ECE 697J – Advanced Topics in Computer Networks

Active Networks
10/09/03
Dynamic Functionality

• Networks implement various packet processing functions
• These applications change over time
  – New protocols
  – New application
  – Etc.
• We need a mechanism to easily deploy such functions
  – Without need for hardware replacement
  – Without complex administration
  – Packets should be able to “choose” functions
• One approach are “Active Networks”
Active Networks

• Basic idea: packets carry processing code
  – Processing code describes how to handle packet
  – Node/router executes code on packet data

• Routers need to provide processing infrastructure
  – Unified, interoperable programming interface/language
  – Safe, efficient execution environment

• Two approaches:
  – Capsule approach: each packet carries code
  – Programmable router: preinstalled functions, packet selects

• What are the major challenges here?
Some AN Research

• Active Network Architecture
  – NodeOS for Interoperability
  – Execution Environments

• Packet format
  – Active Network Encapsulation Protocol

• Programming Languages
  – Java bytecode
  – PLAN & SNAP
**ANEP Header**

- Layered approach:
  - IP
  - ANEP (general AN)
  - Particular EE

<table>
<thead>
<tr>
<th>Bit 0</th>
<th>8</th>
<th>16</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Header</td>
<td>Router Alert option</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ver</td>
<td>Flags</td>
<td>Type ID</td>
<td></td>
</tr>
<tr>
<td>Header Length</td>
<td>Packet Length</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- ANEP Header
  - Source Identifier
  - Destination Identifier
  - Integrity Checksum
  - Authentication Option

<table>
<thead>
<tr>
<th>Ver</th>
<th>Type</th>
<th>Context</th>
<th>Sequence Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Smart Packet Payload</td>
</tr>
</tbody>
</table>
Programming Languages

• Java bytecode:
  – Easy to use (executes on Java VM)
  – Problems?

• Alternative: PLAN & SNAP
  – PLAN (Packet Language for Active Networks)
  – SNAP (Safe and Nimble Active Packets)
  – Very restrictive constructs
    • No backward jumps
    • No loops
  – Achieves resource bounds proportional to packet size
  – Problems?

• Yet another approach:
  – Kernel module without any constraints
  – Problems?
Security for Active Networks

- Packets need to be encrypted in a special way to support hop-by-hop processing:

General Crypto Protected Packet

- header
- identity
- payload
- protection

SANTS Protected Packet

- header
- credential(s)
- static (code & data)
- variable data
- digital signature
- hop-hop integrity
Routing and Path Selection

- Processing is essential part of communication
  - Path needs to be chosen to traverse processing nodes

**Active Pipe Description**

**Physical Network**

- Link costs
- Processing costs
- Candidate sites for second step
- Candidate sites for first step

(mapping onto physical network)
Real-World AN Application

• WaveVideo multicast with congestion adaptation
• WaveVideo encoding:
Scaling with sub bands
Congestion Control

- Each subband is sent in a separate packet
  - Header information indicates subband level
- When link becomes congested
  - Drop high-frequency packets
  - Keep low-frequency packets
- Decision is per-packet and needs local information
  - End-to-end approach is more difficult
- Simple scheme but highly effective
WaveVideo under Congestion

PSNR (dB)

Active queuing

Lossless

Passive queuing
WaveVideo Visualization

Drop occurs in high frequency sub bands

Drop across all sub bands
Active Networks Summary

- Active Networks allow dynamic programmability
  - “Capsules” with Java bytecode
  - Programmable routers with predefined functions
- Lots of system issues need to be solved
  - Save and secure execution
  - Resource sharing
  - Programming abstractions
  - Interoperability
- Good idea but too dynamic
  - Not every end-user (or application) can program the network
- Concepts that have come from AN research
  - Some more or less programmable functions on routers
  - Need for high-performance processing platforms: Network Processors
Next Class

• Introduction of Network Processors
  – Read chapters 11 & 12