Advanced Topics in Distributed Systems

Dr. Ayman A. Abdel-Hamid

College of Computing and Information Technology

Arab Academy for Science & Technology and Maritime Transport

Domain Name System Papers
Outline

• Discussion of the 3 assigned papers
  ➢ SA+02: Internet Indirection Infrastructure
  ➢ RS04: The Design and Implementation of a Next Generation Name Service for the Internet
  ➢ BLR+04: A Layered Naming Architecture for The Internet
• Overlay-based Internet Indirection Infrastructure

• Rendezvous-based communication abstraction
  ➢ Implement multicast, anycast, and mobility on top of abstraction

• Each packet is associated with an identifier
  ➢ Identifier used by receiver to obtain delivery of the packet

• Service model
  ➢ Sources send packets to a logical identifier
  ➢ Receivers express interest in packets sent to an identifier
id represents a logical rendezvous between the sender’s packets and the receiver’s trigger.
Identifiers are $m$ bits long and there is some exact-match threshold $k$ with $k < m$

In other words, a trigger identifier $id_t$ matches a packet identifier $id$ if and only if $id_t$ is a longest prefix match (among all other trigger identifiers) and this prefix match is at least as long as the exact-match threshold $k$. The value $k$ is chosen to be large enough so that the probability that two randomly chosen identifiers match is negligible. This allows end-hosts to choose the identifiers independently with negligible chance of collision.
Figure 2: Communication abstractions provided by i3. (a) Mobility: The change of the receiver’s address from $R$ to $R'$ is transparent to the sender. (b) Multicast: Every packet $(id, data)$ is forwarded to each receiver $R_i$ that inserts the trigger $(id, R_i)$. (c) Anycast: The packet matches the trigger of receiver $R_2$. $id_p | id_s$ denotes an identifier of size $m$, where $id_p$ represents the prefix of the $k$ most significant bits, and $id_s$ represents the suffix of the $m - k$ least significant bits.
• Cooperative Domain Name System
  ➢ Uses peer-to-peer overlays and a replication framework

• Problems with legacy DNS
  ➢ Security issues: DoS attacks
  ➢ Bottlenecks
  ➢ Implementation errors
  ➢ Latency (1 sec for up to 30% of web object retrievals)
  ➢ Misconfigurations
  ➢ Load Imbalance
  ➢ Update Propagation (because of caching)
Cooperative Domain Name System Architecture

• Based on a proactive caching layer
• Proactive replication framework
What is DHT?

- based on hash tables
- enables identification and retrieving information in distributed systems like some P2P networks
- The whole table is distributed on the network: each node has a part of it.
- Which node responsible for storing data associated with key?
- Nodes form an overlay network

• A DHT would implement a lookup (key) service
  - For storage (convert name to numeric key using a hash function)
  - For Retrieval?
  - For a complete storage solution, would have to deal with replication, caching, and authentication
To implement DHT

- Mapping keys to nodes in a load-balanced way
- Forwarding a lookup for a key to an appropriate node.
- Building routing tables

For more information about DHT see for example
Three levels of name resolution

- From user-level descriptors to service identifiers
- From service identifiers to endpoint identifiers
- From endpoint identifiers to IP addresses
Three levels of name resolution

user-level descriptor (ULD) lookup
(e.g., e-mail address, search string, etc.)

App obtains SIDs corresponding to ULD
using a lookup or search service

SID resolution

App’s session protocol (e.g., HTTP) resolves
SID to EIDs using SID resolution service

EID resolution

Transport protocol resolves EID to
IP addresses using EID resolution service

IP address “resolution” (routing)
EID-level delegation. A source queries on a given EID and is given the IP address of a delegate. The source could also be given the delegate’s EID or multiple EIDs (not shown).