MULTICAST SESSION ADDRESS ALLOCATION & DIRECTORY ARCHITECTURE

Piyush Harsh

PhD Proposal Presentation

March 28 2008

Research Goals

- Propose globally scalable multicast session directory architecture

- Propose user-friendly URL scheme for multicast sessions retrieval

- Efficient multicast address allocation infrastructure

- Possible elimination of globally scoped multicast address collisions among sessions in various administrative domains

Research Motivation

- In near future networks will move to SSM mode from ASM mode
  - Source discovery burden will lie with end users
- ASM still will continue to coexist for foreseeable period
- IPv6 deployment will take time
  - There is a need to better manage scarce multicast addresses in IPv4
  - Globally scoped addresses are an even smaller set
- Usability concerns necessitate some form of URL design for ease of accessing multicast sessions
- Reduction of crosstalk would improve end hosts IP stack performance (requires better address allocation management)

MY CONTRIBUTIONS

○ We proposed a globally scalable multicast address allocation scheme

(Piyush Harsh and Richard Newman - "An overlay solution to IP-Multicast address collision prevention", proceeding of IASTED EuroIMSA 2008, March 2008)

○ We proposed a hierarchical multicast session directory architecture

(Piyush Harsh and Richard Newman - "mDNS - A Proposal for Hierarchical Multicast Session Directory Architecture", submitted to ICOMP 2008)

○ We have also proposed a URL design scheme for accessing multicast content on the Internet.

My Publications

1. Piyush Harsh and Richard Newman - "mDNS - A Proposal for Hierarchical Multicast Session Directory Architecture", submitted to ICOMP 2008.

2. Piyush Harsh and Richard Newman - "An overlay solution to IP-Multicast address collision prevention", proceeding of IASTED EuroIMSA 2008, March 2008.

3. Piyush Harsh and Richard Newman - "Usability and Acceptance of UF-IBA, an Image-Based Authentication System", proceedings of IEEE ICCST 2007, October 2007.

4. Richard E. Newman, Piyush Harsh and Prashant Jayaraman - "Security Analysis of and Proposal for Image-based authentication", proceedings of IEEE ICCST 2005, p. 141, October 2005.

What Follows Next?

- Brief introduction to IP-Multicast
- IP Multicast Address Classifications
- IP Multicast Address Allocation Problem
- Multicast Sessions Directory Problem
- What remains to be done
- Research Timeline

IP-MULTICAST – BRIEF INTRODUCTION

- Internet has 3 modes of data delivery
    - Unicast – packet forwarding based on destination address
    - Anycast – packet forwarding based on destination address
    - Multicast – packet forwarding based on source address (RPF)
- Multicast exists in 2 flavors
    - ASM – Any source multicast
        - In ASM, receivers initially join groups without any specific knowledge of the sources
        - Hosts sends (*,G) Join to multicast enabled routers
        - Data distribution is through distribution trees rooted at RPs.
    - SSM – Source specific multicast
        - Receivers must know about the source whose session they intend to join, sends (S,G) Join to multicast enabled routers
        - Data distribution is via distribution tree rooted at the source

IP-Multicast – brief introduction contd.

- Building blocks of IP-Multicast
  - IGMP (Internet Group Management Protocol)
    - 3 versions – IGMP v1, IGMP v2, IGMP v3 (draft)
    - Allows hosts to express interest in a multicast session to the first hop router
  - Intra-AS multicast protocols
    - DVMRP, MOSPF, CBT, PIM
      - Many of these operate in either DM or SM configuration
      - PIM – Protocol Independent Multicast – uses IP routing table as RPF check table.
      - PIM-SM (sparse mode) – most widely used Intra-AS protocol
  - Inter-AS multicast protocols
    - BGMP – extensions to BGP to enable multicast route characterization

IP-Multicast Address Classification

- Static allocation controlled by IANA – generally used for network control algorithms
  - Link Local Multicast Addresses
    - 224.0.0.0 – 224.0.0.255
    - Network control messages on LAN
  - Specifically Allocated Multicast Address
    - 224.0.1.xxx
    - Assigned to specific network protocol or applications that justified technical merit for their own multicast address
      - Example: 224.0.1.21 – Mtrace
- Static allocation controlled by ISPs – generally assigned to long lived consumer applications ex. Stock ticker.
  - GLOP Static Addressing Scheme
    - 233.{first byte of ASN}.{second byte of ASN}.0/24
    - Allows 255 static allocations per AS number.
    - This range is global in scope

IP-Multicast Address Classification

- Dynamic multicast address allocation range specified by IANA
  - Administratively scoped multicast addresses
    - 239.0.0.0 – 239.255.255.255
    - Reserved for private multicast networks
    - Border routers must filter both incoming / outgoing packets belonging in this address range
  - Globally scoped multicast addresses
    - 224.2.0.0/16 – SAP / SDP range
    - 232.0.0.0/8 – SSM range
- To summarize: Multicast Addresses lies in range 224.0.0.0 – 239.255.255.255. First 4 bits are set to 1110.

IP-Multicast Address Allocation

HOMA – An Overlay Solution to IP-Multicast Address Collision Prevention

Why do we need an allocation service?

- Reduce amount of cross-talk between different applications
- Minimize the probability of address clash in the global scope
- Intelligent allocator could result in improved routing in the network.

DESIGN GOALS

Any global service architecture proposal should try to incorporate these design goals -

- Deployment on existing infrastructure
- Scalability
- High Availability
- Resilience against DDoS
- Low bandwidth Usage

EXISTING SOLUTIONS

- 'sdr' – session directory tool used in MBONE
  - IRMA (Informed Random)
    - Not scalable globally
    - Depends heavily on control message delays and freq.
    - Performance decreases heavily with packet loss rates
  - IPRMA (Informed Partitioned)
    - Uneven utilization of certain partitions
- MASC / BGMP (Prefix / Hierarchical)
- Cyclic (Contiguous Allocation Scheme)
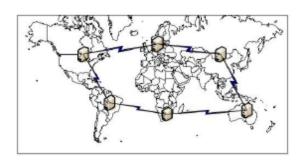
HYBRID OVERLAY-MULTICAST ADDRESS ALLOCATOR (HOMA)
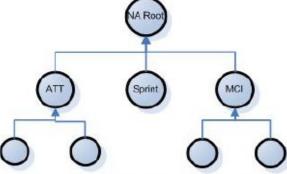

Figure 1: Global TLDs Overlay

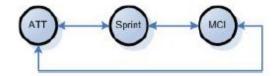Figure 2: ISP Tree rooted at global TLD

Figure 3: Peer n/w among sibling nodes

- HOMA is a hierarchical design
- Root level consists of several global TLDs (Top Level Domain)
- Each global TLD acts as the root of hierarchical tree of regional ISPs and enterprise networks
- All sibling nodes form a peer network among themselves.
- This is facilitated through the parent node (provides the peer network details ex. Multicast channel / key etc.)

HOMA Address Allocation Algorithm

Each HOMA Node maintains these internal variables independently of others:

- α – address demand trend parameter
- β – address release trend parameter
- λ – # of new address requested in a given 5 minute slot
- μ – # of address released in a given 5 minutes slot
- γ – address utilization factor
- δ – additional address anticipated until lease expires
- φ – possible disposable address count
- N – # of 5 minutes slots until address lease expires

HOMA Address Allocation Algorithm

$$\alpha_{new} = \lambda.p + \alpha_{old}(1 - p)$$
$$\beta_{new} = \mu.p' + \beta_{old}(1 - p')$$

$$N = \lceil \text{lease time} - \text{current time} \rceil \div 5$$
$$\delta = [(\alpha - \beta) \times N] - \#free\_addresses\_remaining$$

Pseudo-code for address allocator module –
If incoming request is for a new channel address by a multicast application –

- If a free channel address is available then allocate the address to the requesting application after negotiating the address lease time properly.
  - Update γ, λ

- If a free channel address is not available, then allocate a channel address randomly from the parent's address space.
  - Update λ

If incoming request is to release one of the already allotted addresses by a multicast application –

- If the address belongs to the set owned by this HOMA node, then add it to the free address list.
  - Update γ, μ
- If the address does not belong to the address set owned by the HOMA node, do not add to free address list
  - Update μ

At every 5 minutes interval –

- Recompute α, β
- Set $\lambda = \mu = 0$

HOMA Address Allocation Algorithm

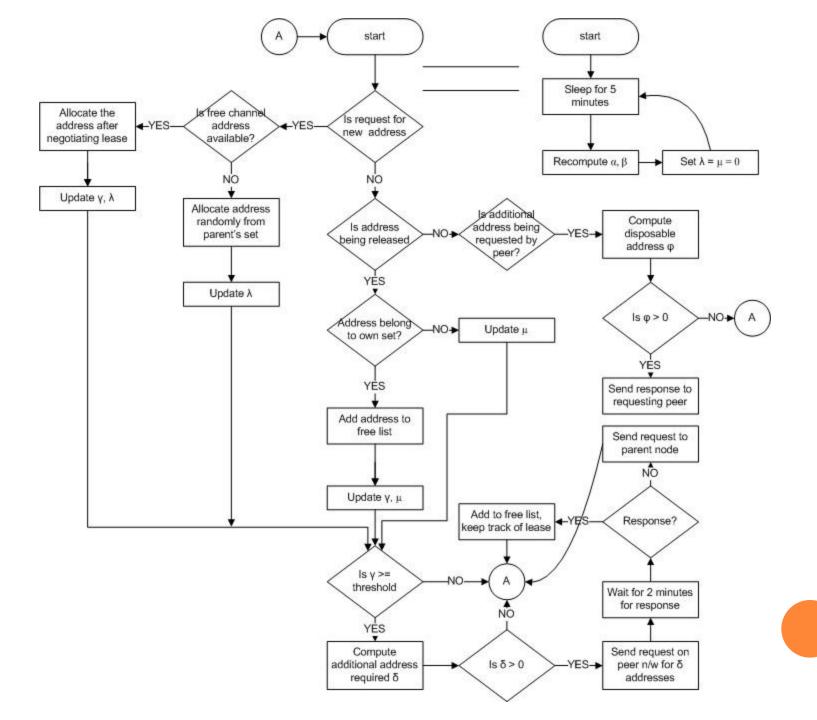After every address allocation / de-allocation check the value of updated γ.

- If $\gamma <$ threshold: Do nothing.
- If $\gamma \geq$ threshold
  - Compute the anticipated additional address required δ
  - If $\delta > 0$, initiate a request for δ number of addresses on the sibling peer network and wait for 2 minutes for responses.
    - If any response comes, add addresses to the free address pool keeping track of the lease associated with those addresses.
    - If no response comes, initiate additional address request to parent HOMA node.

If additional address request is received on the sibling peer network –

- Compute possible disposable address count ϕ using the following relation:

$$\Phi = \#free\_addresses\_remaining - [(\alpha - \beta) \times N]$$

  - If $\phi > 0$, indicate willingness to allocate ϕ set of addresses to the sibling node. Treat this allocation just like any other address allocation.
  - If $\phi \leq 0$, then do nothing.

HOMA Allocation Pseudocode Flowchart

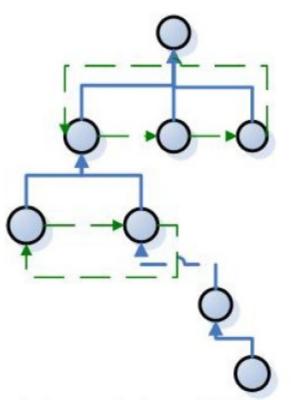# TIME DELAY ANALYSIS OF HOMA

Figure 4: A general scheme of HOMA nodes

- Let π be the probability that additional address demand is satisfied by one or more sibling nodes
- Worst case scenario: node must wait for 2 minutes before sending the request to its parent node
- If tree depth is 'd', then overall delay could be modeled by a recursive equation:

$$\text{Delay} = 2\pi + (2 + \Lambda_d)(1 - \pi)$$

where Λ_d is the delay when the request is made to ones parent.

$$\Lambda_d = 2\pi + (2 + \Lambda_{d-1})(1 - \pi)$$

The value π is experimentally determined

POSSIBLE ADVANTAGES OF HOMA

- Could minimize routing flux because of its hierarchical structure.
- Possibly better address space utilization compared to MASC / BGMP scheme
- Lot better delay characteristics compared to MASC / BGMP which has a 48 hours observation window for address set claim.
- TLDs are well known hosts and their immediate child ISP nodes are also well known, this could be used to prevent DDoS attacks at the top levels
- Algorithm implementable in layer 5, easily deployable on existing infrastructure (possibly on a co-located server alongside multicast router)

MULTICAST SESSION DISCOVERY

mDNS – A Proposal for Hierarchical Multicast Session Directory Architecture

WHY DO WE NEED A MULTICAST SESSION DIRECTORY?

- For multicast content discovery on the fly
- With SSM and IGMP v3, source discovery burden will rest on end users
- Integration of multicast session directory with a clever URL design scheme would make multicast more usable for general user community.
- Keywords based sessions tagging may provide multi-dimensional and more sophisticated search capability to the end user
- This in turn would result in more precise content retrieval in real time.

EXISTING PROPOSALS & SOLUTIONS

- sdr – session directory tool – MBone's favorite tool
  - Used for session creation and advertisement
  - Caches session details received on well known globally scoped multicast channel from other sdr clients
  - Suffers from huge delays in the order of 10s of minutes
  - Performance depends on network delays, packet drop rate
  - Maintains Internet-wide flat directory hierarchy
  - Not scalable if IP-multicast really takes off
  - Cache size may be an issue

EXISTING PROPOSALS & SOLUTIONS

- Light-weight multimedia sessions framework
  - 2 tiered SAP announcement approach to overcome latency issues with sdr
    - Announcements at higher frequency under local scope
  - Split SAP clients in two parts
    - Persistent client runs at border nodes and caches all global advertisements
    - Ephemeral client that contacts the persistent SAP client in order to retrieve sessions list as needed
  - Suffers from many of same sdr drawback
    - Flat hierarchy of Internet wide sessions
    - Would not scale
    - Cache size issue still remains

EXISTING PROPOSALS & SOLUTIONS

- Namburi and Sarac have proposed a 2-tier SSM only session announcement scheme
  - Only for SSM networks
  - Tier 2 server addition in the backbone network may require administrative overhead
  - SAP related delays at tier-1 reduced by increasing SAP bandwidth from 4kbps to 50 kbps
  - Cache size issue still remains at tier-1 servers
- Few proposals based on semantic description of sessions have been proposed
  - Suffer from coarse semantic division
  - Multicast scoping requirements have been generally ignored in all proposals.
- All proposals are push-based approaches

mDNS: DNS-AWARE MULTICAST SESSION DIRECTORY ARCHITECTURE

- Designed on similar lines of unicast DNS hierarchy
- URL scheme for multicast sessions is proposed
- Compared to push-based approaches ours is pull-based design
  - Minimizes bandwidth waste due to periodic state refreshes (even at global scale)
  - Minimizes latency (on the fly information retrieval)
  - Minimizes space waste
    - We do not cache session details at every client node as in sdr
    - Session details are maintained only at domain local servers
- Truly hierarchical design
- Universal (does not depend on ASM or SSM mode)
- If need arises, could be implemented as an IP-unicast overlay too.
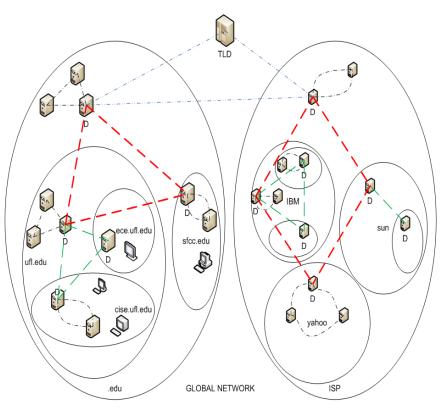
MDNS ARCHITECTURE

- Terminology –
  - MSD_x^y – Multicast Session Directory (MSD) server y in domain x
  - MSD_x^d – Designated MSD server in domain x
  - DNS_x – Domain name server for domain x
  - URS_x – URL registration server in domain x

- Assumption –
  - Each domain / sub-domain knows its DNS server address
  - DNS server knows about it's parent DNS server
  - At least one MSD server coexists with the DNS server at each domain level

- Proposed added entry into DNS record table
  ```
  @MCAST{
          ANYCAST=a.b.c.d    #anycast address of MSD servers in this DNS server's domain
          CMCAST=233.[ASN Byte 1].[ASN Byte 2].xyz    #for group formation with child subnets
          PMCAST=233.[ASN Byte 1].[ASN Byte 2].pqr    #for group formation with parent domain
          URS=x.y.z.w            #address of URS server in this domain (optional)
  }
  ```
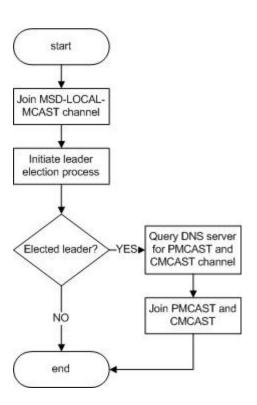
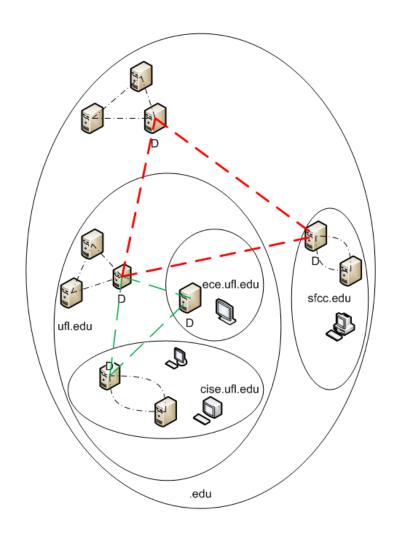# MDNS ARCHITECTURE - CONSTRUCTION

- All MSD servers in a particular administrative domain listens to a fixed (possibly IANA assigned) administratively scoped channel address.

- Refer to that channel as MSD-LOCAL-MCAST

- There will be no cross talk between different administrative domain on MSD-LOCAL-MCAST
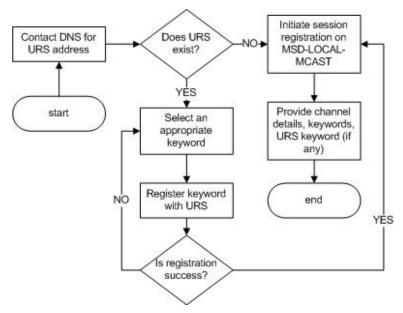
mDNS architecture - construction

- If a domain maintains multiple MSD servers
  - One is selected as designated MSD^d server
  - Any leader election algorithm can be used
- MSD^d joins two additional multicast channels
  - CMCAST (if NULL do not join)
  - PMCAST (if NULL do not join)
- All MSD servers in a particular domain are any-casted at the IP address specified in DNS server's MCAST record at that level.

start → Join MSD-LOCAL-MCAST channel → Initiate leader election process → Elected leader? —YES→ Query DNS server for PMCAST and CMCAST channel → Join PMCAST and CMCAST → end

Elected leader? —NO→ end

MDNS – SESSION REGISTRATION

We place no restriction on choice of keywords in our mDNS scheme, hence our scheme can accommodate wide range of multimedia streams and channels.

MDNS – URL SCHEME

- In mDNS sessions can be accessed directly is creator successfully registered keyword with it's domain's URS server.
- mDNS URL syntax –

<protocol>://<domain URL>/<URS Keyword>

- Protocol could be http or any valid protocol type
- Domain URL helps resolve MSD server located in the creator domain, it begins with *mcast*.
- Example mDNS URL could be –

http://mcast.cise.ufl.edu/gators

- This would refer to multicast session hosted under cise domain with keyword 'gators' registered with cise URS server.

MDNS – DOMAIN SPECIFIC SEARCH

- Domain specific search could also be carried out using mDNS URLs.
- User now must use qualifiers 'search' & 'keyword' in the URL string.
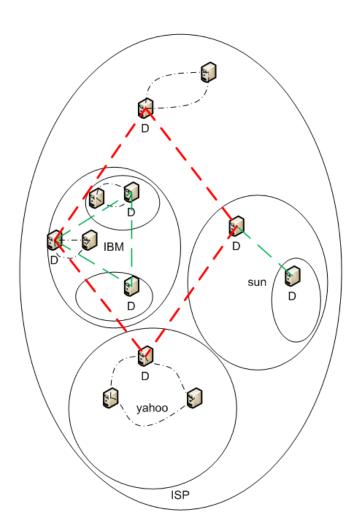- An example:

  mcast.cise.ufl.edu/search=all&keyword=gators

  - Requesting host's multicast search application would resolve the mcast.cise.ufl.edu Anycast address.
  - CISE MSD servers would perform database search for keyword 'gators' and all content types.
  - Content type could be audio, video, whiteboard etc..
  - Search propagates to all sub-domains in top-down fashion starting with domain in the search string.
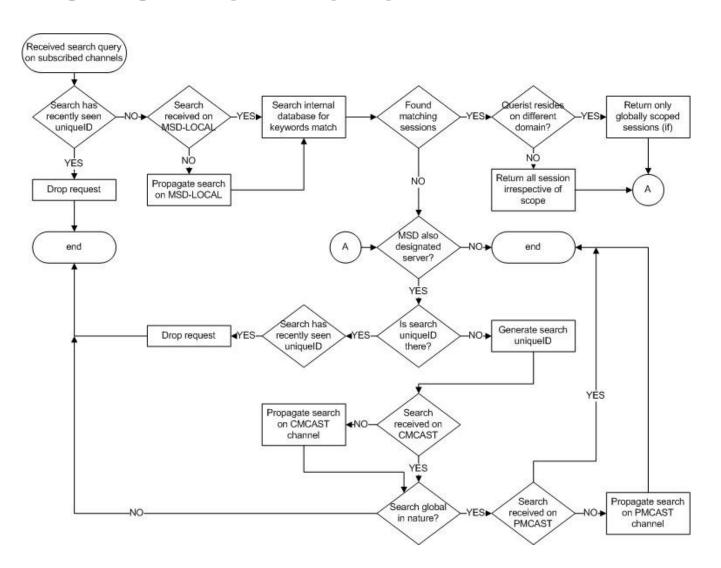
MDNS – GLOBAL SEARCH

- User wishing to perform a global keyword search must contact any MSD server located in it's DNS server's domain.

- Global search is propagated by MSD servers additionally on both PMCAST and CMCAST channels as well as MSD-LOCAL-MCAST channel

- This propagation is done by designated MSD servers only.

- Search is never re-propagated on channel on which it was received.

- Originator MSD^d server adds a unique identification tag along with the search query before propagating it on other channels.

MDNS – SEARCH EXAMPLE

MSD pseudocode for search
received search query on subscribed channels
search sessions internal database for search match
if multicast sessions found then
  if querist resides in another domain then
    return only globally scoped session details (if any)
  else
    return every result found directly to the querist
  end if
end if
if MSD server is a designated MSD server then
  if search unique ID is missing then
    generate unique search ID
  end if
  if query request was not received on CMCAST channel then
    propagate search on CMCAST channel
  end if
  if search is global in nature then
    if query request was not received on PMCAST channel then
      propagate search on PMCAST channel
    end if
  end if
  if search query has self generated previous ID then
    drop search request
  end if
end if

MDNS – SEARCH FLOWCHART

MDNS - ANALYSIS

- mDNS URL would make bookmarking of popular sessions possible just like html bookmarks.
- Drawbacks –
  - Vulnerable to DDoS attacks on particular host
    - Because under current scheme, each MSD server responds to search query results directly to the requesting host.
  - Global search activates every existing MSD servers
    - Could be addressed by smart placements of intelligent caches
- Benefits –
  - Database space saving – under mDNS, session details are stored only at session's hosting domain MSD server
  - We conjecture under mDNS, session query will be much faster compared to session discovery in any sdr based approach.
  - URL benefits are self evident.

What Remains to be Done …

TASKS REMAINING

- System Simulation Setup
  - Simulation parameter space selection
  - Request arrival modeling (Poisson with varying rate)
  - Session duration modeling (log-normal, uniform etc.)
  - HOMA parameters p, p', Π
  - Address space utilization factor comparison with MASC
- Comparison with existing solution like sdr, MASC
- mDNS analysis
  - Space savings compared to sdr and other schemes
  - Search speedup analysis
  - Network stress factor under global searches

TASKS REMAINING – CONT.

- Protocol Design
  - Future readiness in protocol design
  - Control protocol + communication protocol
- mDNS and HOMA integration issues identification
- Time Permitting –
  - mDNS – subscription based approach (threshold based)
  - Intelligent cache placements to reduce global search MSD servers activation issue
  - Detailed security vulnerability analysis for mDNS proposal
    - Possible solutions to identified security threat
  - mDNS database record structure design

RESEARCH TIMELINE

Depends on you committee members …

REALLY!!!

But I would love to finish as soon as possible …

Questions & Comments