

L05d. Active Networks

Introduction:

- The primary issue once a packet leaves a node in a distributed system is to route the packet reliably and quickly to its destination.
- The intermediate routers have routing tables. The routing table determine, given the packet destination, what the next hub for this packet is. This is done by a simple table lookup.
- We can change this simple lookup to a code execution on the router. This is called Active Networks.
- The packets will carry the payload and code to be executed by the router to determine what the next step for this packet is.
- The idea is to make the nodes on the network active by making them look into the message and figure out what to do with it.



How to implement Active Networks:

- The OS provides Quality of Service (QoS) APIs to the application. These APIs will be used by the application to give the OS an idea about the nature of the message.
- The OS uses QoS to synthesize code and creates the packet (IP-header + code + payload) that will be handed to the internet.
- There're two problems with this approach:
 - Changing the protocol stack to handle active networks is not easy.
 - We cannot expect that every node on the internet will be able to execute the code I'm sending.

Active Node Transfer System (ANTS) Toolkit:

- The ANTS Toolkit is an application level package that takes the payload and QoS from the application and create a capsule (ANTS header + payload) that is passed to the protocol stack.
- The protocol stack will create the packet by adding an IP header to the capsule.
- If the receiving node is a normal one, it will just use the IP header to decide where to send the packet.
- If the receiving node is Active, it will process the ANTS header to decide if the packet has to be de-multiplexed and sent to different routes.
- Another addition is, with ANTS Toolkit, the core IP network will remain unchanged, and the Active Nodes will only be added at the edges of the network.

ANTS Capsules:

- ANTS header consists of:
 - Type field: An MD5 hash of the code to be executed.
 - Prev field: The identity of the up-stream node that successfully executed this capsule.
- ANTS APIs:
 - APIs for routing.
 - APIs for manipulating soft store: Soft store is storage that is available in every routing node for personalizing the network flow with respect to the type field. This is the place where we store the code itself.
 - APIs for querying the node to get information about the node and the network.

Method	Description
int getAddress ()	Get local node address
ChannelObject getChannel ()	Get receive channel
Extension findExtension (String ext)	Locate extended services
long time ()	Get local time
Object put (Object key, Object val, int age)	Put object in soft-store
Object get (Object key)	Get object from soft-store
Object remove (Object key)	Remove object from soft-store
void routeForNode (Capsule c, int n)	Send capsule towards node
void deliverToApp (Capsule c, int a)	Deliver capsule to local application
void log (String msg)	Log a debugging message

- Characteristics of the routing program:
 - Easy to program.
 - Easy to debug and maintain.
 - Very quick.
- When a node receives a capsule, one of two possibilities can happen:
 1. If this node had received capsules of this type before:
 - The node probably already has the capsule's code in its soft store.
 - The node will retrieve the code, execute it and proceed.
 2. If this is the first time the node receives a capsule of this type:
 - The node checks the prev field and send a message to the previous node asking for the code.
 - The previous node retrieves the code from its soft store and sends it to the receiving node.
 - The receiving node computes the fingerprint for the code and compares it with the type field to make sure it's the right code.
 - The receiving node executes that code, store it in its soft store and proceeds.
 - If the code is not present in the soft store of the previous node, the current node will drop the capsule.

How useful are ANTS Toolkit:

- Applications:
 - Protocol independent multicast.
 - Reliable multicast.
 - Congestion notification.
 - Private IP.
 - Any casting.
- Pros: Flexibility from application perspective: You can ignore the physical constraints of the network and add a virtual layer of what needs to be accomplished.
- Cons:
 - Protection threats: We need to make sure that Active nodes should not do any harm to other nodes on the network. ANTS Toolkit addresses the protection threats using:
 1. ANTS runtime safety: ANTS are implemented in Java, and Java sandboxes are used to ensure that the executed code doesn't reach out of the box.
 2. Code spoofing: ANTS uses robust fingerprint.
 3. Soft state integrity: The soft state of each node is limited in size. To prevent a specific network flow from consuming all the available space, a restricted set of APIs is used (as described above).
 - Resource management threats: Flooding the network with capsules. This can be solved by asking at each node whether it will consume more resources than it should.