Key Management for Large Dynamic Groups: One-way Function Trees and Amortized Initialization*

<draft-balenson-groupkeymgmt-oft-00.txt>

IRTF SMUG Meeting
March 15, 1999

David M. Balenson, TIS Labs at Network Associates, Inc.
David McGrew, Cisco Systems, Inc.
Alan Sherman, Univ. of Maryland, Baltimore County

*Work performed as part of the Dynamic Cryptographic Context Management (DCCM) project sponsored by the DARPA High Confidence Networks (HCN) Program, Contract No. F30602-97-C-0277

TIS Labs at network ASSOCIATES

Presentation Outline

- DCCM goals / objectives
- Multi-party applications
- Multi-party application phases
- Group key establishment & re-key methods
- One-way function trees
- Amortized group initialization
- Multi-party application policy
- Concluding remarks

DCCM Goals / Objectives

- Security management for very large, dynamic multi-party applications
  - military command and control, collaborative computing, war gaming, and conferencing w/ up to **100,000** group members
- Dynamically specify/negotiate security policies via cryptographic context
  - managers, participants, security services, crypto mechs, etc.
- Dynamically establish application groups, group keys, change keys as group membership changes
- Develop and distribute prototype software toolkits and conduct demos

Multi-Party Applications

- DCCM application is a managed event requiring secure group communication
  - may be divided into sessions (e.g., secure lecture series)
  - driven by a specific policy controlled by application manager
- Groups have managers, members, and subgroups
  - member / subgroup participation changes dynamically
- Group operations include:
  - start/end session
  - member / subgroup join session, freeze/thaw access to session, leave session, evicted from session

Multi-Party Application Phases

- Application/group/session initialization
  - specify/negotiate security policy -- crypto context
- Group member initialization
  - authenticate public keys -- X.509v3 certificates, secure DNS
  - establish pairwise keys -- IKE
- Group key establishment & re-key
  - SKDC, Group DH, LKH, or OFT
- Group communications
  - confidentiality -- IPSec, TLS, or application layer
  - authentication -- MAC for group, signature for individual

Group Key Establishment and Re-Key

- Simple linear methods -- Simple Key Distribution Center (SKDC)
  - scale poorly, but attractive for small to moderate groups
- Information-Theoretic
  - require exponential member space
- Group Diffie-Hellman (GDH)
  - require slow public-key operations (linear)
  - attractive for small groups when distributed control needed
- Distributed, fault-tolerant systems; dynamic VPNs
  - scale poorly to large groups, but fault-tolerant aspects applicable
- Hierarchical -- LKH, OFT
  - scale best to very large groups

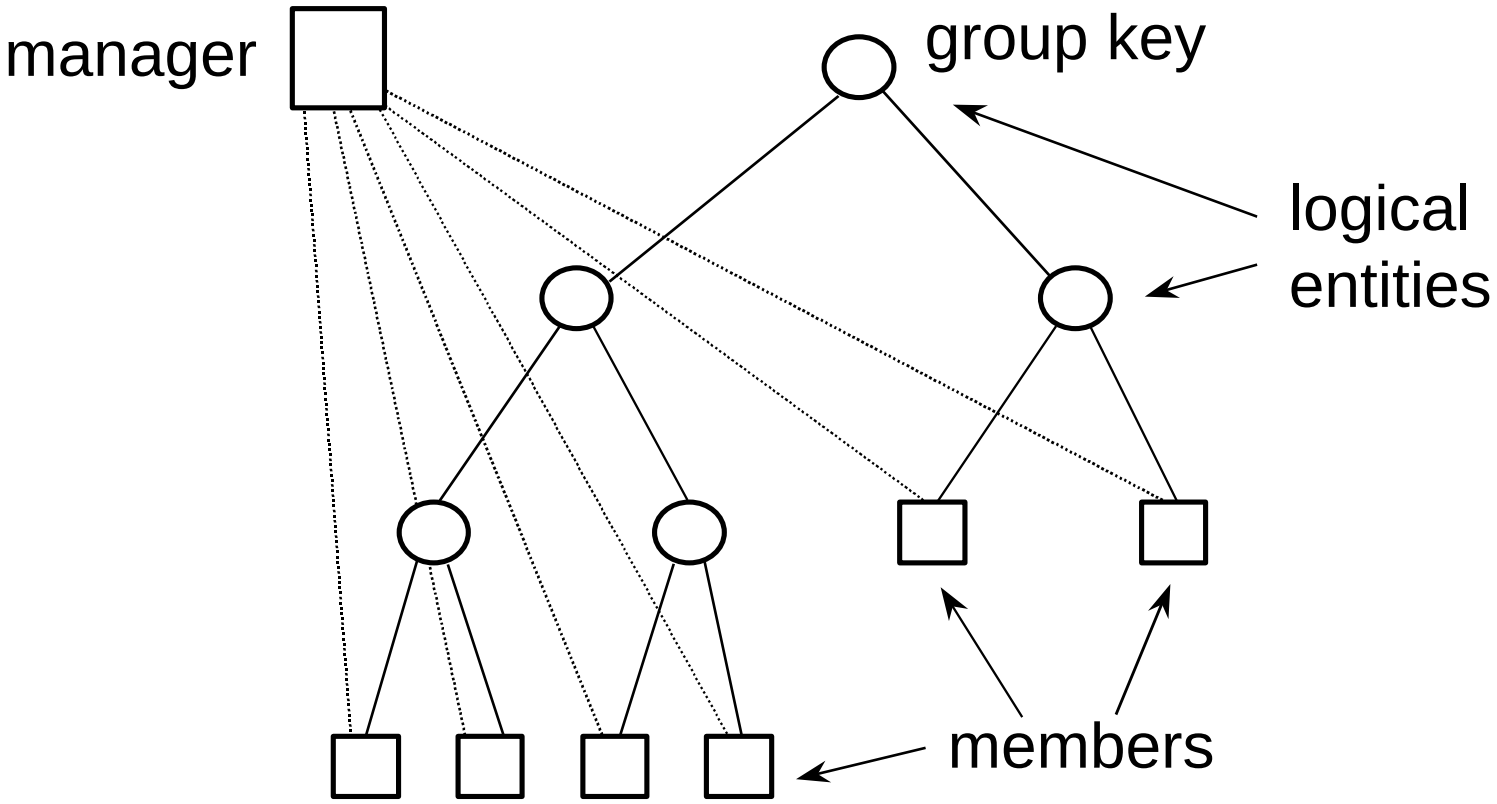
Logical Key Hierarchy (LKH)

- Wallner, Harder, Agee, NSA, 1997
- Update keys via encrypting node keys down a tree with members at the leaves
- Time, space, broadcast scale logarithmically
- Simple security properties

One-way Function Trees (OFT)

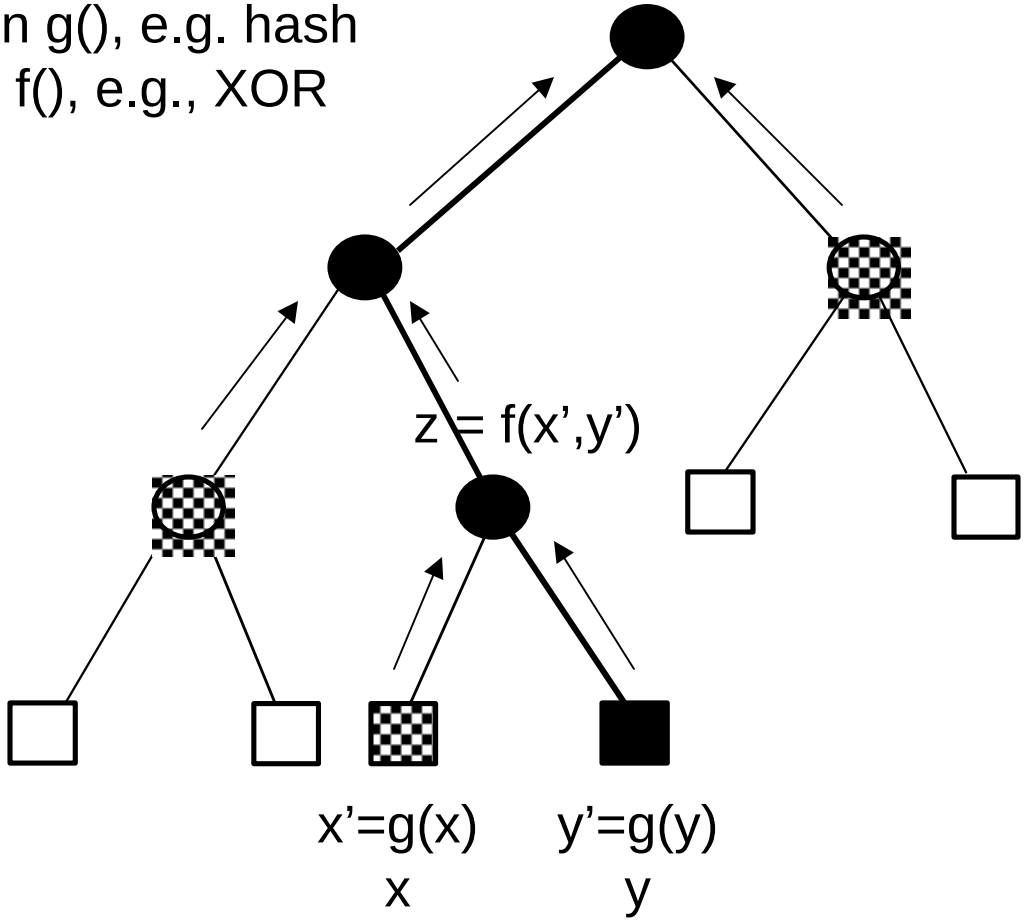
- McGrew, Sherman, TIS, 1997
- Novel application of one-way function trees
- Reduces number of broadcast bits for re-key

Hierarchical Model



One-way Function Tree (OFT)

blinding function g(), e.g. hash
mixing function f(), e.g., XOR



OFT Features / Advantages

- Lowest number of bits transmitted for re-key
- Use of fast cryptographic functions
  - hash for key derivation
  - symmetric encryption for OFT component distribution
- Scalability (growth is O log n)
- Applicable for 1:M or M:M groups
- Forward and backward security
- All members contribute keying material to group key
- Easily accommodates sub-groups
- Amortized group induction costs

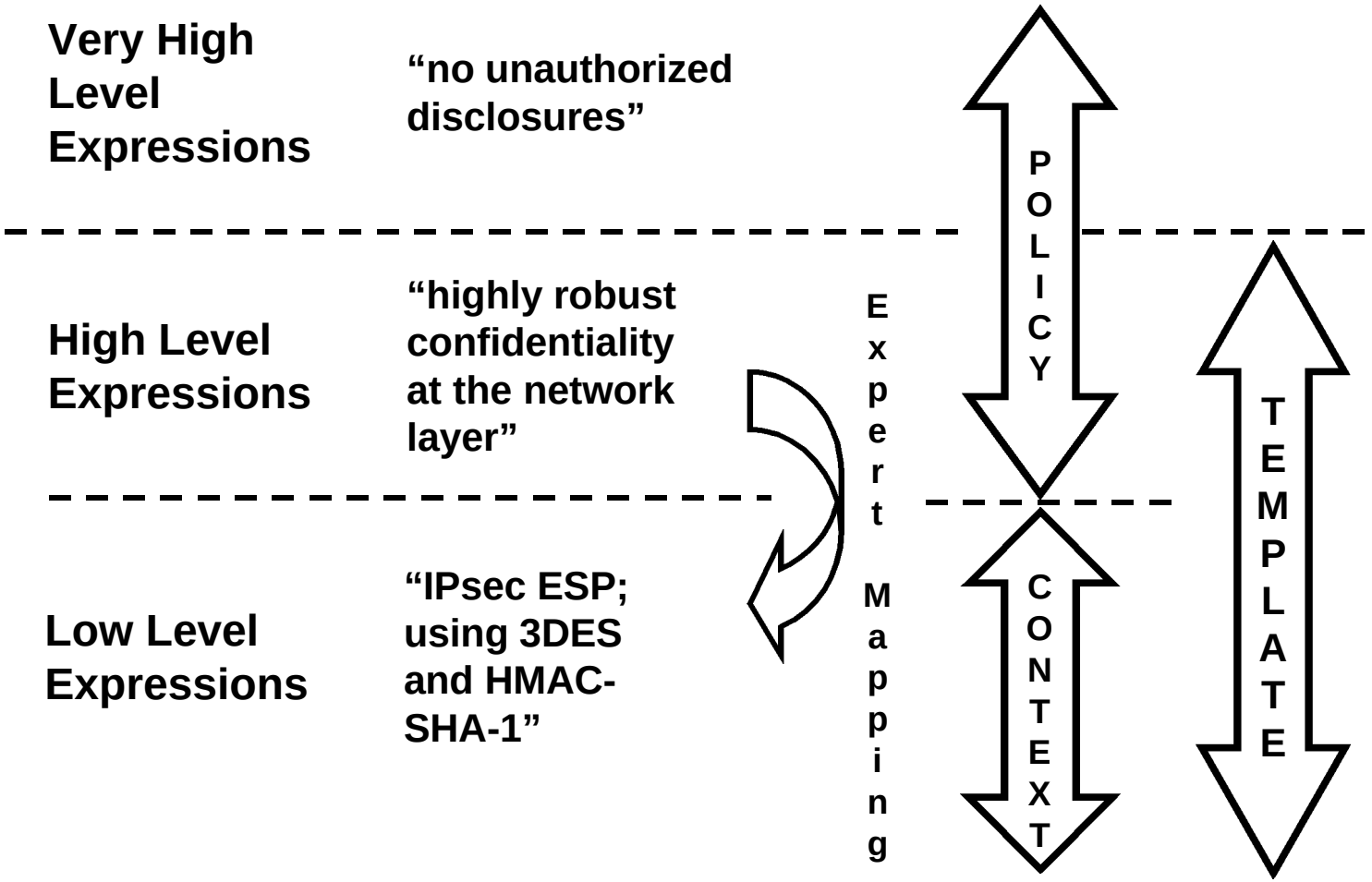
Amortized Group Induction

- Amortizes expensive, e.g. IKE, operations over multiple group memberships for an individual
- Pairwise authenticated key exchange to establish system base key, known only to system manager
  - uses external authentication infrastructure
- Group base key derived from system base key with a one-way function
- For traditional induction, Gn total operations
  - G = number of groups; n = group size
- For amortized induction, N total operations
  - N = universe of possible group members; assumes N << Gn

Multi-Party Application Policies

- Are associated with applications and groups, representing, among other things:
  - managers and participants
  - security services
  - underlying cryptographic mechanisms
  - parameters
  - other security-relevant attributes
- Should be represented and communicated (or negotiated) via a policy specification language and/or policy “construct”
  - e.g., cryptographic context
  - e.g., policy token

Policy, Context and Template



Some Concluding Remarks

- No “one size fits all” group key management method
- Build general framework that allows multiple alternatives
- Specify acceptable methods and other attributes via policy construct, e.g., cryptographic context
- Develop a standardized API for group keying methods
- Follow a phased plan, with both short term and long term objectives