CERIAS

Structure-Based Routing for Secure Content Dissemination

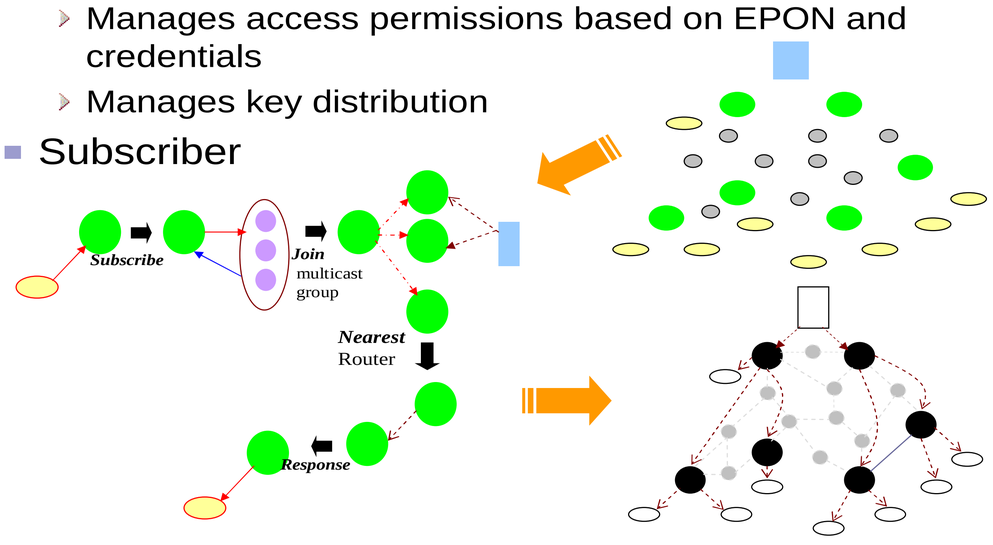
Ashish Kundu, Elisa Bertino
CERIAS, Purdue University, {*ashishk, bertino*}@cs.purdue.edu

Introduction

- Content Dissemination: important to the Web.
  - Hierarchical model (XML) - the *de facto* standard.
- Issues
  - Confidentiality, Integrity & Authenticity: of content.
  - Privacy: of user- and content-specific information.
  - Efficiency: D^3 - Density, Distribution and Dynamicity of content users.
- How to disseminate minimal content in a secure, privacy-preserving manner?
- Pub/Sub dissemination model
- Structure-based content routing using Encrypted Post Order Numbering (EPON).

Structure-Based Pub/Sub

- Content publisher publishes the encoded content
  - Manages access permissions based on EPON and credentials
  - Manages key distribution
- Subscriber

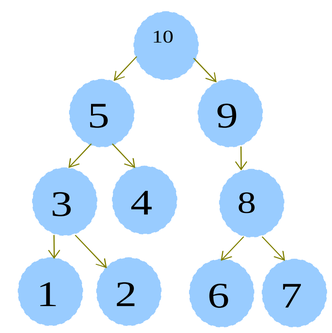


Structure-Based Routing

- EPON-based Structural Identifier(s) embedded onto router through subscription
- Upon receipt of new content, the router
  - Traverses content in DFS
  - At each node *x,* identify subscribers having access to T_x by matching structural identifier S_x.
  - For users, apply access permissions on T_x
  - Delivers all the matching subtrees to the subscribers using PKI.

Some Simple Observations

- Post order number (PON) P_x of a node *x* in tree *T*
  - uniquely determines location of *x* in *T*
  - uniquely determines subtree T_x, root at *x*.
- $P_m \le P_y \le P_x$; *y* is any node in T_x, P_m - lowest post order number in T_x
- $P_L < P_R$; *L*, *R* – left, right siblings in *T*
- Is tree F_z is isomorphic to some T_x - linear and simple!!!
- Pigeonhole Principle => content overlap among users, |Users| > |Content Nodes|



Content Encoding

- Encrypted Post Order Number (EPON) e_x
  - $(e_i,...,e_x, ...e_n) = E^o(f(e_i),..., f(e_x), ... f(e_n))$
    - E^o: encryption function preserving order among PONs.
    - *f* is function that adds noise to P_x while preserving order
- Each node *x* encoded: $C_x = (S_x, I_x, DTD\text{-}URI)$
  - Structural S_x: (e_x, e_{lowest}), e_{lowest} lowest in T_x.
  - Integrity I_x: Merkle Hash of only the node *x*
- Encryption: $E^x_s = K_s(K_m, K_m(K_m, C_x, x))$
  - K_s: shared key between publisher and subscriber.
  - K_m: Merkle Hash of whole content

Content Validation

- Authenticity Check
  - K_m decryption would work if the content is from authentic source
  - *DTD-URI* is used to check if content belongs to such a DTD
- Integrity check
  - Use properties of post order numbers (or EPON) to verify if nodes added or dropped or reordered

References

1. Ashish Kundu, Elisa Bertino, *Scalable and Secure Dissemination of Hierarchical Content*, Under Preparation.