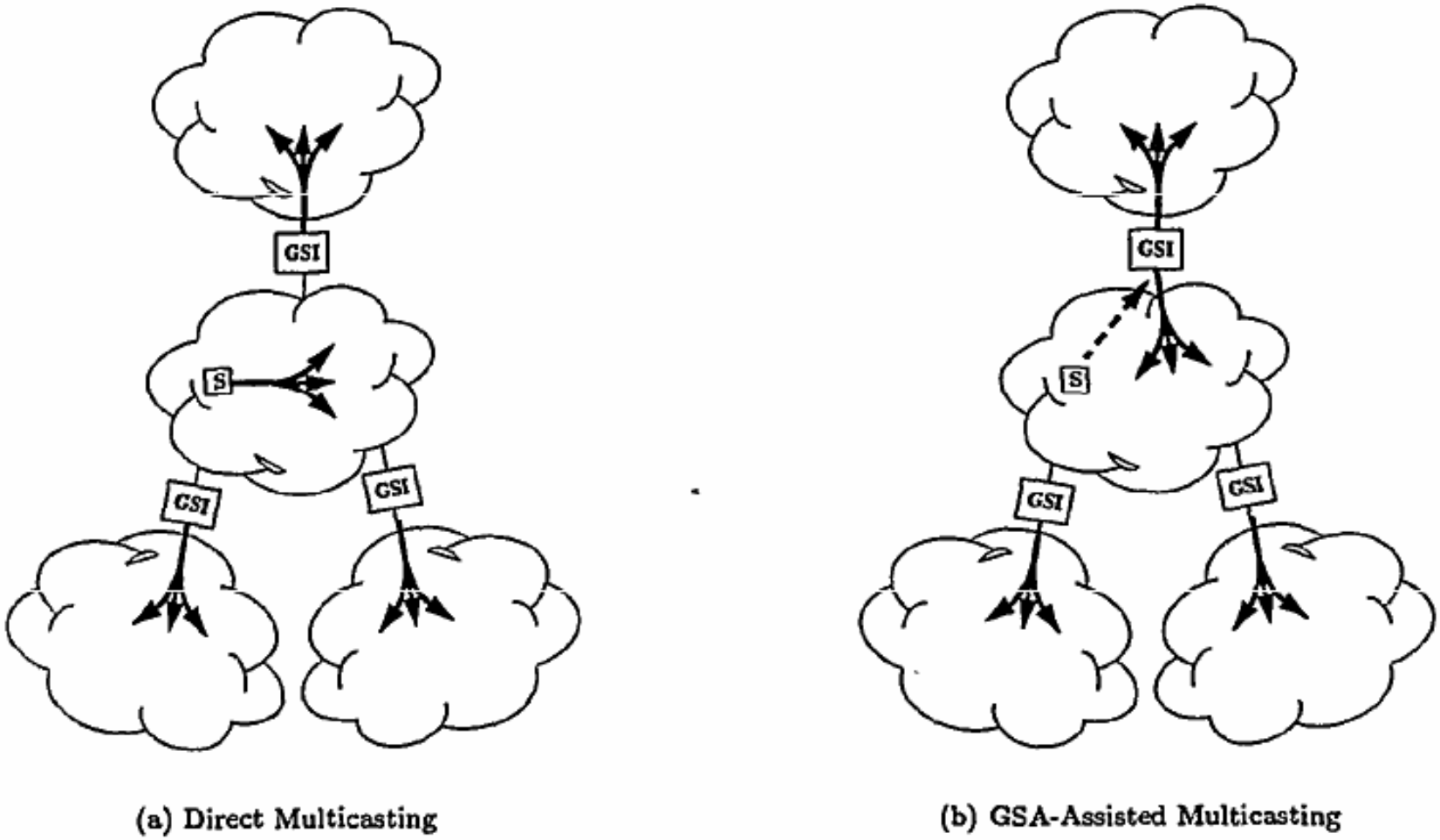


Figure 6: Two Methods for Sending

Re-keying

- Scheme used for changing K_{sgrp} on eaves to apprise the group members of $K`_{sgrp}$

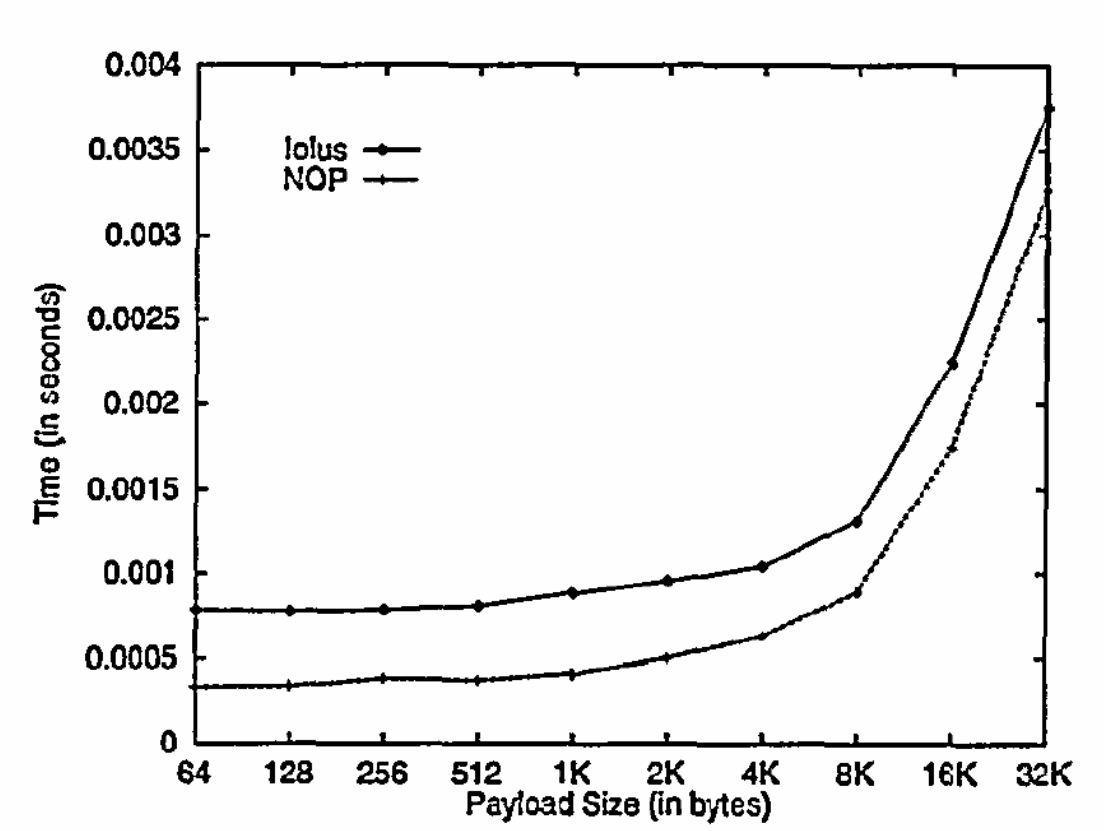
Shutdown

- GSC for secure multicast group shuts down after sending GRP_END notification to all members in the top-level subgroup
- Any GSAs attached to top-level subgroup will then multicast the GRP_END message and then shutdown and so-on.

Deployment Issues

- Secure Distribution Tree Management
- Secure Distribution Tree Construction
- Secure Distribution Tree Discovery
- Admission Control

GSA Multicast Forwarding Performance



- Approximately 450 micro-seconds delay introduced (due to cryptographic operations)

Iolus Benefits

- Achieves Scalability
- Enhanced Security
- Flexible Management
- Flexible Pricing