

-
-
-

ABone: Outstanding Issues

Active Nets Workshop, Portland, OR
May 24, 2000

Bob Braden, USC/ISI



Some ABone Issues

- Anetd Performance
- JVM Versions
- Active Network Security
- ANEP issues
- ABone Tools
- ABone Management
- Debugging Issues
- NodeOS Integration



Anetd Performance

- Measurement of per-packet delay imposed by Anetd on input packets:
165 usec on 600MHz Pentium-III under Linux



JVM Versions

- *“The JVM is more or less platform-independent, but it is certainly not release-independent”* [Livio R]
- Anetd allows each EE to have its own JVM version
- Java 1.1.x
 - Kaffe users are stuck with 1.1 for a long time(?)
- Java 1.2
 - Important for security features, bug fixes, and zero-copy I/O.
 - Affects Anetd’s class loader and Security Manager
 - FreeBSD: need Release 3.4_stable

Active Network Security

- “Network Security” -- as opposed to Node security --
Makes NodeOS and EEs secure against bad active packets. Integrity, authorization, ...
- We think of this as a general AN problem, perhaps simplified for ABone.
- Maybe, for the near future,
AN security should be an EE-thing?

ANEP Issues

- Include remote (IP addr, UDP port) so EE can distinguish pt-pt virtual links in packet received via Anetd.



- SANEP or ANEP: include security extensions?
- Is ANEP really a new network header? If not, what are src & dest address options for?
- If not Layer 3, is it Layer 2.5, 3.5, or 4.5?
- Need AflowID, for efficient demux to AA with node OS interface.
- Time to define ANEP v2?? (*Hack or fundamental?*)

-
-
-

ABone Tools

- We need a lot of them, to aid ABone users as well as the ABOCC.
- Monitoring, control, & display tools for Anetd and nodes.
- EE-specific tools:
 - Monitoring & debugging
 - Generating and installing configurations
- AA-specific tools:
 - Active traps (“probes”)

ABone Management

- Levels:
 - Node OS [use SNMP?]
 - Anetd
 - EE
 - To what extent is this EE-specific??
 - AA
 - To what extent is this EE-specific? AA-specific?
- Suggest:
 - Issue is “management AAs”, not “management EE”.
 - Could use “any old EE” (with instrumentation hooks), if can present adequate credential to EE and/or node OS.

Active Network Management

- I mean both parsings of the title above.
- Want to use active network techniques for ABone mgt.
 - Active nets => don't need protocol standards;
active NM should => don't need MIB standards.

[“Free us from the tyranny of the MIB and ASN.1!”]

- Active traps (currently called “probes” at ISI): code that monitors specific state in node, sends spontaneous or periodic msgs. Needs some design work.
- Need standard instrumentation interfaces

Management Variables (1)

- Node management
 - Liveness & connectivity (ICMP ping)
 - RTT measurement
 - CPU resources: 'ps' command
 - File system: 'ls' command
 - Routing: 'netstat' command
 - Traffic control: queue lengths
 - Retrieve syslog
- Anetd management
 - For each account: UDP port, version, account.
 - Liveness
 - For each account and each started EE:
 - Name, start time, TypeID, permanent/not, packet count.
 - Retrieve Anetd log file
 - Force Anetd restart

Management Variables (2)

- EE management
 - Version (date, number, ...)
 - Up-time
 - Liveness
 - RTT measurements (in virtual topology)
 - List of loaded AAs
 - For each loaded AA: name, time loaded, resources allocated
 - Virtual topology and virtual-real map
 - Query (virtual) routing table
 - (Virtual) traffic control: queue lengths
 - Retrieve EE log file
 - Halt, reboot (privileged)

Debugging Issues

- Overlaps with network management
- EE debugging
 - Exec EE under jdb/gdb on each node [done]
 - Pipe stdout back to client
- AA debugging
 - Harder, in general: debugging distributed algorithms
 - May need both code breakpoints and “packet breakpoints”
(*Sequester* input packets in EE or in channel, release manually).
 - Another technique: *active probes*

Integration of Node OSs

- [To what extent] can EEs be portable across Node OSs and Unix/Anetd?
- E.g.: EEs written to Node OS spec; in Unix, run on shim layer that maps Posix into Node OS interface?
 - Probably limitations -- how severe?
 - E.g., implementation of general packet filtering semantics of Node OS may require kernel changes in Linux and/or FreeBSD.
- Accept limitations of Unix systems, vs. modify kernels ?